

Lenovo RackSwitch G8296

Application Guide

for Networking OS 8.2

LenovoTM

Note: Before using this information and the product it supports, read the general information in the *Safety information and Environmental Notices and User Guide* documents on the *Lenovo Documentation CD* and the *Warranty Information* document that comes with the product.

First Edition (August 2015)

© Copyright Lenovo 2015
Portions © Copyright IBM Corporation 2014.

LIMITED AND RESTRICTED RIGHTS NOTICE: If data or software is delivered pursuant a General Services Administration "GSA" contract, use, reproduction, or disclosure is subject to restrictions set forth in Contract No. GS-35F-05925.

Lenovo and the Lenovo logo are trademarks of Lenovo in the United States, other countries, or both.

Contents

Preface	19
Who Should Use This Guide20
What You'll Find in This Guide.21
Additional References24
Typographic Conventions25
Part 1: Getting Started.	27
Chapter 1. Switch Administration	29
Administration Interfaces30
Browser-Based Interface30
Establishing a Connection31
Using Telnet32
Using Secure Shell.32
Using SSH with Password Authentication33
Using SSH with Public Key Authentication33
Using a Web Browser34
Using Simple Network Management Protocol.37
BOOTP/DHCP Client IP Address Services38
Global BOOTP Relay Agent Configuration38
Domain-Specific BOOTP Relay Agent Configuration.39
DHCP Option 8239
Switch Login Levels.40
Setup vs. the Command Line42
Idle Disconnect42
Boot Strict Mode43
Acceptable Cipher Suites45
Configuring Strict Mode46
SSL/TLS Version Limitation47
Limitations.47
Chapter 2. Initial Setup.	49
Information Needed for Setup50
Default Setup Options.50
Stopping and Restarting Setup Manually51
Setup Part 1: Basic System Configuration52
Setup Part 2: Port Configuration54
Setup Part 3: VLANs56
Setup Part 4: IP Configuration57
IP Interfaces57
Loopback Interfaces58
Default Gateways59
IP Routing59
Setup Part 5: Final Steps61
Optional Setup for Telnet Support62

Chapter 3. Service Location Protocol	63
Active DA Discovery	63
SLP Configuration.	64
Chapter 4. Switch Software Management	65
Loading New Software to Your Switch	66
Loading Software via the ISCLI.	66
Loading Software via BBI	67
USB Options	68
USB Boot.	68
USB Copy	69
The Boot Management Menu	70
Recovering from a Failed Upgrade	70
Part 2:. Securing the Switch	73
Chapter 5. Securing Administration	75
Secure Shell and Secure Copy	76
Configuring SSH/SCP Features on the Switch.	76
Configuring the SCP Administrator Password	77
Using SSH and SCP Client Commands	77
SSH and SCP Encryption of Management Messages	79
Generating RSA Host Key for SSH Access	79
SSH/SCP Integration with Radius Authentication	79
SSH/SCP Integration with TACACS+ Authentication	79
End User Access Control	80
Considerations for Configuring End User Accounts	80
Strong Passwords.	80
User Access Control.	81
Listing Current Users	82
Logging into an End User Account	82
Password Fix-Up Mode	82
Chapter 6. Authentication & Authorization Protocols	83
RADIUS Authentication and Authorization	84
How RADIUS Authentication Works	84
Configuring RADIUS on the Switch	84
RADIUS Authentication Features in Networking OS	85
Switch User Accounts	86
RADIUS Attributes for Networking OS User Privileges	86
TACACS+ Authentication	88
How TACACS+ Authentication Works	88
TACACS+ Authentication Features in Networking OS	88
Command Authorization and Logging	90
Configuring TACACS+ Authentication on the Switch	90
LDAP Authentication and Authorization	92
Chapter 7. 802.1X Port-Based Network Access Control	95
Extensible Authentication Protocol over LAN	96
EAPoL Authentication Process	97

EAPoL Message Exchange98
EAPoL Port States99
Guest VLAN99
Supported RADIUS Attributes	100
EAPoL Configuration Guidelines	102
Chapter 8. Access Control Lists103
Summary of Packet Classifiers	104
Summary of ACL Actions	106
Assigning Individual ACLs to a Port	106
ACL Order of Precedence	107
ACL Metering and Re-Marking.	107
ACL Port Mirroring.	108
Viewing ACL Statistics	109
ACL Configuration Examples	110
VLAN Maps	112
Management ACLs	114
Using Storm Control Filters	115
Part 3:. Switch Basics117
Chapter 9. VLANs119
VLANs Overview.	120
VLANs and Port VLAN ID Numbers	120
VLAN Numbers	120
PVID/Native VLAN Numbers	121
VLAN Tagging/Trunk Mode	122
VLAN Topologies and Design Considerations	126
Multiple VLANs with Tagging/Trunk Mode Adapters	126
VLAN Configuration Example	128
Private VLANs	129
Private VLAN Ports	129
Configuration Guidelines	129
Configuration Example	130
Chapter 10. Ports and Trunking131
Configuring QSFP+ Ports	132
Trunking Overview	134
Static Trunks	135
Static Trunk Requirements	135
Static Trunk Group Configuration Rules	135
Configuring a Static Port Trunk	136
Link Aggregation Control Protocol	138
LACP Overview	138
Static LACP Trunks	139
LACP Port Modes.	139
LACP Individual	140
LACP Minimum Links Option	141
Configuring LACP	141
Configurable Trunk Hash Algorithm	142

Chapter 11. Spanning Tree Protocols	145
Spanning Tree Protocol Modes	146
Global STP Control	147
PVRST Mode	147
Port States	148
Bridge Protocol Data Units	148
Bridge Protocol Data Units Overview	148
Determining the Path for Forwarding BPDUs	148
Simple STP Configuration	151
Per-VLAN Spanning Tree Groups	153
Using Multiple STGs to Eliminate False Loops	153
VLANs and STG Assignment	153
Manually Assigning STGs	154
Guidelines for Creating VLANs	155
Rules for VLAN Tagged/Trunk Mode Ports	155
Adding and Removing Ports from STGs	156
The Switch-Centric Model	157
Configuring Multiple STGs	158
Rapid Spanning Tree Protocol	160
Port States	160
RSTP Configuration Guidelines	160
RSTP Configuration Example	160
Multiple Spanning Tree Protocol	162
MSTP Region	162
Common Internal Spanning Tree	162
MSTP Configuration Guidelines	162
MSTP Configuration Examples	163
Port Type and Link Type	165
Edge/Portfast Port	165
Link Type	165
Chapter 12. Virtual Link Aggregation Groups	167
VLAG Capacities	170
VLAGs versus Port Trunks	170
Configuring VLAGs	172
Basic VLAG Configuration	173
VLAG Configuration - VLANs Mapped to MSTI	175
Configuring Health Check	177
VLAGs with VRRP	178
Configuring VLAGs in Multiple Layers	184
VLAG with PIM	187
Traffic Forwarding	187
Health Check	188
Chapter 13. Quality of Service	189
QoS Overview	190
Using ACL Filters	191
Summary of ACL Actions	191
ACL Metering and Re-Marking	192

Using DSCP Values to Provide QoS	193
Differentiated Services Concepts	193
Per Hop Behavior	194
QoS Levels	195
DSCP Re-Marking and Mapping	196
DSCP Re-Marking Configuration Examples	196
Using 802.1p Priority to Provide QoS	199
Queuing and Scheduling	200
Control Plane Protection.	200
WRED with ECN	201
How WRED/ECN work together	201
Configuring WRED/ECN.	202
WRED/ECN Configuration Example.	203
Chapter 14. Precision Time Protocol207
Ordinary Clock Mode	208
Transparent Clock Mode	208
Tracing PTP Packets.	209
Viewing PTP Information	209
Part 4:. Advanced Switching Features211
Chapter 15. OpenFlow213
OpenFlow Overview	214
Switch Profiles	214
OpenFlow Versions	214
OpenFlow Instance	215
Flow Tables	215
Static Flows.	217
Table-Miss	222
Fail Secure Mode	223
Emergency Mode	223
OpenFlow Ports.	224
Data Path ID	226
sFlow Compatibility.	226
OpenFlow Groups	226
Configuring OpenFlow	228
Configuration Example 1 - <i>OpenFlow</i> Boot Profile	228
Configuration Example 2 - <i>Default</i> Boot Profile	231
Feature Limitations	233
Chapter 16. Deployment Profiles235
Available Profiles.	236
Selecting Profiles	237
Automatic Configuration Changes	238
Chapter 17. Virtualization.239
Chapter 18. VMready241
VE Capacity	242

Defining Server Ports	242
VM Group Types	242
Local VM Groups	243
Distributed VM Groups	245
VM Profiles	245
Initializing a Distributed VM Group	245
Assigning Members	246
Synchronizing the Configuration	246
Removing Member VEs	247
VMcheck	248
Virtual Distributed Switch	250
Prerequisites	250
Guidelines	250
Migrating to vDS	251
Virtualization Management Servers	252
Assigning a vCenter	252
vCenter Scans	252
Deleting the vCenter	253
Exporting Profiles	254
VMware Operational Commands	254
Pre-Provisioning VEs	255
VLAN Maps	256
VM Policy Bandwidth Control	257
VM Policy Bandwidth Control Commands	257
Bandwidth Policies vs. Bandwidth Shaping	257
VMready Information Displays	259
VMready Configuration Example	263
Chapter 19. FCoE and CEE	265
Fibre Channel over Ethernet	267
The FCoE Topology	267
FIP Snooping Requirements	268
Converged Enhanced Ethernet	269
Turning CEE On or Off	269
Effects on Link Layer Discovery Protocol	269
Effects on 802.1p Quality of Service	270
Effects on Flow Control	271
FCoE Initialization Protocol Snooping	272
Global FIP Snooping Settings	272
FIP Snooping for Specific Ports	272
Port FCF and ENode Detection	273
FCoE Connection Timeout	273
FCoE ACL Rules	273
FCoE VLANs	274
Viewing FIP Snooping Information	274
Operational Commands	275
FIP Snooping Configuration	275

FCoE Forwarder	277
Configuring FCF	277
Fibre Channel Services	277
FCF VLAN Configuration	278
Full Fabric Zoning	278
Zones	279
Zonesets	279
Defining and Activating Zoning	280
Changing the Active Zoneset	281
FCF Scalability	282
Priority-Based Flow Control	283
Global Configuration	284
PFC Configuration Example	284
Enhanced Transmission Selection	286
802.1p Priority Values	286
Priority Groups	287
PGID	287
Assigning Priority Values to a Priority Group	288
Deleting a Priority Group	288
Allocating Bandwidth	288
Configuring ETS	289
Data Center Bridging Capability Exchange	292
DCBX Settings	292
Configuring DCBX	294
Port Trunking	295
Chapter 20. Edge Virtual Bridging	297
EVB Operations Overview	298
VSIDB Synchronization	298
VLAN Behavior	299
EVB Configuration	300
Limitations	302
Unsupported features	302
Chapter 21. Static Multicast ARP	303
Configuring Static Multicast ARP	304
Configuration Example	304
Limitations	305
Chapter 22. Unified Fabric Port	307
UFP Limitations	308
Virtual Ports Modes	309
vPort-S-Tag Mapping	309
vPort-VLAN Mapping	309
UFP vPort Mode	309
UFP Bandwidth Provisioning	312
UFP Strict Bandwidth Provisioning Mode	312

Using UFP with Other RackSwitch G8296 Features	313
Layer 2 Failover	313
Increased VLAN Limits	313
VMReady	313
802.1Qbg	313
UFP Configuration Examples	314
Example 1: Access Mode	314
Example 2: Trunk Mode	315
Example 3: Auto-VLAN Mode	317
Example 4: Tunnel Mode	318
Example 5: FCoE Mode	319
Example 6: Layer 2 Failover Configuration	320
Part 5:. IP Routing	323
Chapter 23. Basic IP Routing	325
IP Routing Benefits	326
Routing Between IP Subnets	326
Example of Subnet Routing	327
Using VLANs to Segregate Broadcast Domains	328
Configuration Example	328
ARP - Local Proxy	331
Network Address Translation	332
Static NAT	333
Unidirectional NAT Translation.	334
Network Address Port Translation	334
Dynamic NAT	334
NAT Proxy ARP	335
Global NAT Settings	335
Scaling and Limitations	336
ECMP Static Routes.	337
ECMP Route Hashing	337
Configuring ECMP Static Routes	338
Dynamic Host Configuration Protocol.	339
Chapter 24. Policy-Based Routing	341
PBR Policies and ACLs	341
Applying PBR ACLs	341
Configuring Route Maps	342
Match Clauses	342
Set Clauses.	342
Configuring Health Check	343
Example PBR Configuration	345
Configuring PBR with other Features	346
Unsupported Features	346
Dynamic PBR	347
Features and Limitations.	347
Example Configuration	347

Chapter 25. Routed Ports.	.349
Overview	349
Configuring a Routed Port	351
Configuring OSPF on Routed Ports	351
OSPF Configuration Example	352
Configuring RIP on Routed Ports	352
RIP Configuration Example	352
Configuring PIM on Routed Ports	353
PIM Configuration Example	353
Configuring BGP on Routed Ports	354
Configuring IGMP on Routed Ports	354
Limitations	355
Chapter 26. Internet Protocol Version 6.	.357
IPv6 Limitations	358
IPv6 Address Format	359
IPv6 Address Types	360
IPv6 Address Autoconfiguration	362
IPv6 Interfaces	363
Neighbor Discovery	364
Supported Applications	366
Configuration Guidelines	367
IPv6 Configuration Examples	368
Chapter 27. IPsec with IPv6.	.371
IPsec Protocols	372
Using IPsec with the RackSwitch G8296	373
Setting up Authentication	373
Creating an IKEv2 Proposal	374
Importing an IKEv2 Digital Certificate	374
Generating an IKEv2 Digital Certificate	375
Enabling IKEv2 Preshared Key Authentication	375
Setting Up a Key Policy	375
Using a Manual Key Policy	377
Using a Dynamic Key Policy	379
Chapter 28. Routing Information Protocol.	.381
Distance Vector Protocol	382
Stability	382
Routing Updates	382
RIPv1	383
RIPv2	383
RIPv2 in RIPv1 Compatibility Mode	383
RIP Features	384
RIP Configuration Example	385
Chapter 29. Internet Group Management Protocol	.387
IGMP Terms	388
How IGMP Works	389
IGMP Capacity and Default Values	390

IGMP Snooping	391
IGMP Querier	391
IGMP Groups	392
IGMPv3 Snooping	392
IGMP Snooping Configuration Guidelines	394
IGMP Snooping Configuration Example	395
Advanced Configuration Example: IGMP Snooping	396
Prerequisites	397
Configuration.	397
Troubleshooting	401
IGMP Relay	404
Configuration Guidelines	404
Configure IGMP Relay	405
Advanced Configuration Example: IGMP Relay	406
Prerequisites	406
Configuration.	407
Troubleshooting	410
Additional IGMP Features.	413
FastLeave	413
IGMP Filtering	413
Static Multicast Router.	414
Chapter 30. Multicast Listener Discovery	415
MLD Terms	416
How MLD Works	417
MLD Querier.	418
Dynamic M routers	419
MLD Capacity and Default Values	420
Configuring MLD	421
Chapter 31. Border Gateway Protocol	423
Internal Routing Versus External Routing	424
Route Reflector	425
Restrictions.	427
Forming BGP Peer Routers	428
Static Peers.	428
Dynamic Peers	428
Loopback Interfaces	430
What is a Route Map?.	430
Next Hop Peer IP Address	431
Incoming and Outgoing Route Maps	431
Precedence.	432
Configuration Overview	432
Aggregating Routes.	434
Redistributing Routes.	434
BGP Communities	435
BGP Attributes.	436
Selecting Route Paths in BGP.	438
Equal Cost Multi-Path.	438
Multipath Relax	438

BGP Failover Configuration	439
Default Redistribution and Route Aggregation Example	441
Chapter 32. OSPF	443
OSPFv2 Overview	444
Types of OSPF Areas	444
Types of OSPF Routing Devices	445
Neighbors and Adjacencies	446
The Link-State Database	446
The Shortest Path First Tree	448
Internal Versus External Routing	448
OSPFv2 Implementation in Networking OS	449
Configurable Parameters	449
Defining Areas	450
Assigning the Area Index	450
Using the Area ID to Assign the OSPF Area Number	451
Attaching an Area to a Network	451
Interface Cost	452
Electing the Designated Router and Backup	452
Summarizing Routes	452
Default Routes	453
Virtual Links	454
Router ID	454
Authentication	455
Configuring Plain Text OSPF Passwords	456
Configuring MD5 Authentication	456
Host Routes for Load Balancing	457
Loopback Interfaces in OSPF	458
OSPF Features Not Supported in This Release	458
OSPFv2 Configuration Examples	459
Example 1: Simple OSPF Domain	460
Example 2: Virtual Links	462
Example 3: Summarizing Routes	466
Verifying OSPF Configuration	467
OSPFv3 Implementation in Networking OS	468
OSPFv3 Differences from OSPFv2	468
OSPFv3 Requires IPv6 Interfaces	468
OSPFv3 Uses Independent Command Paths	468
OSPFv3 Identifies Neighbors by Router ID	469
Other Internal Improvements	469
OSPFv3 Limitations	469
OSPFv3 Configuration Example	469
Neighbor Configuration Example	472
Chapter 33. Protocol Independent Multicast	473
PIM Overview	474
Supported PIM Modes and Features	475

Basic PIM Settings	476
Globally Enabling or Disabling the PIM Feature.	476
Defining a PIM Network Component	476
Defining an IP Interface for PIM Use	476
PIM Neighbor Filters	477
Additional Sparse Mode Settings	479
Specifying the Rendezvous Point	479
Influencing the Designated Router Selection	479
Specifying a Bootstrap Router	480
Configuring a Loopback Interface	480
Using PIM with Other Features.	482
PIM Configuration Examples	483
Part 6:. High Availability Fundamentals	487
Chapter 34. Basic Redundancy	489
Trunking for Link Redundancy.	490
Virtual Link Aggregation	490
Hot Links	491
Forward Delay	491
Preemption	491
FDB Update	491
Configuration Guidelines	491
Configuring Hot Links	492
Chapter 35. Layer 2 Failover	493
Monitoring Trunk Links.	493
Setting the Failover Limit	494
Manually Monitoring Port Links	495
L2 Failover with Other Features	496
Static Trunks	496
LACP	496
Spanning Tree Protocol	496
Configuration Guidelines	497
Configuring Layer 2 Failover.	497
Chapter 36. Virtual Router Redundancy Protocol	499
VRRP Overview	500
VRRP Components	500
VRRP Operation	501
Selecting the Master VRRP Router	502
Failover Methods.	503
Active-Active Redundancy.	503
Virtual Router Group	503
Networking OS Extensions to VRRP	504
Virtual Router Deployment Considerations	505
High Availability Configurations.	507
VRRP High-Availability Using Multiple VIRs	507
VRRP High-Availability Using VLAGs	511

Part 7: Network Management	.513
Chapter 37. Link Layer Discovery Protocol	.515
LLDP Overview	516
Enabling or Disabling LLDP	517
Global LLDP Setting	517
Transmit and Receive Control	517
LLDP Transmit Features	518
Scheduled Interval	518
Minimum Interval	518
Time-to-Live for Transmitted Information	518
Trap Notifications	519
Changing the LLDP Transmit State	519
Types of Information Transmitted	520
LLDP Receive Features	522
Types of Information Received	522
Viewing Remote Device Information	522
Time-to-Live for Received Information	524
LLDP Example Configuration	526
Chapter 38. Simple Network Management Protocol	.527
SNMP Version 1 & Version 2	528
SNMP Version 3	529
Configuring SNMP Trap Hosts	531
SNMP MIBs	534
Switch Images and Configuration Files	541
Loading a New Switch Image	542
Loading a Saved Switch Configuration	542
Saving the Switch Configuration	543
Saving a Switch Dump	543
Chapter 39. NETCONF	.545
NETCONF Overview	546
XML Requirements	547
Installing the NETCONF Client	548
Using Juniper Perl Client	550
Establishing a NETCONF Session	551
NETCONF Operations	553

Protocol Operations Examples	554
<get-config>	554
<edit-config>	555
<copy-config>	557
<delete-config>	558
<lock>	558
<unlock>.	559
<get>	560
<close-session>	561
<kill-session>.	561
<get-configuration>	562
<get-interface-information>	563
Part 8:. Monitoring	567
Chapter 40. Remote Monitoring	569
RMON Overview.	569
RMON Group 1—Statistics	570
RMON Group 2—History	571
History MIB Object ID.	571
Configuring RMON History	571
RMON Group 3—Alarms	572
Alarm MIB objects	572
Configuring RMON Alarms	572
RMON Group 9—Events	573
Chapter 41. sFlow	575
sFlow Statistical Counters	575
sFlow Network Sampling	575
sFlow Example Configuration	576
Chapter 42. Port Mirroring	577
Part 9:. Appendices	579
Appendix A. Glossary	581
Appendix B. Getting help and technical assistance.	583
Appendix C. Notices	585
Trademarks	587
Important Notes	588
Recycling Information.	589
Particulate Contamination.	590
Telecommunication Regulatory Statement	591

Electronic Emission Notices	592
Federal Communications Commission (FCC) Statement	592
Industry Canada Class A Emission Compliance Statement	592
Avis de Conformité à la Réglementation d'Industrie Canada	592
Australia and New Zealand Class A Statement	592
European Union EMC Directive Conformance Statement.	592
Germany Class A Statement	593
Japan VCCI Class A Statement	594
Japan Electronics and Information Technology Industries Association (JEITA) Statement	595
Korea Communications Commission (KCC) Statement.	595
Russia Electromagnetic Interference (EMI) Class A statement.	596
People's Republic of China Class A electronic emission statement.	597
Taiwan Class A compliance statement.	598
Index599

Preface

The *Lenovo RackSwitch G8296 Application Guide for Networking OS 8.2* describes how to configure and use the Lenovo Network Operating System 8.2 software on the RackSwitch G8296 (referred to as G8296 throughout this document). For documentation on installing the switch physically, see the *Installation Guide* for your G8296.

Who Should Use This Guide

This guide is intended for network installers and system administrators engaged in configuring and maintaining a network. The administrator should be familiar with Ethernet concepts, IP addressing, Spanning Tree Protocol, and SNMP configuration parameters.

What You'll Find in This Guide

This guide will help you plan, implement, and administer Networking OS software. Where possible, each section provides feature overviews, usage examples, and configuration instructions. The following material is included:

Part 1: Getting Started

This material is intended to help those new to N/OS products with the basics of switch management. This part includes the following chapters:

- [Chapter 1, “Switch Administration,”](#) describes how to access the G8296 to configure the switch and view switch information and statistics. This chapter discusses a variety of manual administration interfaces, including local management via the switch console, and remote administration via Telnet, a web browser, or via SNMP.
- [Chapter 2, “Initial Setup,”](#) describes how to use the built-in Setup utility to perform first-time configuration of the switch.
- [Chapter 4, “Switch Software Management,”](#) describes how to update the N/OS software operating on the switch.

Part 2: Securing the Switch

- [Chapter 5, “Securing Administration,”](#) describes methods for using Secure Shell for administration connections, and configuring end-user access control.
- [Chapter 6, “Authentication & Authorization Protocols,”](#) describes different secure administration for remote administrators. This includes using Remote Authentication Dial-in User Service (RADIUS), as well as TACACS+ and LDAP.
- [Chapter 8, “Access Control Lists,”](#) describes how to use filters to permit or deny specific types of traffic, based on a variety of source, destination, and packet attributes.

Part 3: Switch Basics

- [Chapter 9, “VLANs,”](#) describes how to configure Virtual Local Area Networks (VLANs) for creating separate network segments, including how to use VLAN tagging for devices that use multiple VLANs. This chapter also describes Protocol-based VLANs, and Private VLANs.
- [Chapter 10, “Ports and Trunking,”](#) describes how to group multiple physical ports together to aggregate the bandwidth between large-scale network devices.
- [Chapter 11, “Spanning Tree Protocols,”](#) discusses how Spanning Tree Protocol (STP) configures the network so that the switch selects the most efficient path when multiple paths exist. Covers Rapid Spanning Tree Protocol (RSTP), Per-VLAN Rapid Spanning Tree (PVRST), and Multiple Spanning Tree Protocol (MSTP).
- [Chapter 12, “Virtual Link Aggregation Groups,”](#) describes using Virtual Link Aggregation Groups (VLAG) to form trunks spanning multiple VLAG-capable aggregator switches.

- [Chapter 13, “Quality of Service,”](#) discusses Quality of Service (QoS) features, including IP filtering using Access Control Lists (ACLs), Differentiated Services, and IEEE 802.1p priority values.

Part 4: Advanced Switching Features

- [Chapter 17, “Virtualization,”](#) provides an overview of allocating resources based on the logical needs of the data center, rather than on the strict, physical nature of components.
- [Chapter 18, “VMready,”](#) discusses virtual machine (VM) support on the G8296.

Part 5: IP Routing

- [Chapter 23, “Basic IP Routing,”](#) describes how to configure the G8296 for IP routing using IP subnets, BOOTP, and DHCP Relay.
- [Chapter 26, “Internet Protocol Version 6,”](#) describes how to configure the G8296 for IPv6 host management.
- [Chapter 27, “IPsec with IPv6,”](#) describes how to configure Internet Protocol Security (IPsec) for securing IP communications by authenticating and encrypting IP packets, with emphasis on Internet Key Exchange version 2, and authentication/confidentiality for OSPFv3.
- [Chapter 28, “Routing Information Protocol,”](#) describes how the N/OS software implements standard Routing Information Protocol (RIP) for exchanging TCP/IP route information with other routers.
- [Chapter 29, “Internet Group Management Protocol,”](#) describes how the N/OS software implements IGMP Snooping or IGMP Relay to conserve bandwidth in a multicast-switching environment.
- [Chapter 30, “Multicast Listener Discovery,”](#) describes how Multicast Listener Discovery (MLD) is used with IPv6 to support host users requests for multicast data for a multicast group.
- [Chapter 31, “Border Gateway Protocol,”](#) describes Border Gateway Protocol (BGP) concepts and features supported in N/OS.
- [Chapter 32, “OSPF,”](#) describes key Open Shortest Path First (OSPF) concepts and their implemented in N/OS, and provides examples of how to configure your switch for OSPF support.

Part 6: High Availability Fundamentals

- [Chapter 34, “Basic Redundancy,”](#) describes how the G8296 supports redundancy through trunking, and hotlinks.
- [Chapter 35, “Layer 2 Failover,”](#) describes how the G8296 supports high-availability network topologies using Layer 2 Failover.
- [Chapter 36, “Virtual Router Redundancy Protocol,”](#) describes how the G8296 supports high-availability network topologies using Virtual Router Redundancy Protocol (VRRP).

Part 7: Network Management

- [Chapter 37, “Link Layer Discovery Protocol,”](#) describes how Link Layer Discovery Protocol helps neighboring network devices learn about each others’ ports and capabilities.
- [Chapter 38, “Simple Network Management Protocol,”](#) describes how to configure the switch for management through an SNMP client.

Part 8: Monitoring

- [Chapter 40, “Remote Monitoring,”](#) describes how to configure the RMON agent on the switch, so that the switch can exchange network monitoring data.
- [Chapter 41, “sFlow,”](#) described how to use the embedded sFlow agent for sampling network traffic and providing continuous monitoring information to a central sFlow analyzer.
- [Chapter 42, “Port Mirroring,”](#) discusses tools how copy selected port traffic to a monitor port for network analysis.

Part 9: Appendices

- [Appendix A, “Glossary,”](#) describes common terms and concepts used throughout this guide.
- [Appendix B, “Getting help and technical assistance,”](#) provides details on where to go for additional information about Lenovo and Lenovo products.
- [Appendix C, “Notices,”](#) contains copyright and trademark notices.

Additional References

Additional information about installing and configuring the G8296 is available in the following guides:

- *Lenovo RackSwitch G8296 ISCLI Command Reference for Networking OS 8.2*
- *Lenovo RackSwitch G8296 Installation Guide*

Typographic Conventions

The following table describes the typographic styles used in this book.

Table 1. *Typographic Conventions*

Typeface or Symbol	Meaning	Example
ABC123	This type is used for names of commands, files, and directories used within the text. It also depicts on-screen computer output and prompts.	View the <code>readme.txt</code> file. Main#
ABC123	This bold type appears in command examples. It shows text that must be typed in exactly as shown.	Main# sys
<ABC123>	This italicized type appears in command examples as a parameter placeholder. Replace the indicated text with the appropriate real name or value when using the command. Do not type the brackets. This also shows book titles, special terms, or words to be emphasized.	To establish a Telnet session, enter: host# telnet <i><IP address></i> Read your <i>User's Guide</i> thoroughly.
[]	Command items shown inside brackets are optional and can be used or excluded as the situation demands. Do not type the brackets.	host# ls [-a]
	The vertical bar () is used in command examples to separate choices where multiple options exist. Select only one of the listed options. Do not type the vertical bar.	host# set left right
AaBbCc123	This block type depicts menus, buttons, and other controls that appear in Web browsers and other graphical interfaces.	Click the Save button.

Part 1: Getting Started

Chapter 1. Switch Administration

Your RackSwitch G8296 (G8296) is ready to perform basic switching functions right out of the box. Some of the more advanced features, however, require some administrative configuration before they can be used effectively.

The extensive Lenovo Network Operating System switching software included in the G8296 provides a variety of options for accessing the switch to perform configuration, and to view switch information and statistics.

This chapter discusses the various methods that can be used to administer the switch.

Administration Interfaces

Networking OS provides a variety of user-interfaces for administration. These interfaces vary in character and in the methods used to access them: some are text-based, and some are graphical; some are available by default, and some require configuration; some can be accessed by local connection to the switch, and others are accessed remotely using various client applications. For example, administration can be performed using any of the following:

- A built-in, text-based command-line interface and menu system for access via serial-port connection or an optional Telnet or SSH session
- The built-in Browser-Based Interface (BBI) available using a standard web-browser
- SNMP support for access through network management software such as IBM Director or HP OpenView

The specific interface chosen for an administrative session depends on user preferences, as well as the switch configuration and the available client tools.

In all cases, administration requires that the switch hardware is properly installed and turned on. (see the *RackSwitch G8296 Installation Guide*).

Browser-Based Interface

The Browser-based Interface (BBI) provides access to the common configuration, management and operation features of the G8296 through your Web browser.

For more information, refer to the *BBI Quick Guide*.

Establishing a Connection

The factory default settings permit initial switch administration through *only* the built-in serial port. All other forms of access require additional switch configuration before they can be used.

Remote access using the network requires the accessing terminal to have a valid, routable connection to the switch interface. The client IP address may be configured manually, or an IPv4 address can be provided automatically through the switch using a service such as DHCP or BOOTP relay (see [“BOOTP/DHCP Client IP Address Services” on page 38](#)), or an IPv6 address can be obtained using IPv6 stateless address configuration.

Note: Throughout this manual, *IP address* is used in places where either an IPv4 or IPv6 address is allowed. IPv4 addresses are entered in dotted-decimal notation (for example, 10.10.10.1), while IPv6 addresses are entered in hexadecimal notation (for example, 2001:db8:85a3::8a2e:370:7334). In places where only one type of address is allowed, *IPv4 address* or *IPv6 address* is specified.

1. Log on to the switch.
2. Enter IP interface mode.

```
RS8296> enable
RS8296# configure terminal
RS8296(config)# interface ip <IP interface number>
```

3. Configure the management IP interface/mask.
 - o Using IPv4:

```
RS8296(config-ip-if)# ip address <management interface IPv4 address>
RS8296(config-ip-if)# ip netmask <IPv4 subnet mask>
```

- o Using IPv6:

```
RS8296(config-ip-if)# ipv6 address <management interface IPv6 address>
RS8296(config-ip-if)# ipv6 prefixlen <IPv6 prefix length>
```

4. Configure the VLAN, and enable the interface.

```
RS8296(config-ip-if)# vlan 1
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
```

5. Configure the default gateway.

- o If using IPv4:

```
RS8296(config)# ip gateway <gateway number> address <IPv4 address>
RS8296(config)# ip gateway <gateway number> enable
```

- o If using IPv6:

```
RS8296(config)# ip gateway6 <gateway number> address <IPv6 address>
RS8296(config)# ip gateway6 <gateway number> enable
```

Once you configure the IP address and have a network connection, you can use the Telnet program from an external management station to access and control the switch. Once the default gateway is enabled, the management station and your switch do not need to be on the same IP subnet.

The G8296 supports an industry standard command-line interface (ISCLI) that you can use to configure and control the switch over the network using the Telnet program. You can use the ISCLI to perform many basic network management functions. In addition, you can configure the switch for management using an SNMP-based network management system or a Web browser.

For more information, see the documents listed in [“Additional References” on page 24](#).

Using Telnet

A Telnet connection offers the convenience of accessing the switch from a workstation connected to the network. Telnet access provides the same options for user and administrator access as those available through the console port.

By default, Telnet access is enabled. Use the following commands to disable or re-enable Telnet access:

```
RS8296(config)# [no] access telnet enable
```

Once the switch is configured with an IP address and gateway, you can use Telnet to access switch administration from any workstation connected to the management network.

To establish a Telnet connection with the switch, run the Telnet program on your workstation and issue the following Telnet command:

```
telnet <switch IPv4 or IPv6 address>
```

You will then be prompted to enter a password as explained [“Switch Login Levels” on page 40](#).

Two attempts are allowed to log in to the switch. After the second unsuccessful attempt, the Telnet client is disconnected via TCP session closure.

Using Secure Shell

Although a remote network administrator can manage the configuration of a G8296 via Telnet, this method does not provide a secure connection. The Secure Shell (SSH) protocol enables you to securely log into another device over a network to execute commands remotely. As a secure alternative to using Telnet to manage switch configuration, SSH ensures that all data sent over the network is encrypted and secure.

The switch can do only one session of key/cipher generation at a time. Thus, a SSH/SCP client will not be able to login if the switch is doing key generation at that time. Similarly, the system will fail to do the key generation if a SSH/SCP client is logging in at that time.

The supported SSH encryption and authentication methods are:

- Server Host Authentication: Client RSA-authenticates the switch when starting each connection
- Key Exchange: ecdh-sha2-nistp521, ecdh-sha2-nistp384, ecdh-sha2-nistp256, ecdh-sha2-nistp224, ecdh-sha2-nistp192, rsa2048-sha256, rsa1024-sha1, diffie-hellman-group-exchange-sha256, diffie-hellman-group-exchange-sha1, diffie-hellman-group14-sha1, diffie-hellman-group1-sha1
- Encryption: aes128-ctr, aes128-cbc, rijndael128-cbc, blowfish-cbc,3des-cbc, arcfour256, arcfour128, arcfour
- MAC: hmac-sha1, hmac-sha1-96, hmac-md5, hmac-md5-96
- User Authentication: Local password authentication, public key authentication, RADIUS, TACACS+

Lenovo Network Operating System implements the SSH version 2.0 standard and is confirmed to work with SSH version 2.0-compliant clients such as the following:

- OpenSSH_5.4p1 for Linux
- Secure CRT Version 5.0.2 (build 1021)
- Putty SSH release 0.60

Using SSH with Password Authentication

By default, the SSH feature is disabled. Once the IP parameters are configured and the SSH service is enabled, you can access the command line interface using an SSH connection.

To establish an SSH connection with the switch, run the SSH program on your workstation by issuing the SSH command, followed by the switch IPv4 or IPv6 address:

```
# ssh <switch IP address>
```

You will then be prompted to enter a password as explained [“Switch Login Levels” on page 40](#).

Using SSH with Public Key Authentication

SSH can also be used for switch authentication based on asymmetric cryptography. Public encryption keys can be uploaded on the switch and used to authenticate incoming login attempts based on the clients’ private encryption key pairs. After a predefined number of failed public key login attempts, the switch reverts to password-based authentication.

To set up public key authentication:

1. Enable SSH:

```
RS8296(config)# ssh enable
```

2. Import the public key file using SFTP or TFTP for the admin user account::

```
RS8296(config)# copy {sftp|tftp} public-key  
Port type ["DATA"/"MGT"]: mgt  
Address or name of remote host: 9.43.101.151  
Source file name: 11.key  
Username of the public key: admin  
Confirm download operation (y/n) ? y
```

Notes:

- When prompted to input a username, a valid user account name must be entered. If no username is entered, the key is stored on the switch, and can be assigned to a user account later.
 - A user account can have up to 100 public keys set up on the switch.
3. Configure a maximum number of 3 failed public key authentication attempts before the system reverts to password-based authentication:

```
RS8296(config)# ssh maxauthattempts 3
```

Once the public key is configured on the switch, the client can use SSH to login from a system where the private key pair is set up:

```
# ssh <switch IP address>
```

Using a Web Browser

The switch provides a Browser-Based Interface (BBI) for accessing the common configuration, management and operation features of the G8296 through your Web browser.

By default, BBI access via HTTP is enabled on the switch.

You can also access the BBI directly from an open Web browser window. Enter the URL using the IP address of the switch interface (for example, `http://<IPv4 or IPv6 address>`).

Configuring HTTP Access to the BBI

By default, BBI access via HTTP is enabled on the switch.

To disable or re-enable HTTP access to the switch BBI, use the following commands:

```
RS8296(config)# access http enable    (Enable HTTP access)  
  
-or-  
  
RS8296(config)# no access http enable (Disable HTTP access)
```

The default HTTP web server port to access the BBI is port 80. However, you can change the default Web server port with the following command:

```
RS8296(config)# access http port <TCP port number>
```

To access the BBI from a workstation, open a Web browser window and type in the URL using the IP address of the switch interface (for example, `http://<IPv4 or IPv6 address>`).

Configuring HTTPS Access to the BBI

The BBI can also be accessed via a secure HTTPS connection.

1. Enable HTTPS.

By default, BBI access via HTTPS is disabled on the switch. To enable BBI Access via HTTPS, use the following command:

```
RS8296(config)# access https enable
```

2. Set the HTTPS server port number (optional).

To change the HTTPS Web server port number from the default port 443, use the following command:

```
RS8296(config)# access https port <x>
```

3. Generate the HTTPS certificate.

Accessing the BBI via HTTPS requires that you generate a certificate to be used during the key exchange. A default certificate is created the first time HTTPS is enabled, but you can create a new certificate defining the information you want to be used in the various fields.

```
RS8296(config)# access https generate-certificate
Country Name (2 letter code) []: <country code>
State or Province Name (full name) []: <state>
Locality Name (eg, city) []: <city>
Organization Name (eg, company) []: <company>
Organizational Unit Name (eg, section) []: <org.unit>
Common Name (eg, YOUR name) []: <name>
Email (eg, email address) []: <email address>
Confirm generating certificate? [y/n]: y
Generating certificate. Please wait (approx 30 seconds)
restarting SSL agent
```

4. Save the HTTPS certificate.

The certificate is valid only until the switch is rebooted. To save the certificate so it is retained beyond reboot or power cycles, use the following command:

```
RS8296(config)# access https save-certificate
```

When a client (such as a web browser) connects to the switch, the client is asked to accept the certificate and verify that the fields match what is expected. Once BBI access is granted to the client, the BBI can be used as described in the *Lenovo Network Operating System 8.2 BBI Quick Guide*.

Browser-Based Interface Summary

The BBI is organized at a high level as follows:

Context buttons—These buttons allow you to select the type of action you wish to perform. The *Configuration* button provides access to the configuration elements for the entire switch. The *Statistics* button provides access to the switch statistics and state information. The *Dashboard* button allows you to display the settings and operating status of a variety of switch features.

Navigation Window—This window provides a menu list of switch features and functions:

- **System**—this folder provides access to the configuration elements for the entire switch.
- **Switch Ports**—Configure each of the physical ports on the switch.
- **Port-Based Port Mirroring**—Configure port mirroring behavior.
- **Layer 2**—Configure Layer 2 features for the switch.
- **RMON Menu**—Configure Remote Monitoring features for the switch.
- **Layer 3**—Configure Layer 3 features for the switch.
- **QoS**—Configure Quality of Service features for the switch.
- **Access Control**—Configure Access Control Lists to filter IP packets.
- **Virtualization**—Configure VMready.

Using Simple Network Management Protocol

N/OS provides Simple Network Management Protocol (SNMP) version 1, version 2, and version 3 support for access through any network management software, such as IBM Director or HP-OpenView.

Note: SNMP read and write functions are enabled by default. For best security practices, if SNMP is not needed for your network, it is recommended that you disable these functions prior to connecting the switch to the network.

To access the SNMP agent on the G8296, the read and write community strings on the SNMP manager must be configured to match those on the switch. The default read community string on the switch is `public` and the default write community string is `private`.

The read and write community strings on the switch can be configured using the following commands:

```
RS8296(config)# snmp-server read-community <1-32 characters>
-and-
RS8296(config)# snmp-server write-community <1-32 characters>
```

The SNMP manager must be able to reach any one of the IP interfaces on the switch.

For the SNMP manager to receive the SNMPv1 traps sent out by the SNMP agent on the switch, configure the trap host on the switch with the following commands:

```
RS8296(config)# snmp-server trap-source <trap source IP interface>
RS8296(config)# snmp-server host <IPv4 address> <trap host community string>
```

To restrict SNMP access to specific IPv4 subnets, use the following commands:

```
RS8296(config)# access management-network <IPv4 address> <subnet mask> snmp-ro
-and-
RS8296(config)# access management-network <IPv4 address> <subnet mask> snmp-rw
```

For IPv6 networks, use:

```
RS8296(config)# access management-network6 <IPv6 address> <IPv6 prefix length>
snmp-ro
-and-
RS8296(config)# access management-network6 <IPv6 address> <IPv6 prefix length>
snmp-rw
```

Note: Subnets allowed for SNMP read-only access must not overlap with subnets allowed for SNMP read-write access.

For more information on SNMP usage and configuration, see [“Simple Network Management Protocol” on page 527](#).

BOOTP/DHCP Client IP Address Services

For remote switch administration, the client terminal device must have a valid IP address on the same network as a switch interface. The IP address on the client device may be configured manually, or obtained automatically using IPv6 stateless address configuration, or an IPv4 address may be obtained automatically via BOOTP or DHCP relay as discussed in the next section.

The G8296 can function as a relay agent for Bootstrap Protocol (BOOTP) or DHCP. This allows clients to be assigned an IPv4 address for a finite lease period, reassigning freed addresses later to other clients.

Acting as a relay agent, the switch can forward a client's IPv4 address request to up to five BOOTP/DHCP servers. In addition to the five global BOOTP/DHCP servers, up to five domain-specific BOOTP/DHCP servers can be configured for each of up to 10 VLANs.

When a switch receives a BOOTP/DHCP request from a client seeking an IPv4 address, the switch acts as a proxy for the client. The request is forwarded as a UDP Unicast MAC layer message to the BOOTP/DHCP servers configured for the client's VLAN, or to the global BOOTP/DHCP servers if no domain-specific BOOTP/DHCP servers are configured for the client's VLAN. The servers respond to the switch with a Unicast reply that contains the IPv4 default gateway and the IPv4 address for the client. The switch then forwards this reply back to the client.

DHCP is described in RFC 2131, and the DHCP relay agent supported on the G8296 is described in RFC 1542. DHCP uses UDP as its transport protocol. The client sends messages to the server on port 67 and the server sends messages to the client on port 68.

BOOTP and DHCP relay are collectively configured using the BOOTP commands and menus on the G8296.

Global BOOTP Relay Agent Configuration

To enable the G8296 to be a BOOTP (or DHCP) forwarder, enable the BOOTP relay feature, configure up to four global BOOTP server IPv4 addresses on the switch, and enable BOOTP relay on the interface(s) on which the client requests are expected.

Generally, it is best to configure BOOTP for the switch IP interface that is closest to the client, so that the BOOTP server knows from which IPv4 subnet the newly allocated IPv4 address will come.

In the G8296 implementation, there are no primary or secondary BOOTP servers. The client request is forwarded to all the global BOOTP servers configured on the switch (if no domain-specific servers are configured). The use of multiple servers provides failover redundancy. However, no health checking is supported.

1. Use the following commands to configure global BOOTP relay servers:

```
RS8296(config)# ip bootp-relay enable
RS8296(config)# ip bootp-relay server <1-5> address <IPv4 address>
```

2. Enable BOOTP relay on the appropriate IP interfaces.

BOOTP/DHCP Relay functionality may be assigned on a per-interface basis using the following commands:

```
RS8296(config)# interface ip <interface number>
RS8296(config-ip-if)# relay
RS8296(config-ip-if)# exit
```

Domain-Specific BOOTP Relay Agent Configuration

Use the following commands to configure up to five domain-specific BOOTP relay agents for each of up to 10 VLANs:

```
RS8296(config)# ip bootp-relay bcast-domain <1-10> vlan <VLAN number>
RS8296(config)# ip bootp-relay bcast-domain <1-10> server <1-5> address
<IPv4 address>
RS8296(config)# ip bootp-relay bcast-domain <1-10> enable
```

As with global relay agent servers, domain-specific BOOTP/DHCP functionality may be assigned on a per-interface basis (see [Step 2](#) in [page 39](#)).

DHCP Option 82

DHCP Option 82 provides a mechanism for generating IP addresses based on the client device's location in the network. When you enable the DHCP relay agent option on the switch, it inserts the relay agent information option 82 in the packet, and sends a unicast BOOTP request packet to the DHCP server. The DHCP server uses the option 82 field to assign an IP address, and sends the packet, with the original option 82 field included, back to the relay agent. DHCP relay agent strips off the option 82 field in the packet and sends the packet to the DHCP client.

Configuration of this feature is optional. The feature helps resolve several issues where untrusted hosts access the network. See RFC 3046 for details.

Use the following commands to configure DHCP Option 82:

```
RS8296(config)# ip bootp-relay information enable (Enable Option 82)
RS8296(config)# ip bootp-relay enable (Enable DHCP relay)
RS8296(config)# ip bootp-relay server <1-5> address <IP address>
```

Switch Login Levels

To enable better switch management and user accountability, three levels or *classes* of user access have been implemented on the G8296. Levels of access to CLI, Web management functions, and screens increase as needed to perform various switch management tasks. Conceptually, access classes are defined as follows:

- User interaction with the switch is completely passive—nothing can be changed on the G8296. Users may display information that has no security or privacy implications, such as switch statistics and current operational state information.
- Operators can only effect temporary changes on the G8296. These changes will be lost when the switch is rebooted/reset. Operators have access to the switch management features used for daily switch operations. Because any changes an operator makes are undone by a reset of the switch, operators cannot severely impact switch operation.
- Administrators are the only ones that may make permanent changes to the switch configuration—changes that are persistent across a reboot/reset of the switch. Administrators can access switch functions to configure and troubleshoot problems on the G8296. Because administrators can also make temporary (operator-level) changes as well, they must be aware of the interactions between temporary and permanent changes.

Access to switch functions is controlled through the use of unique user names and passwords. Once you are connected to the switch via console, remote Telnet, or SSH, you are prompted to enter a password. The default user names/password for each access level are listed in the following table.

Note: It is recommended that you change default switch passwords after initial configuration and as regularly as required under your network security policies.

Table 2. *User Access Levels - Default Settings*

User Account	Password	Description and Tasks Performed	Status
user	user	The User has no direct responsibility for switch management. He or she can view all switch status information and statistics, but cannot make any configuration changes to the switch.	Disabled
oper	oper	The Operator manages all functions of the switch. The Operator can reset ports, except the management ports.	Disabled
admin	admin	The superuser Administrator has complete access to all menus, information, and configuration commands on the G8296, including the ability to change both the user and administrator passwords.	Enabled

Note: Access to each user level (except admin account) can be disabled by setting the password to an empty value. To disable admin account, use the command: `RS8296(config)# no access user administrator-enable`. Admin account can be disabled only if there is at least one user account enabled and configured with administrator privilege.

Setup vs. the Command Line

Once the administrator password is verified, you are given complete access to the switch. If the switch is still set to its factory default configuration, the system will ask whether you wish to run Setup (see [“Initial Setup” on page 49](#)), a utility designed to help you through the first-time configuration process. If the switch has already been configured, the command line is displayed instead.

Idle Disconnect

By default, the switch will disconnect your Telnet session after 10 minutes of inactivity. This function is controlled by the idle timeout parameter, which can be set from 0 to 60 minutes, where 0 means the session will never timeout.

Use the following command to set the idle timeout value:

```
RS8296(config)# system idle <0-60>
```

Boot Strict Mode

The implementations specified in this section are compliant with National Institute of Standards and Technology (NIST) Special Publication (SP) 800-131A.

The RackSwitch G8296 can operate in two boot modes:

- Compatibility mode (default): This is the default switch boot mode. This mode may use algorithms and key lengths that may not be allowed/acceptable by NIST SP 800-131A specification. This mode is useful in maintaining compatibility with previous releases and in environments that have lesser data security requirements.
- Strict mode: Encryption algorithms, protocols, and key lengths in strict mode are compliant with NIST SP 800-131A specification.

When in boot strict mode, the switch uses Secure Sockets Layer (SSL)/Transport Layer Security (TLS) 1.2 protocols to ensure confidentiality of the data to and from the switch.

Before enabling strict mode, ensure the following:

- The software version on all connected switches is Networking OS 8.2.
- The supported protocol versions and cryptographic cipher suites between clients and servers are compatible. For example: if using SSH to connect to the switch, ensure that the SSH client supports SSHv2 and a strong cipher suite that is compliant with the NIST standard.
- Compliant Web server certificate is installed on the switch, if using BBI.
- A new self-signed certificate is generated for the switch (`RS8296(config)# access https generate-certificate`). The new certificate is generated using 2048-bit RSA key and SHA-256 digest.
- Protocols that are not NIST SP 800-131A compliant must be disabled or not used.
- Only SSHv2 or higher is used.
- The current configuration, if any, is saved in a location external to the switch. When the switch reboots, both the startup and running configuration are lost.
- Only protocols/algorithms compliant with NIST SP 800-131A specification are used/enabled on the switch. Please see the NIST SP 800-131A publication for details. The following table lists the acceptable protocols and algorithms:

Table 3. *Acceptable Protocols and Algorithms*

Protocol/Function	Strict Mode Algorithm	Compatibility Mode Algorithm
BGP	BGP does not comply with NIST SP 800-131A specification. When in strict mode, BGP is disabled. However, it can be enabled, if required.	Acceptable
Certificate Generation	RSA-2048 SHA-256	RSA 2048 SHA 256
Certificate Acceptance	RSA 2048 or higher SHA 224 or higher	RSA SHA, SHA2

Table 3. *Acceptable Protocols and Algorithms*

Protocol/Function	Strict Mode Algorithm	Compatibility Mode Algorithm
HTTPS	TLS 1.2 only See "Acceptable Cipher Suites" on page 45;	TLS 1.0, 1.1, 1.2 See "Acceptable Cipher Suites" on page 45;
IKE		
Key Exchange	DH Group 24	DH group 1, 2, 5, 14, 24
Encryption	3DES, AES-128-CBC	3DES, AES-128-CBC
Integrity	HMAC-SHA1	HMAC-SHA1, HMAC-MD5
IPSec		
AH	HMAC-SHA1	HMAC-SHA1, HMAC-MD5
ESP	3DES, AES-128-CBC, HMAC-SHA1	3DES, AES-128-CBC, HMAC-SHA1, HMAC-MD5
LDAP	LDAP does not comply with NIST SP 800-131A specification. When in strict mode, LDAP is disabled. However, it can be enabled, if required.	Acceptable
OSPF	OSPF does not comply with NIST SP 800-131A specification. When in strict mode, OSPF is disabled. However, it can be enabled, if required.	Acceptable
RADIUS	RADIUS does not comply with NIST SP 800-131A specification. When in strict mode, RADIUS is disabled. However, it can be enabled, if required.	Acceptable
Random Number Generator	NIST SP 800-90A AES CTR DRBG	NIST SP 800-90A AES CTR DRBG
Secure NTP	Secure NTP does not comply with NIST SP 800-131A specification. When in strict mode, secure NTP is disabled. However, it can be enabled, if required.	Acceptable
SLP	SHA-256 or higher RSA/DSA 2048 or higher	
SNMP	SNMPv3 only AES-128-CFB-128/SHA1 Note: Following algorithms are acceptable if you choose to support old SNMPv3 factory default users: AES-128-CFB/SHA1 DES/MD5 AES-128-CFB-128/SHA1	SNMPv1, SNMPv2, SNMPv3 DES/MD5, AES-128-CFB-128/SHA1

Table 3. *Acceptable Protocols and Algorithms*

Protocol/Function	Strict Mode Algorithm	Compatibility Mode Algorithm
SSH/SFTP		
Host Key	SSH-RSA	SSH-RSA
Key Exchange	ECDH-SHA2-NISTP521 ECDH-SHA2-NISTP384 ECDH-SHA2-NISTP256 ECDH-SHA2-NISTP224 RSA2048-SHA256 DIFFIE-HELL- MAN-GROUP-EXCHANGE-SHA2 56 DIFFIE-HELL- MAN-GROUP-EXCHANGE-SHA1	ECDH-SHA2-NISTP521 ECDH-SHA2-NISTP384 ECDH-SHA2-NISTP256 ECDH-SHA2-NISTP224 ECDH-SHA2-NISTP192 RSA2048-SHA256 RSA1024-SHA1 DIFFIE-HELL- MAN-GROUP-EXCHANGE-SH A256 DIFFIE-HELL- MAN-GROUP-EXCHANGE-SH A1 DIFFIE-HELL- MAN-GROUP14-SHA1 DIFFIE-HELL- MAN-GROUP1-SHA1
Encryption	AES128-CTR AES128-CBC 3DES-CBC	AES128-CTR AES128-CBC RIJNDAEL128-CBC BLOWFISH-CBC 3DES-CBC
MAC	HMAC-SHA1 HMAC-SHA1-96	HMAC-SHA1 HMAC-SHA1-96 HMAC-MD5 HMAC-MD5-96
TACACS+	TACACS+ does not comply with NIST SP 800-131A specification. When in strict mode, TACACS+ is disabled. However, it can be enabled, if required.	Acceptable

Acceptable Cipher Suites

The following cipher suites are acceptable (listed in the order of preference) when the RackSwitch G8296 is in compatibility mode:

Table 4. *List of Acceptable Cipher Suites in Compatibility Mode*

Cipher ID	Key Exchange	Authentication	Encryption	MAC	Cipher Name
0xC027	ECDHE	RSA	AES_128_CBC	SHA256	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
0xC013	ECDHE	RSA	AES_128_CBC	SHA1	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
0xC012	ECDHE	RSA	3DES	SHA1	SSL_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA
0xC011	ECDHE	RSA		SHA1	SSL_ECDHE_RSA_WITH_RC4_128_SHA
0x002F	RSA	RSA	AES_128_CBC	SHA1	TLS_RSA_WITH_AES_128_CBC_SHA
0x003C	RSA	RSA	AES_128_CBC	SHA256	TLS_RSA_WITH_AES_128_CBC_SHA256
0x0005	RSA	RSA		SHA1	SSL_RSA_WITH_RC4_128_SHA

Table 4. List of Acceptable Cipher Suites in Compatibility Mode (continued)

Cipher ID	Key Exchange	Authentication	Encryption	MAC	Cipher Name
0x000A	RSA	RSA	3DES	SHA1	SSL_RSA_WITH_3DES_EDE_CBC_SHA
0x0033	DHE	RSA	AES-128_CBC	SHA1	TLS_DHE_RSA_WITH_AES_128_CBC_SHA
0x0067	DHE	RSA	AES_128_CBC	SHA256	TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
0x0016	DHE	RSA	3DES	SHA1	SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA

The following cipher suites are acceptable (listed in the order of preference) when the RackSwitch G8296 is in strict mode:

Table 5. List of Acceptable Cipher Suites in Strict Mode

Cipher ID	Key Exchange	Authentication	Encryption	MAC	Cipher Name
0xC027	ECDHE	RSA	AES_128_CBC	SHA256	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA256
0xC013	ECDHE	RSA	AES_128_CBC	SHA1	TLS_ECDHE_RSA_WITH_AES_128_CBC_SHA
0xC012	ECDHE	RSA	3DES	SHA1	SSL_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA
0x0033	DHE	RSA	AES-128_CBC	SHA1	TLS_DHE_RSA_WITH_AES_128_CBC_SHA
0x0067	DHE	RSA	AES_128_CBC	SHA256	TLS_DHE_RSA_WITH_AES_128_CBC_SHA256
0x0016	DHE	RSA	3DES	SHA1	SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA
0x002F	RSA	RSA	AES_128_CBC	SHA1	TLS_RSA_WITH_AES_128_CBC_SHA
0x003C	RSA	RSA	AES_128_CBC	SHA256	TLS_RSA_WITH_AES_128_CBC_SHA256
0x000A	RSA	RSA	3DES	SHA1	SSL_RSA_WITH_3DES_EDE_CBC_SHA

Configuring Strict Mode

To change the switch mode to boot strict mode, use the following command:

```
RS8296(config)# [no] boot strict enable
```

When strict mode is enabled, you will see the following message:

```
Warning, security strict mode limits the cryptographic algorithms used by secure protocols on this switch. Please see the documentation for full details, and verify that peer devices support acceptable algorithms before enabling this mode. The mode change will take effect after reloading the switch and the configuration will be wiped during the reload. System will enter security strict mode with default factory configuration at next boot up.
```

```
Do you want SNMPV3 support old default users in strict mode (y/n)?
```

For SNMPv3 default users, see [“SNMP Version 3” on page 529](#).

When strict mode is disabled, the following message is displayed:

```
Warning, disabling security strict mode. The mode change will take effect after reloading the switch.
```

You must reboot the switch for the boot strict mode enable/disable to take effect.

SSL/TLS Version Limitation

Each of the following successive encryption protocol versions provide more security and less compatibility: SSLv3, TLS1.0, TLS1.1, TLS1.2. When negotiating the encryption protocol during the SSL handshake, the switch will accept, by default, the latest (and most secure) protocol version supported by the client equipment. To enforce a minimal level of security acceptable for the connections, use the following command:

```
RS8296(config)# ssl minimum-version {tls10|tls11|tls12}
```

Limitations

In Networking OS 8.2, consider the following limitation/restrictions if you need to operate the switch in boot strict mode:

- Power ITEs and High-Availability features do not comply with NIST SP 800-131A specification.
- The G8296 will not discover Platform agents/Common agents that are not in strict mode.
- Web browsers that do not use TLS 1.2 cannot be used.
- Limited functions of the switch managing Windows will be available.

Chapter 2. Initial Setup

To help with the initial process of configuring your switch, the Lenovo Network Operating System software includes a Setup utility. The Setup utility prompts you step-by-step to enter all the necessary information for basic configuration of the switch.

Setup can be activated manually from the command line interface any time after login.

Information Needed for Setup

Setup requests the following information:

- Basic system information
 - Date & time
 - Whether to use Spanning Tree Group or not
- Optional configuration for each port
 - Speed, duplex, flow control, and negotiation mode (as appropriate)
 - Whether to use VLAN trunk mode/tagging or not (as appropriate)
- Optional configuration for each VLAN
 - Name of VLAN
 - Which ports are included in the VLAN
- Optional configuration of IP parameters
 - IP address/mask and VLAN for each IP interface
 - IP addresses for default gateway
 - Whether IP forwarding is enabled or not

Default Setup Options

The Setup prompt appears automatically whenever you login as the system administrator under the factory default settings.

1. Connect to the switch.

After connecting, the login prompt appears.

Enter Password:

2. Enter **admin** as the default administrator password.

Note: If the default **admin** login is unsuccessful, or if the administrator Main Menu appears instead, the system configuration has probably been changed from the factory default settings. If desired, return the switch to its factory default configuration.

3. Enter **y** to begin the initial configuration of the switch, or **n** to bypass the Setup facility.

Stopping and Restarting Setup Manually

Stopping Setup

To abort the Setup utility, press <Ctrl-C> during any Setup question. When you abort Setup, the system will prompt:

```
Would you like to run from top again? [y/n]
```

Enter **n** to abort Setup, or **y** to restart the Setup program at the beginning.

Restarting Setup

You can restart the Setup utility manually at any time by entering the following command at the configuration prompt:

```
RS8296(config)# setup
```

Setup Part 1: Basic System Configuration

When Setup is started, the system prompts:

"Set Up" will walk you through the configuration of System Date and Time, Spanning Tree, Port Speed/Mode, VLANs, and IP interfaces. [type Ctrl-C to abort "Set Up"]

1. Enter **y** if you will be configuring VLANs. Otherwise enter **n**.

If you decide not to configure VLANs during this session, you can configure them later using the configuration menus, or by restarting the Setup facility. For more information on configuring VLANs, see the *Lenovo Network Operating System Application Guide*.

Next, the Setup utility prompts you to input basic system information.

2. Enter the year of the current date at the prompt:

System Date:
Enter year [2009]:

Enter the four-digits that represent the year. To keep the current year, press <Enter>.

3. Enter the month of the current system date at the prompt:

System Date:
Enter month [1]:

Enter the month as a number from 1 to 12. To keep the current month, press <Enter>.

4. Enter the day of the current date at the prompt:

Enter day [3]:

Enter the date as a number from 1 to 31. To keep the current day, press <Enter>.

The system displays the date and time settings:

System clock set to 18:55:36 Wed Jan 28, 2009.

5. Enter the hour of the current system time at the prompt:

System Time:
Enter hour in 24-hour format [18]:

Enter the hour as a number from 00 to 23. To keep the current hour, press <Enter>.

6. Enter the minute of the current time at the prompt:

Enter minutes [55]:

Enter the minute as a number from 00 to 59. To keep the current minute, press <Enter>.

7. Enter the seconds of the current time at the prompt:

```
Enter seconds [37]:
```

Enter the seconds as a number from 00 to 59. To keep the current second, press <Enter>. The system then displays the date and time settings:

```
System clock set to 8:55:36 Wed Jan 28, 2009.
```

8. Turn Spanning Tree Protocol on or off at the prompt:

```
Spanning Tree:  
Current Spanning Tree Group 1 setting: ON  
Turn Spanning Tree Group 1 OFF? [y/n]
```

Enter **y** to turn off Spanning Tree, or enter **n** to leave Spanning Tree on.

Setup Part 2: Port Configuration

Note: When configuring port options for your switch, some prompts and options may be different.

1. Select whether you will configure VLANs and VLAN trunk mode/tagging for ports:

```
Port Config:
Will you configure VLANs and VLAN Tagging/Trunk-Mode for ports? [y/n]
```

If you wish to change settings for VLANs, enter **y**, or enter **n** to skip VLAN configuration.

Note: The sample screens that appear in this document might differ slightly from the screens displayed by your system. Screen content varies based on the firmware versions and options that are installed.

2. Select the port to configure, or skip port configuration at the prompt:

If you wish to change settings for individual ports, enter the number of the port you wish to configure. To skip port configuration, press <Enter> without specifying any port and go to [“Setup Part 3: VLANs” on page 56](#).

3. Configure Gigabit Ethernet port flow parameters.

The system prompts:

```
Gig Link Configuration:
Port Flow Control:
Current Port EXT1 flow control setting:    both
Enter new value ["rx"/"tx"/"both"/"none"]:
```

Enter **rx** to enable receive flow control, **tx** for transmit flow control, **both** to enable both, or **none** to turn flow control off for the port. To keep the current setting, press <Enter>.

4. Configure Gigabit Ethernet port autonegotiation mode.

If you selected a port that has a Gigabit Ethernet connector, the system prompts:

```
Port Auto Negotiation:
Current Port EXT1 autonegotiation:        on
Enter new value ["on"/"off"]:
```

Enter **on** to enable port autonegotiation, **off** to disable it, or press <Enter> to keep the current setting.

5. If configuring VLANs, enable or disable VLAN trunk mode/tagging for the port.

If you have selected to configure VLANs back in Part 1, the system prompts:

```
Port VLAN tagging/trunk mode config (tagged/trunk mode port can be a member
of multiple VLANs)
Current VLAN tagging/trunk mode support:    disabled
Enter new VLAN tagging/trunk mode support [d/e]:
```

Enter **d** to disable VLAN trunk mode/tagging for the port or enter **e** to enable VLAN tagging for the port. To keep the current setting, press <Enter>.

6. The system prompts you to configure the next port:

Enter port (1-64):

When you are through configuring ports, press <Enter> without specifying any port. Otherwise, repeat the steps in this section.

Setup Part 3: VLANs

If you chose to skip VLANs configuration back in Part 2, skip to [“Setup Part 4: IP Configuration” on page 57](#).

1. Select the VLAN to configure, or skip VLAN configuration at the prompt:

```
VLAN Config:
Enter VLAN number from 2 to 4094, NULL at end:
```

If you wish to change settings for individual VLANs, enter the number of the VLAN you wish to configure. To skip VLAN configuration, press <Enter> without typing a VLAN number and go to [“Setup Part 4: IP Configuration” on page 57](#).

2. Enter the new VLAN name at the prompt:

```
Current VLAN name: VLAN 2
Enter new VLAN name:
```

Entering a new VLAN name is optional. To use the pending new VLAN name, press <Enter>.

3. Enter the VLAN port numbers:

```
Define Ports in VLAN:
Current VLAN 2: empty
Enter ports one per line, NULL at end:
```

Enter each port, by port number or port alias, and confirm placement of the port into this VLAN. When you are finished adding ports to this VLAN, press <Enter> without specifying any port.

4. Configure Spanning Tree Group membership for the VLAN:

```
Spanning Tree Group membership:
Enter new Spanning Tree Group index [1-]:
```

5. The system prompts you to configure the next VLAN:

```
VLAN Config:
Enter VLAN number from 2 to 4094, NULL at end:
```

Repeat the steps in this section until all VLANs have been configured. When all VLANs have been configured, press <Enter> without specifying any VLAN.

Setup Part 4: IP Configuration

The system prompts for IPv4 parameters.

Although the switch supports both IPv4 and IPv6 networks, the Setup utility permits only IPv4 configuration. For IPv6 configuration, see [“Internet Protocol Version 6” on page 357](#).

IP Interfaces

IP interfaces are used for defining the networks to which the switch belongs.

Up to 126 IP interfaces can be configured on the RackSwitch G8296 (G8296). The IP address assigned to each IP interface provides the switch with an IP presence on your network. No two IP interfaces can be on the same IP network. The interfaces can be used for connecting to the switch for remote configuration, and for routing between subnets and VLANs (if used).

1. Select the IP interface to configure, or skip interface configuration at the prompt:

```
IP Config:
IP interfaces:
Enter interface number: (1-126)
```

If you wish to configure individual IP interfaces, enter the number of the IP interface you wish to configure. To skip IP interface configuration, press <Enter> without typing an interface number and go to [“Default Gateways” on page 59](#).

2. For the specified IP interface, enter the IP address in IPv4 dotted decimal notation:

```
Current IP address:    0.0.0.0
Enter new IP address:
```

To keep the current setting, press <Enter>.

3. At the prompt, enter the IPv4 subnet mask in dotted decimal notation:

```
Current subnet mask:    0.0.0.0
Enter new subnet mask:
```

To keep the current setting, press <Enter>. If configuring VLANs, specify a VLAN for the interface.

This prompt appears if you selected to configure VLANs back in Part 1:

```
Current VLAN:    1
Enter new VLAN [1-4094]:
```

Enter the number for the VLAN to which the interface belongs, or press <Enter> without specifying a VLAN number to accept the current setting.

4. At the prompt, enter **y** to enable the IP interface, or **n** to leave it disabled:

```
Enable IP interface? [y/n]
```

5. The system prompts you to configure another interface:

```
Enter interface number: (1-126)
```

Repeat the steps in this section until all IP interfaces have been configured. When all interfaces have been configured, press <Enter> without specifying any interface number.

Loopback Interfaces

A loopback interface provides an IP address, but is not otherwise associated with a physical port or network entity. Essentially, it is a virtual interface that is perceived as being “always available” for higher-layer protocols to use and advertise to the network, regardless of other connectivity.

Loopback interfaces improve switch access, increase reliability, security, and provide greater flexibility in Layer 3 network designs. They can be used for many different purposes, but are most commonly for management IP addresses, router IDs for various protocols, and persistent peer IDs for neighbor relationships.

In Networking OS 8.2, loopback interfaces have been expanded for use with routing protocols such as OSPF and BGP. Loopback interfaces can also be specified as the source IP address for syslog, SNMP, RADIUS, TACACS+, NTP, and router IDs.

Loopback interfaces must be configured before they can be used in other features. Up to five loopback interfaces are currently supported. They can be configured using the following commands:

```
RS8296(config)# interface loopback <1-5>  
RS8296(config-ip-loopback)# [no] ip address <IPv4 address> <mask> enable  
RS8296(config-ip-loopback)# exit
```

Using Loopback Interfaces for Source IP Addresses

The switch can use loopback interfaces to set the source IP addresses for a variety of protocols. This assists in server security, as the server for each protocol can be configured to accept protocol packets only from the expected loopback address block. It may also make it easier to locate or process protocol information, since packets have the source IP address of the loopback interface, rather than numerous egress interfaces.

Configured loopback interfaces can be applied to the following protocols:

- Syslogs

```
RS8296(config)# logging source-interface loopback <1-5>
```

- SNMP traps

```
RS8296(config)# snmp-server trap-source loopback <1-5>
```

- RADIUS

```
RS8296(config)# ip radius source-interface loopback <1-5>
```

- TACACS+

```
RS8296(config)# ip tacacs source-interface loopback <1-5>
```

- NTP

```
RS8296(config)# ntp source loopback <1-5>
```

Loopback Interface Limitation

- ARP is not supported. Loopback interfaces will ignore ARP requests.
- Loopback interfaces cannot be assigned to a VLAN.

Default Gateways

To set up a default gateway:

1. At the prompt, select an IP default gateway for configuration, or skip default gateway configuration:

```
IP default gateways:
Enter default gateway number: (1-4)
```

Enter the number for the IP default gateway to be configured. To skip default gateway configuration, press <Enter> without typing a gateway number and go to [“IP Routing” on page 59](#).

2. At the prompt, enter the IPv4 address for the selected default gateway:

```
Current IP address:    0.0.0.0
Enter new IP address:
```

Enter the IPv4 address in dotted decimal notation, or press <Enter> without specifying an address to accept the current setting.

3. At the prompt, enter **y** to enable the default gateway, or **n** to leave it disabled:

```
Enable default gateway? [y/n]
```

4. The system prompts you to configure another default gateway:

```
Enter default gateway number: (1-4)
```

Repeat the steps in this section until all default gateways have been configured. When all default gateways have been configured, press <Enter> without specifying any number.

IP Routing

When IP interfaces are configured for the various IP subnets attached to your switch, IP routing between them can be performed entirely within the switch. This eliminates the need to send inter-subnet communication to an external router

device. Routing on more complex networks, where subnets may not have a direct presence on the G8296, can be accomplished through configuring static routes or by letting the switch learn routes dynamically.

This part of the Setup program prompts you to configure the various routing parameters.

At the prompt, enable or disable forwarding for IP Routing:

Enable IP forwarding? [y/n]

Enter **y** to enable IP forwarding. To disable IP forwarding, enter **n**. To keep the current setting, press <Enter>.

Setup Part 5: Final Steps

1. When prompted, decide whether to restart Setup or continue:

Would you like to run from top again? [y/n]

Enter **y** to restart the Setup utility from the beginning, or **n** to continue.

2. When prompted, decide whether you wish to review the configuration changes:

Review the changes made? [y/n]

Enter **y** to review the changes made during this session of the Setup utility. Enter **n** to continue without reviewing the changes. We recommend that you review the changes.

3. Next, decide whether to apply the changes at the prompt:

Apply the changes? [y/n]

Enter **y** to apply the changes, or **n** to continue without applying. Changes are normally applied.

4. At the prompt, decide whether to make the changes permanent:

Save changes to flash? [y/n]

Enter **y** to save the changes to flash. Enter **n** to continue without saving the changes. Changes are normally saved at this point.

5. If you do not apply or save the changes, the system prompts whether to abort them:

Abort all changes? [y/n]

Enter **y** to discard the changes. Enter **n** to return to the "Apply the changes?" prompt.

Note: After initial configuration is complete, it is recommended that you change the default passwords.

Optional Setup for Telnet Support

Note: This step is optional. Perform this procedure only if you are planning on connecting to the G8296 through a remote Telnet connection.

Telnet is enabled by default. To change the setting, use the following command:

```
RS8296(config)# no access telnet
```

Chapter 3. Service Location Protocol

Service Location Protocol (SLP) allows the switch to provide dynamic directory services that help users find servers by attributes rather than by name or address. SLP eliminates the need for a user to know the name of a network host supporting a service. SLP allows the user to bind a service description to the network address of the service.

Service Location Protocol is described in RFC 2608.

SLP defines specialized components called agents that perform tasks and support services as follows:

- User Agent (UA) supports service query functions. It requests service information for user applications. The User Agent retrieves service information from the Service Agent or Directory Agents. A Host On-Demand client is an example of a User Agent.
- Service Agent (SA) provides service registration and service advertisement.
Note: In this release, SA supports UA/DA on Linux with SLPv2 support.
- Directory Agent (DA) collects service information from Service Agents to provide a repository of service information in order to centralize it for efficient access by User Agents. There can only be one Directory Agent present per given host.

The Directory Agent acts as an intermediate tier in the SLP architecture, placed between the User Agents and the Service Agents, so they communicate only with the Directory Agent instead of with each other. This eliminates a large portion of the multicast request or reply traffic on the network, and it protects the Service Agents from being overwhelmed by too many service requests.

Services are described by the configuration of attributes associated with a type of service. A User Agent can select an appropriate service by specifying the attributes that it needs in a service request message. When service replies are returned, they contain a Uniform Resource Locator (URL) pointing to the service desired, and other information, such as server load, needed by the User Agent.

Active DA Discovery

When a Service Agent or User Agent initializes, it can perform Active Directory Agent Discovery using a multicast service request and specifies the special, reserved service type (`service:directory-agent`). Active DA Discovery is achieved through the same mechanism as any other discovery using SLP.

The Directory Agent replies with unicast service replies, which provides the URLs and attributes of the requested service.

SLP Configuration

Use the following ISCLI commands to configure SLP for the switch:

Table 6. *SLP ISCLI Commands*

Command Syntax and Usage
[no] ip slp enable Enables or disables SLP on the switch. Command mode: Global configuration
[no] ip slp active-da-discovery enable Enables or disables Active DA Discovery. Command mode: Global configuration
clear ip slp directory-agents Clears all Directory Agents learned by the switch. Command mode: Global configuration
show ip slp information Displays SLP information. Command mode: All
show ip slp directory-agents Displays Directory Agents learned by the switch. Command mode: All
show ip slp user-agents Displays User Agents information. Command mode: All
show ip slp counter Displays SLP statistics. Command mode: All
clear ip slp counters Clears all Directory Agents learned by the switch. Command mode: Global configuration

Chapter 4. Switch Software Management

The switch software image is the executable code running on the G8296. A version of the image comes pre-installed on the device. As new versions of the image are released, you can upgrade the software running on your switch. To get the latest version of software supported for your G8296, go to the following website:

<http://www.lenovo.com/support/>

To determine the software version currently used on the switch, use the following switch command:

```
RS8296# show boot
```

The typical upgrade process for the software image consists of the following steps:

- Load a new software image and boot image onto an FTP or TFTP server on your network.
- Transfer the new images to your switch.
- Specify the new software image as the one which will be loaded into switch memory the next time a switch reset occurs.
- Reset the switch.

For instructions on the typical upgrade process, see “[Loading New Software to Your Switch](#)” on page 66.



CAUTION:

Although the typical upgrade process is all that is necessary in most cases, upgrading from (or reverting to) some versions of Lenovo Network Operating System requires special steps prior to or after the software installation process. Please be sure to follow all applicable instructions in the release notes document for the specific software release to ensure that your switch continues to operate as expected after installing new software.

Loading New Software to Your Switch

The G8296 can store up to two different switch software images (called `image1` and `image2`) as well as special boot software (called `boot`). When you load new software, you must specify where it is placed: either into `image1`, `image2`, or `boot`.

For example, if your active image is currently loaded into `image1`, you would probably load the new image software into `image2`. This lets you test the new software and reload the original active image (stored in `image1`), if needed.



CAUTION:

When you upgrade the switch software image, always load the new boot image and the new software image before you reset the switch. If you do not load a new boot image, your switch might not boot properly (To recover, see [“Recovering from a Failed Upgrade”](#) on page 70).

To load a new software image to your switch, you will need the following:

- The image and boot software loaded on an FTP or TFTP server on your network.

Note: Be sure to download both the new boot file and the new image file.

- The hostname or IP address of the FTP or TFTP server

Note: The DNS parameters must be configured if specifying hostnames.

- The name of the new software image or boot file

When the software requirements are met, use one of the following procedures to download the new software to your switch. You can use the ISCLI, USB, or the BBI to download and activate new software.

Loading Software via the ISCLI

1. In Privileged EXEC mode, enter the following command:

```
Router# copy {tftp|ftp} {image1|image2|boot-image}
```

2. Enter the hostname or IP address of the FTP or TFTP server.

```
Address or name of remote host: <name or IP address>
```

3. Enter the name of the new software file on the server.

```
Source file name: <filename>
```

The exact form of the name will vary by server. However, the file location is normally relative to the FTP or TFTP directory (for example, `tftpboot`).

4. If required by the FTP or TFTP server, enter the appropriate username and password.

5. The switch will prompt you to confirm your request.

Once confirmed, the software will begin loading into the switch.

6. When loading is complete, use the following commands to enter Global Configuration mode to select which software image (`image1` or `image2`) you want to run in switch memory for the next reboot:

```
Router# configure terminal  
Router(config)# boot image {image1|image2}
```

The system will then verify which image is set to be loaded at the next reset:

```
Next boot will use switch software image1 instead of image2.
```

7. Reboot the switch to run the new software:

```
Router(config)# reload
```

The system prompts you to confirm your request. Once confirmed, the switch will reboot to use the new software.

Loading Software via BBI

You can use the Browser-Based Interface to load software onto the G8296. The software image to load can reside in one of the following locations:

- FTP server
- TFTP server
- Local computer

After you log onto the BBI, perform the following steps to load a software image:

1. Click the Configure context tab in the toolbar.
2. In the Navigation Window, select System > Config/Image Control.

The Switch Image and Configuration Management page appears.

3. If you are loading software from your computer (HTTP client), skip this step and go to the next. Otherwise, if you are loading software from a FTP/TFTP server, enter the server's information in the FTP/TFTP Settings section.
4. In the Image Settings section, select the image version you want to replace (Image for Transfer).
 - If you are loading software from a FTP/TFTP server, enter the file name and click **Get Image**.
 - If you are loading software from your computer, click **Browse**.

In the File Upload Dialog, select the file and click **OK**. Then click **Download via Browser**.

Once the image has loaded, the page refreshes to show the new software.

USB Options

You can insert a USB drive into the USB port on the G8296 and use it to work with switch image and configuration files. You can boot the switch using files located on the USB drive, or copy files to and from the USB drive.

To safely remove the USB drive, first use the following command to un-mount the USB file system:

system usb-eject

Command mode: Global configuration

USB Boot

USB Boot allows you to boot the switch with a software image file, boot file, or configuration file that resides on a USB drive inserted into the USB port. Use the following command to enable or disable USB Boot:

[no] boot usbboot enable

Command mode: Global configuration

When enabled, when the switch is reset/reloaded, it checks the USB port. If a USB drive is inserted into the port, the switch checks the root directory on the USB drive for software and image files. If a valid file is present, the switch loads the file and boots using the file.

Note: The following file types are supported: FAT32, NTFS (read-only), EXT2, and EXT3.

The following list describes the valid file names, and describes the switch behavior when it recognizes them. The file names must be exactly as shown, or the switch will not recognize them.

- **RS8296_Boot.img**
The switch replaces the current boot image with the new image, and boots with the new image.
- **RS8296_OS.img**
The switch boots with the new software image. The existing images are not affected.
- **RS8296_replace1_OS.img**
The switch replaces the current software image1 with the new image, and boots with the new image. **RS8296_replace1_OS.img** takes precedence over **RS8296_OS.img**
- **RS8296_replace2_OS.img** takes precedence over **RS8296_OS.img**
- **RS8296.cfg**
The switch boots with the new configuration file. The existing configuration files (active and backup) are not affected.
- **RS8296_replace.cfg**
The switch replaces the active configuration file with the new file, and boots with the new file. This file takes precedence over any other configuration files that may be present on the USB drive.

If more than one valid file is present, the switch loads all valid files and boots with them. For example, you may simultaneously load a new boot file, image file, and configuration file from the USB drive.

The switch ignores any files that do not match the valid file names or that have the wrong format.

USB Copy

If a USB drive is inserted into the USB port, you can copy files from the switch to the USB drive, or from the USB drive to the switch. USB Copy is available only for software image 1 and the active configuration.

Copy to USB

Use the following command to copy a file from the switch to the USB drive (Privileged EXEC mode):

```
usbcopy tusb <filename> {boot|image1|active|syslog|crashdump}
```

In this example, the active configuration file is copied to a directory on the USB drive:

```
G8296(config)# usbcopy tusb a_folder/myconfig.cfg active
```

Copy from USB

Use the following command to copy a file from the USB drive to the switch:

```
usbcopy fromusb <filename> {boot|image1|active}
```

In this example, the active configuration file is copied from a directory on the USB drive:

```
G8296(config)# usbcopy fromusb a_folder/myconfig.cfg active
```

The new file replaces the current file.

Note: Do not use two consecutive dot characters (..). Do not use a slash character (/) to begin a filename.

The Boot Management Menu

The Boot Management menu allows you to switch the software image, reset the switch to factory defaults, or to recover from a failed software download.

You can interrupt the boot process and enter the Boot Management menu from the serial console port. When the system displays Memory Test, press **<Shift B>**. The Boot Management menu appears.

```
Resetting the System ...
Memory Test .....

Boot Menu mode

FLASH: 16 MB
L2: 512 KB enabled
PCIE1: Root Complex of PCIE, x2, regs @ 0xffe0a000
PCIE1: Bus 00 - 01
MMC: FSL_ESDHC: 0
Net: eTSEC1, eTSEC2 [PRIME]

Boot Management Menu
    I - Change booting image
    C - Change configuration block
    R - Boot in recovery mode (tftp and xmodem download of images to
recover switch)
    Q - Reboot
    E - Exit
Please choose your menu option: I
Current boot image is 1. Enter image to boot: 1 or 2: 2
Booting from image 2
```

The Boot Management menu allows you to perform the following actions:

- To change the booting image, press **I** and follow the screen prompts.
- To change the configuration block, press **C** and follow the screen prompts.
- To reboot the switch, press **Q**. The booting process restarts.
- To boot in recovery mode, press **R** and follow the screen prompts.
- To exit the Boot Management menu, press **E**. The booting process continues.

Recovering from a Failed Upgrade

Use the following procedure to recover from a failed software upgrade.

1. Connect a PC to the serial port of the switch.
2. Open a terminal emulator program that supports XModem Download (such as HyperTerminal or PuTTY) and select the following serial port characteristics:
 - Speed: 9600 bps
 - Data Bits: 8
 - Stop Bits: 1
 - Parity: None
 - Flow Control: None

3. Boot the switch and access the Boot Management menu by pressing **<Shift B>** while the Memory Test is in progress and the dots are being displayed.
4. Select **R** for Recovery mode, then select one of the following options:
 - **T** – Configure networking and tftp download an image
 - **X** – Use xmodem 1K to serial download an image
 - **P** – Physical presence (low security mode)
 - **R** – Reboot
 - **E** – Exit
5. Select **X** for Xmodem download. When you see the following message, change the Serial Port characteristics to 115200 bps:

```
## Switch baudrate to 115200 bps and press ENTER ...
```

6. Press **<Enter>** to set the system into download accept mode. When the readiness meter displays (a series of "C" characters), start XModem on your terminal emulator.

Part 2: Securing the Switch

Chapter 5. Securing Administration

Secure switch management is needed for environments that perform significant management functions across the Internet. Common functions for secured management are described in the following sections:

- [“Secure Shell and Secure Copy” on page 76](#)
- [“End User Access Control” on page 80](#)

Note: SNMP read and write functions are enabled by default. For best security practices, if SNMP is not needed for your network, it is recommended that you disable these functions prior to connecting the switch to the network (see [“Using Simple Network Management Protocol” on page 37](#)).

Secure Shell and Secure Copy

Because using Telnet does not provide a secure connection for managing a G8296, Secure Shell (SSH) and Secure Copy (SCP) features have been included for G8296 management. SSH and SCP use secure tunnels to encrypt and secure messages between a remote administrator and the switch.

SSH is a protocol that enables remote administrators to log securely into the G8296 over a network to execute management commands.

SCP is typically used to copy files securely from one machine to another. SCP uses SSH for encryption of data on the network. On a G8296, SCP is used to download and upload the switch configuration via secure channels.

Although SSH and SCP are disabled by default, enabling and using these features provides the following benefits:

- Identifying the administrator using Name/Password
- Authentication of remote administrators
- Authorization of remote administrators
- Determining the permitted actions and customizing service for individual administrators
- Encryption of management messages
- Encrypting messages between the remote administrator and switch
- Secure copy support

Lenovo Network Operating System implements the SSH version 2.0 standard and is confirmed to work with SSH version 2.0-compliant clients such as the following:

- OpenSSH_5.4p1 for Linux
- Secure CRT Version 5.0.2 (build 1021)
- Putty SSH release 0.60

Configuring SSH/SCP Features on the Switch

SSH and SCP features are disabled by default. To change the SSH/SCP settings, using the following procedures.

Note: To use SCP, you must first enable SSH.

To Enable or Disable the SSH Feature

Connect to the console port and enter the following command:

```
RS8296(config)# [no] ssh enable
```

To Enable or Disable SCP Apply and Save

Enter the following command from the switch CLI to enable the SCP `putcfg_apply` and `putcfg_apply_save` commands:

```
RS8296(config)# [no] ssh scp-enable
```

Configuring the SCP Administrator Password

To configure the SCP-only administrator password, enter the following command (the default password is admin):

```
RS8296(config)# [no] ssh scp-password
Changing SCP-only Administrator password; validation required...
Enter current administrator password: <password>
Enter new SCP-only administrator password: <new password>
Re-enter new SCP-only administrator password: <new password>
New SCP-only administrator password accepted.
```

Using SSH and SCP Client Commands

This section shows the format for using some client commands. The following examples use 205.178.15.157 as the IP address of a sample switch.

To Log In to the Switch

Syntax:

```
>> ssh [-4|-6] <switch IP address>
-or-
>> ssh [-4|-6] <login name>@<switch IP address>
```

Note: The -4 option (the default) specifies that an IPv4 switch address will be used. The -6 option specifies IPv6.

Example:

```
>> ssh scpadmin@205.178.15.157
```

To Copy the Switch Configuration File to the SCP Host

Syntax:

```
>> scp [-4|-6] <username>@<switch IP address>:getcfg <local filename>
```

Example:

```
>> scp scpadmin@205.178.15.157:getcfg ad4.cfg
```

To Load a Switch Configuration File from the SCP Host

Syntax:

```
>> scp [-4|-6] <local filename> <username>@<switch IP address>:putcfg
```

Example:

```
>> scp ad4.cfg scpadmin@205.178.15.157:putcfg
```

To Apply and Save the Configuration

When loading a configuration file to the switch, the `apply` and `save` commands are still required for the configuration commands to take effect. The `apply` and `save` commands may be entered manually on the switch, or by using SCP commands.

Syntax:

```
>> scp [-4|-6] <local filename> <username>@<switch IP address>:putcfg_apply
>> scp [-4|-6] <local filename> <username>@<switch IP address>:putcfg_apply_save
```

Example:

```
>> scp ad4.cfg scpadmin@205.178.15.157:putcfg_apply
>> scp ad4.cfg scpadmin@205.178.15.157:putcfg_apply_save
```

- The CLI `diff` command is automatically executed at the end of `putcfg` to notify the remote client of the difference between the new and the current configurations.
- `putcfg_apply` runs the `apply` command after the `putcfg` is done.
- `putcfg_apply_save` saves the new configuration to the flash after `putcfg_apply` is done.
- The `putcfg_apply` and `putcfg_apply_save` commands are provided because extra `apply` and `save` commands are usually required after a `putcfg`; however, an SCP session is not in an interactive mode.

To Copy the Switch Image and Boot Files to the SCP Host

Syntax:

```
>> scp [-4|-6] <username>@<switch IP address>:getimg1 <local filename>
>> scp [-4|-6] <username>@<switch IP address>:getimg2 <local filename>
>> scp [-4|-6] <username>@<switch IP address>:getboot <local filename>
```

Example:

```
>> scp scpadmin@205.178.15.157:getimg1 6.1.0_os.img
```

To Load Switch Configuration Files from the SCP Host

Syntax:

```
>> scp [-4|-6] <local filename> <username>@<switch IP address>:putimg1
>> scp [-4|-6] <local filename> <username>@<switch IP address>:putimg2
>> scp [-4|-6] <local filename> <username>@<switch IP address>:putboot
```

Example:

```
>> scp 6.1.0_os.img scpadmin@205.178.15.157:putimg1
```

SSH and SCP Encryption of Management Messages

The following encryption and authentication methods are supported for SSH and SCP:

- Server Host Authentication: Client RSA authenticates the switch at the beginning of every connection
- Key Exchange: RSA
- Encryption: 3DES-CBC
- User Authentication: Local password authentication, RADIUS

Generating RSA Host Key for SSH Access

To support the SSH host feature, an RSA host key is required. The host key is 2048 bits and is used to identify the G8296.

To configure RSA host key, first connect to the G8296 through the console port (commands are not available via external Telnet connection), and enter the following command to generate it manually.

```
RS8296(config)# ssh generate-host-key
```

When the switch reboots, it will retrieve the host key from the FLASH memory.

Note: The switch will perform only one session of key/cipher generation at a time. Thus, an SSH/SCP client will not be able to log in if the switch is performing key generation at that time. Also, key generation will fail if an SSH/SCP client is logging in at that time.

SSH/SCP Integration with Radius Authentication

SSH/SCP is integrated with RADIUS authentication. After the RADIUS server is enabled on the switch, all subsequent SSH authentication requests will be redirected to the specified RADIUS servers for authentication. The redirection is transparent to the SSH clients.

SSH/SCP Integration with TACACS+ Authentication

SSH/SCP is integrated with TACACS+ authentication. After the TACACS+ server is enabled on the switch, all subsequent SSH authentication requests will be redirected to the specified TACACS+ servers for authentication. The redirection is transparent to the SSH clients.

End User Access Control

Networking OS allows an administrator to define end user accounts that permit end users to perform operation tasks via the switch CLI commands. Once end user accounts are configured and enabled, the switch requires username/password authentication.

For example, an administrator can assign a user, who can then log into the switch and perform operational commands (effective only until the next switch reboot).

Considerations for Configuring End User Accounts

Note the following considerations when you configure end user accounts:

- A maximum of 20 user IDs are supported on the switch.
- N/OS supports end user support for console, Telnet, BBI, and SSHv2 access to the switch.
- If RADIUS authentication is used, the user password on the Radius server will override the user password on the G8296. Also note that the password change command only modifies only the user password on the switch and has no effect on the user password on the Radius server. Radius authentication and user password cannot be used concurrently to access the switch.
- Passwords for end users can be up to 128 characters in length for TACACS, RADIUS, Telnet, SSH, Console, and Web access.

Strong Passwords

The administrator can require use of Strong Passwords for users to access the G8296. Strong Passwords enhance security because they make password guessing more difficult.

The following rules apply when Strong Passwords are enabled:

- Minimum length: 8 characters; maximum length: 64 characters.
- Must contain at least one uppercase alphabet.
- Must contain at least one lowercase alphabet.
- Must contain at least one number.
- Must contain at least one special character.
Supported special characters: ! " # % & ' () ; < = > ? [\] * + , - . / : ^ _ { | } ~
- Cannot be same as the username.
- No consecutive four characters can be the same as in the old password.

When strong password is enabled, users can still access the switch using the old password but will be advised to change to a strong password while attempting to log in.

Strong password requirement can be enabled using the following command:

```
RS8296(config)# access user strong-password enable
```


The administrator can choose the number of days allowed before each password expires. When a strong password expires, the user is allowed to log in one last time to change the password. A warning provides advance notice for users to change the password.

User Access Control

The end-user access control commands allow you to configure end-user accounts.

Setting up User IDs

Up to 20 user IDs can be configured. Use the following commands to define any user name and set the user password at the resulting prompts:

```
RS8296(config)# access user 1 name <1-8 characters>
RS8296(config)# access user 1 password

Changing user1 password; validation required:
Enter current admin password: <current administrator password>
Enter new user1 password: <new user password>
Re-enter new user1 password: <new user password>
New user1 password accepted.
```

Defining a User's Access Level

The end user is by default assigned to the user access level (also known as class of service, or COS). COS for all user accounts have global access to all resources except for User COS, which has access to view only resources that the user owns. For more information, see [Table 7 on page 86](#).

To change the user's level, select one of the following options:

```
RS8296(config)# access user 1 level {user|operator|administrator}
```

Validating a User's Configuration

```
RS8296# show access user uid 1
```

Enabling or Disabling a User

An end user account must be enabled before the switch recognizes and permits login under the account. Once enabled, the switch requires any user to enter both username and password.

```
RS8296(config)# [no] access user 1 enable
```

Locking Accounts

To protect the switch from unauthorized access, the account lockout feature can be enabled. By default, account lockout is disabled. To enable this feature, ensure the strong password feature is enabled (See ["Strong Passwords" on page 80](#)). Then use the following command:

```
RS8296(config)# access user strong-password lockout
```

After multiple failed login attempts, the switch locks the user account if lockout has been enabled on the switch.

Re-enabling Locked Accounts

The administrator can re-enable a locked account by reloading the switch or by using the following command:

```
RS8296(config)# access user strong-password clear local user lockout
username <user name>
```

However, the above command cannot be used to re-enable an account disabled by the administrator.

To re-enable all locked accounts, use the following command:

```
RS8296(config)# access user strong-password clear local user lockout all
```

Listing Current Users

The following command displays defined user accounts and whether or not each user is currently logged into the switch.

```
RS8296# show access user

Usernames:
  user   - Enabled - offline
  oper   - Disabled - offline
  admin  - Always Enabled - online 1 session

Current User ID table:
1: name jane   , ena, cos user   , password valid, online 1 session
2: name john   , ena, cos user   , password valid, online 2 sessions
```

Logging into an End User Account

Once an end user account is configured and enabled, the user can login to the switch using the username/password combination. The level of switch access is determined by the COS established for the end user account.

Password Fix-Up Mode

Password Fix-Up Mode enables admin user account recovery if administrator access is lost. You must connect to the switch over the serial console and log in using the password "forgetME!". This enables the admin account if disabled and a new administrator password can be entered.

To disable the Password Fix-Up functionality, use the following command:

```
RS8296(config)# no access user password-recovery
```

Chapter 6. Authentication & Authorization Protocols

Secure switch management is needed for environments that perform significant management functions across the Internet. The following are some of the functions for secured IPv4 management and device access:

- [“RADIUS Authentication and Authorization” on page 84](#)
- [“TACACS+ Authentication” on page 88](#)
- [“LDAP Authentication and Authorization” on page 92](#)

Note: Lenovo Network Operating System 8.2 does not support IPv6 for RADIUS, TACACS+, or LDAP.

RADIUS Authentication and Authorization

Networking OS supports the RADIUS (Remote Authentication Dial-in User Service) method to authenticate and authorize remote administrators for managing the switch. This method is based on a client/server model. The Remote Access Server (RAS)—the switch—is a client to the back-end database server. A remote user (the remote administrator) interacts only with the RAS, not the back-end server and database.

RADIUS authentication consists of the following components:

- A protocol with a frame format that utilizes UDP over IP (based on RFC 2138 and 2866)
- A centralized server that stores all the user authorization information
- A client: in this case, the switch

The G8296—acting as the RADIUS client—communicates to the RADIUS server to authenticate and authorize a remote administrator using the protocol definitions specified in RFC 2138 and 2866. Transactions between the client and the RADIUS server are authenticated using a shared key that is not sent over the network. In addition, the remote administrator passwords are sent encrypted between the RADIUS client (the switch) and the back-end RADIUS server.

How RADIUS Authentication Works

The RADIUS authentication process follows these steps:

1. A remote administrator connects to the switch and provides a user name and password.
2. Using Authentication/Authorization protocol, the switch sends request to authentication server.
3. The authentication server checks the request against the user ID database.
4. Using RADIUS protocol, the authentication server instructs the switch to grant or deny administrative access.

Configuring RADIUS on the Switch

Use the following procedure to configure Radius authentication on your switch.

1. Configure the IPv4 addresses of the Primary and Secondary RADIUS servers, and enable RADIUS authentication.

```
RS8296(config)# radius-server primary-host 10.10.1.1
RS8296(config)# radius-server secondary-host 10.10.1.2
RS8296(config)# radius-server enable
```

Note: You can use a configured loopback address as the source address so the RADIUS server accepts requests only from the expected loopback address block. Use the following command to specify the loopback interface:

```
RS8296(config)# ip radius source-interface loopback <1-5>
```

2. Configure the RADIUS secret.

```
RS8296(config)# radius-server primary-host 10.10.1.1 key  
<1-32 character secret>  
RS8296(config)# radius-server secondary-host 10.10.1.2 key  
<1-32 character secret>
```

3. If desired, you may change the default UDP port number used to listen to RADIUS.
The well-known port for RADIUS is 1812.

```
RS8296(config)# radius-server port <UDP port number>
```

4. Configure the number retry attempts for contacting the RADIUS server, and the timeout period.

```
RS8296(config)# radius-server retransmit 3  
RS8296(config)# radius-server timeout 5
```

RADIUS Authentication Features in Networking OS

N/OS supports the following RADIUS authentication features:

- Supports RADIUS client on the switch, based on the protocol definitions in RFC 2138 and RFC 2866.
- Allows RADIUS secret password up to 32 bytes and less than 16 octets.
- Supports *secondary authentication server* so that when the primary authentication server is unreachable, the switch can send client authentication requests to the secondary authentication server. Use the following command to show the currently active RADIUS authentication server:

```
RS8296# show radius-server
```

- Supports user-configurable RADIUS server retry and time-out values:
 - Time-out value = 1-10 seconds
 - Retries = 1-3

The switch will time out if it does not receive a response from the RADIUS server in 1-3 retries. The switch will also automatically retry connecting to the RADIUS server before it declares the server down.

- Supports user-configurable RADIUS application port. The default is 1645/UDP-based on RFC 2138. Port 1812 is also supported.
- Allows network administrator to define privileges for one or more specific users to access the switch at the RADIUS user database.

Switch User Accounts

The user accounts listed in [Table 7](#) can be defined in the RADIUS server dictionary file.

Table 7. *User Access Levels*

User Account	Description and Tasks Performed	Password
User	The User has no direct responsibility for switch management. They can view all switch status information and statistics but cannot make any configuration changes to the switch.	user
Operator	The Operator manages all functions of the switch. The Operator can reset ports, except the management port.	oper
Administrator	The super-user Administrator has complete access to all commands, information, and configuration commands on the switch, including the ability to change both the user and administrator passwords.	admin

RADIUS Attributes for Networking OS User Privileges

When the user logs in, the switch authenticates his/her level of access by sending the RADIUS access request, that is, the client authentication request, to the RADIUS authentication server.

If the remote user is successfully authenticated by the authentication server, the switch will verify the *privileges* of the remote user and authorize the appropriate access. The administrator has two options: to allow *backdoor* access via Telnet, SSH, HTTP, or HTTPS; to allow *secure backdoor* access via Telnet, SSH, or BBI. Backdoor and secure backdoor provides access to the switch when the RADIUS servers cannot be reached.

The default G8296 setting for backdoor and secure backdoor access is **disabled**. Backdoor and secure backdoor access is always enabled on the console port.

Irrespective of backdoor/secure backdoor being enabled or not, you can always access the switch via the console port by using `noradius` as radius username. You can then enter the username and password configured on the switch. If you are trying to connect via SSH/Telnet/HTTP/HTTPS (not console port), there are two possibilities:

- Backdoor is enabled: The switch acts like it is connecting via console.
- Secure backdoor is enabled: You must enter the username: `noradius`. The switch checks if RADIUS server is reachable. If it is reachable, then you must authenticate via remote authentication server. Only if RADIUS server is not reachable, you will be prompted for local user/password to be authenticated against these local credentials.

All user privileges, other than those assigned to the Administrator, have to be defined in the RADIUS dictionary. RADIUS attribute 6 which is built into all RADIUS servers defines the administrator. The file name of the dictionary is RADIUS vendor-dependent. The following RADIUS attributes are defined for G8296 user privileges levels:

Table 8. *Networking OS-proprietary Attributes for RADIUS*

User Name/Access	User-Service-Type	Value
User	<i>Vendor-supplied</i>	255
Operator	<i>Vendor-supplied</i>	252
Admin	<i>Vendor-supplied</i>	6

TACACS+ Authentication

N/OS supports authentication and authorization with networks using the Cisco Systems TACACS+ protocol. The G8296 functions as the Network Access Server (NAS) by interacting with the remote client and initiating authentication and authorization sessions with the TACACS+ access server. The remote user is defined as someone requiring management access to the G8296 through a data port.

TACACS+ offers the following advantages over RADIUS:

- TACACS+ uses TCP-based connection-oriented transport; whereas RADIUS is UDP-based. TCP offers a connection-oriented transport, while UDP offers best-effort delivery. RADIUS requires additional programmable variables such as re-transmit attempts and time-outs to compensate for best-effort transport, but it lacks the level of built-in support that a TCP transport offers.
- TACACS+ offers full packet encryption whereas RADIUS offers password-only encryption in authentication requests.
- TACACS+ separates authentication, authorization and accounting.

How TACACS+ Authentication Works

TACACS+ works much in the same way as RADIUS authentication as described on [page 84](#).

1. Remote administrator connects to the switch and provides user name and password.
2. Using Authentication/Authorization protocol, the switch sends request to authentication server.
3. Authentication server checks the request against the user ID database.
4. Using TACACS+ protocol, the authentication server instructs the switch to grant or deny administrative access.

During a session, if additional authorization checking is needed, the switch checks with a TACACS+ server to determine if the user is granted permission to use a particular command.

TACACS+ Authentication Features in Networking OS

Authentication is the action of determining the identity of a user, and is generally done when the user first attempts to log in to a device or gain access to its services. N/OS supports ASCII inbound login to the device. PAP, CHAP and ARAP login methods, TACACS+ change password requests, and one-time password authentication are not supported.

Authorization

Authorization is the action of determining a user's privileges on the device, and usually takes place after authentication.

The default mapping between TACACS+ authorization levels and N/OS management access levels is shown in [Table 9](#). The authorization levels must be defined on the TACACS+ server.

Table 9. *Default TACACS+ Authorization Levels*

N/OS User Access Level	TACACS+ level
user	0
oper	3
admin	6

Alternate mapping between TACACS+ authorization levels and N/OS management access levels is shown in [Table 10](#). Use the following command to set the alternate TACACS+ authorization levels.

```
RS8296(config)# tacacs-server privilege-mapping
```

Table 10. *Alternate TACACS+ Authorization Levels*

N/OS User Access Level	TACACS+ level
user	0 - 1
oper	6 - 8
admin	14 - 15

If the remote user is successfully authenticated by the authentication server, the switch verifies the *privileges* of the remote user and authorizes the appropriate access. The administrator has an option to allow *secure backdoor* access via Telnet/SSH. Secure backdoor provides switch access when the TACACS+ servers cannot be reached. You always can access the switch via the console port, by using `notacacs` and the administrator password, whether secure backdoor is enabled or not.

Note: To obtain the TACACS+ backdoor password for your G8296, contact Technical Support.

Accounting

Accounting is the action of recording a user's activities on the device for the purposes of billing and/or security. It follows the authentication and authorization actions. If the authentication and authorization is not performed via TACACS+, there are no TACACS+ accounting messages sent out.

You can use TACACS+ to record and track software login access, configuration changes, and interactive commands.

The G8296 supports the following TACACS+ accounting attributes:

- protocol (console/Telnet/SSH/HTTP/HTTPS)
- start_time
- stop_time
- elapsed_time

- disc_cause

Note: When using the Browser-Based Interface, the TACACS+ Accounting Stop records are sent only if the **Logout** button on the browser is clicked.

Command Authorization and Logging

When TACACS+ Command Authorization is enabled, N/OS configuration commands are sent to the TACACS+ server for authorization. Use the following command to enable TACACS+ Command Authorization:

```
RS8296(config)# tacacs-server command-authorization
```

When TACACS+ Command Logging is enabled, N/OS configuration commands are logged on the TACACS+ server. Use the following command to enable TACACS+ Command Logging:

```
RS8296(config)# tacacs-server command-logging
```

The following examples illustrate the format of N/OS commands sent to the TACACS+ server:

```
authorization request, cmd=shell, cmd-arg=interface ip
accounting request, cmd=shell, cmd-arg=interface ip
authorization request, cmd=shell, cmd-arg=enable
accounting request, cmd=shell, cmd-arg=enable
```

Configuring TACACS+ Authentication on the Switch

1. Configure the IPv4 addresses of the Primary and Secondary TACACS+ servers, and enable TACACS authentication.

```
RS8296(config)# tacacs-server primary-host 10.10.1.1
RS8296(config)# tacacs-server primary-host mgt-port
RS8296(config)# tacacs-server secondary-host 10.10.1.2
RS8296(config)# tacacs-server secondary-host data-port
RS8296(config)# tacacs-server enable
```

Note: You can use a configured loopback address as the source address so the TACACS+ server accepts requests only from the expected loopback address block. Use the following command to specify the loopback interface:

```
RS8296(config)# ip tacacs source-interface loopback <1-5>
```

2. Configure the TACACS+ secret and second secret.

```
RS8296(config)# tacacs-server primary-host 10.10.1.1 key <1-32 character secret>
RS8296(config)# tacacs-server secondary-host 10.10.1.2 key
<1-32 character secret>
```

3. If desired, you may change the default TCP port number used to listen to TACACS+.

The well-known port for TACACS+ is 49.

```
RS8296(config)# tacacs-server port <TCP port number>
```

4. Configure the number of retry attempts, and the timeout period.

```
RS8296(config)# tacacs-server retransmit 3  
RS8296(config)# tacacs-server timeout 5
```

LDAP Authentication and Authorization

N/OS supports the LDAP (Lightweight Directory Access Protocol) method to authenticate and authorize remote administrators to manage the switch. LDAP is based on a client/server model. The switch acts as a client to the LDAP server. A remote user (the remote administrator) interacts only with the switch, not the back-end server and database.

LDAP authentication consists of the following components:

- A protocol with a frame format that utilizes TCP over IP
- A centralized server that stores all the user authorization information
- A client: in this case, the switch

Each entry in the LDAP server is referenced by its Distinguished Name (DN). The DN consists of the user-account name concatenated with the LDAP domain name. If the user-account name is John, the following is an example DN:

```
uid=John,ou=people,dc=domain,dc=com
```

Configuring the LDAP Server

G8296 user groups and user accounts must reside within the same domain. On the LDAP server, configure the domain to include G8296 user groups and user accounts, as follows:

- User Accounts:
Use the *uid* attribute to define each individual user account.
- User Groups:
Use the *members* attribute in the *groupOfNames* object class to create the user groups. The first word of the common name for each user group must be equal to the user group names defined in the G8296, as follows:
 - o admin
 - o oper
 - o user

Configuring LDAP Authentication on the Switch

1. Turn LDAP authentication on, then configure the IPv4 addresses of the Primary and Secondary LDAP servers.

```
>> # ldap-server enable
>> # ldap-server primary-host 10.10.1.1 mgt-port
>> # ldap-server secondary-host 10.10.1.2 data-port
```

2. Configure the domain name.

```
>> # ldap-server domain <ou=people,dc=my-domain,dc=com>
```

3. You may change the default TCP port number used to listen to LDAP (optional).

The well-known port for LDAP is 389.

```
>> # ldap-server port <1-65000>
```

4. Configure the number of retry attempts for contacting the LDAP server, and the timeout period.

```
>> # ldap-server retransmit 3  
>> # ldap-server timeout 10
```

Chapter 7. 802.1X Port-Based Network Access Control

Port-Based Network Access control provides a means of authenticating and authorizing devices attached to a LAN port that has point-to-point connection characteristics. It prevents access to ports that fail authentication and authorization. This feature provides security to ports of the RackSwitch G8296 (G8296) that connect to blade servers.

The following topics are discussed in this section:

- [“Extensible Authentication Protocol over LAN” on page 96](#)
- [“EAPoL Authentication Process” on page 97](#)
- [“EAPoL Port States” on page 99](#)
- [“Guest VLAN” on page 99](#)
- [“Supported RADIUS Attributes” on page 100](#)
- [“EAPoL Configuration Guidelines” on page 102](#)

Extensible Authentication Protocol over LAN

Lenovo Network Operating System can provide user-level security for its ports using the IEEE 802.1X protocol, which is a more secure alternative to other methods of port-based network access control. Any device attached to an 802.1X-enabled port that fails authentication is prevented access to the network and denied services offered through that port.

The 802.1X standard describes port-based network access control using Extensible Authentication Protocol over LAN (EAPoL). EAPoL provides a means of authenticating and authorizing devices attached to a LAN port that has point-to-point connection characteristics and of preventing access to that port in cases of authentication and authorization failures.

EAPoL is a client-server protocol that has the following components:

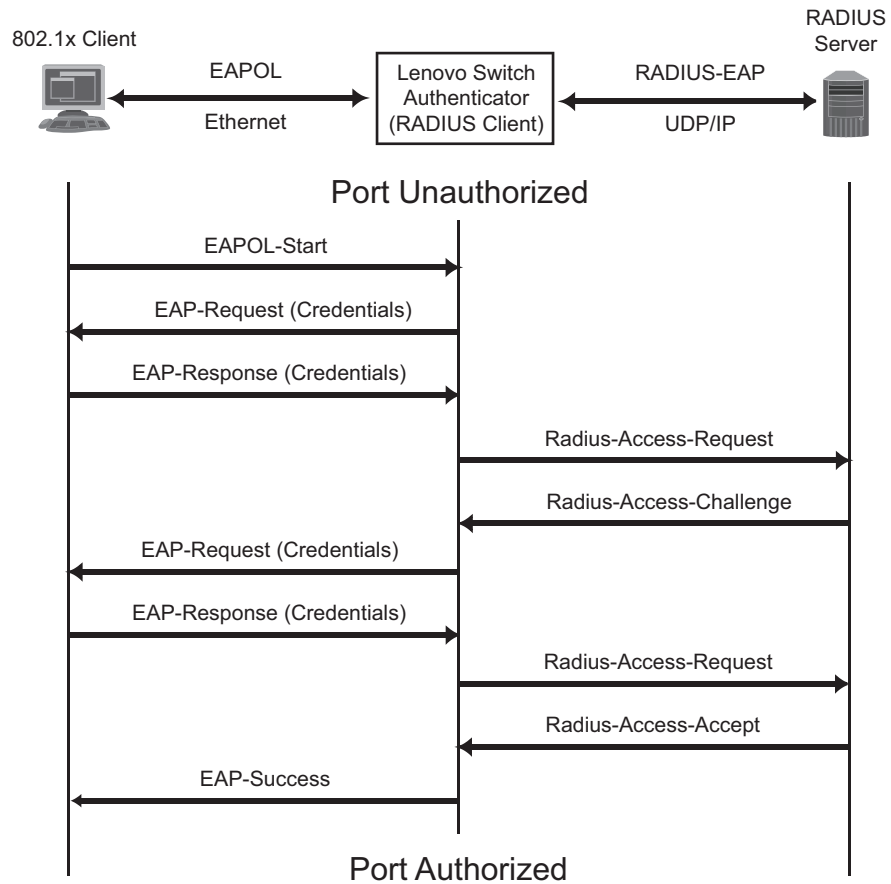
- **Supplicant or Client**
The Supplicant is a device that requests network access and provides the required credentials (user name and password) to the Authenticator and the Authentication Server.
- **Authenticator**
The Authenticator enforces authentication and controls access to the network. The Authenticator grants network access based on the information provided by the Supplicant and the response from the Authentication Server. The Authenticator acts as an intermediary between the Supplicant and the Authentication Server: requesting identity information from the client, forwarding that information to the Authentication Server for validation, relaying the server's responses to the client, and authorizing network access based on the results of the authentication exchange. The G8296 acts as an Authenticator.
- **Authentication Server**
The Authentication Server validates the credentials provided by the Supplicant to determine if the Authenticator ought to grant access to the network. The Authentication Server may be co-located with the Authenticator. The G8296 relies on external RADIUS servers for authentication.

Upon a successful authentication of the client by the server, the 802.1X-controlled port transitions from unauthorized to authorized state, and the client is allowed full access to services through the port. When the client sends an EAP-Logoff message to the authenticator, the port will transition from authorized to unauthorized state.

EAPoL Authentication Process

The clients and authenticators communicate using Extensible Authentication Protocol (EAP), which was originally designed to run over PPP, and for which the IEEE 802.1X Standard has defined an encapsulation method over Ethernet frames, called EAP over LAN (EAPOL). [Figure 1](#) shows a typical message exchange initiated by the client.

Figure 1. Authenticating a Port Using EAPoL



EAPoL Message Exchange

During authentication, EAPoL messages are exchanged between the client and the G8296 authenticator, while RADIUS-EAP messages are exchanged between the G8296 authenticator and the RADIUS server.

Authentication is initiated by one of the following methods:

- The G8296 authenticator sends an EAP-Request/Identity packet to the client
- The client sends an EAPoL-Start frame to the G8296 authenticator, which responds with an EAP-Request/Identity frame.

The client confirms its identity by sending an EAP-Response/Identity frame to the G8296 authenticator, which forwards the frame encapsulated in a RADIUS packet to the server.

The RADIUS authentication server chooses an EAP-supported authentication algorithm to verify the client's identity, and sends an EAP-Request packet to the client via the G8296 authenticator. The client then replies to the RADIUS server with an EAP-Response containing its credentials.

Upon a successful authentication of the client by the server, the 802.1X-controlled port transitions from unauthorized to authorized state, and the client is allowed full access to services through the controlled port. When the client later sends an EAPoL-Logoff message to the G8296 authenticator, the port transitions from authorized to unauthorized state.

If a client that does not support 802.1X connects to an 802.1X-controlled port, the G8296 authenticator requests the client's identity when it detects a change in the operational state of the port. The client does not respond to the request, and the port remains in the unauthorized state.

Note: When an 802.1X-enabled client connects to a port that is not 802.1X-controlled, the client initiates the authentication process by sending an EAPoL-Start frame. When no response is received, the client retransmits the request for a fixed number of times. If no response is received, the client assumes the port is in authorized state, and begins sending frames, even if the port is unauthorized.

EAPoL Port States

The state of the port determines whether the client is granted access to the network, as follows:

- **Unauthorized**
While in this state the port discards all ingress and egress traffic except EAP packets.
- **Authorized**
When the client is successfully authenticated, the port transitions to the authorized state allowing all traffic to and from the client to flow normally.
- **Force Unauthorized**
You can configure this state that denies all access to the port.
- **Force Authorized**
You can configure this state that allows full access to the port.

Use the 802.1X global configuration commands (`dot1x`) to configure 802.1X authentication for all ports in the switch. Use the 802.1X port commands to configure a single port.

Guest VLAN

The guest VLAN provides limited access to unauthenticated ports. The guest VLAN can be configured using the following commands:

```
RS8296(config)# dot1x guest-vlan ?
```

Client ports that have not received an EAPoL response are placed into the Guest VLAN, if one is configured on the switch. Once the port is authenticated, it is moved from the Guest VLAN to its configured VLAN.

When Guest VLAN enabled, the following considerations apply while a port is in the unauthenticated state:

- The port is placed in the guest VLAN.
- The Port VLAN ID (PVID) is changed to the Guest VLAN ID.
- Port tagging is disabled on the port.

Supported RADIUS Attributes

The 802.1X Authenticator relies on external RADIUS servers for authentication with EAP. Table 11 lists the RADIUS attributes that are supported as part of RADIUS-EAP authentication based on the guidelines specified in Annex D of the 802.1X standard and RFC 3580.

Table 11. Support for RADIUS Attributes

#	Attribute	Attribute Value	A-R	A-A	A-C	A-R
1	User-Name	The value of the Type-Data field from the supplicant's EAP-Response/ Identity message. If the Identity is unknown (for example, Type-Data field is zero bytes in length), this attribute will have the same value as the Calling-Station-Id.	1	0-1	0	0
4	NAS-IP-Address	IPv4 address of the authenticator used for Radius communication.	1	0	0	0
5	NAS-Port	Port number of the authenticator port to which the supplicant is attached.	1	0	0	0
24	State	Server-specific value. This is sent unmodified back to the server in an Access-Request that is in response to an Access-Challenge.	0-1	0-1	0-1	0
30	Called-Station-ID	The MAC address of the authenticator encoded as an ASCII string in canonical format, such as 000D5622E3 9F.	1	0	0	0
31	Calling-Station-ID	The MAC address of the supplicant encoded as an ASCII string in canonical format, such as 00034B436206.	1	0	0	0
64	Tunnel-Type	Only VLAN (type 13) is currently supported (for 802.1X RADIUS VLAN assignment). The attribute must be untagged (the Tag field must be 0).	0	0-1	0	0
65	Tunnel-Medium-Type	Only 802 (type 6) is currently supported (for 802.1X RADIUS VLAN assignment). The attribute must be untagged (the Tag field must be 0).	0	0-1	0	0

Table 11. Support for RADIUS Attributes (continued)

#	Attribute	Attribute Value	A-R	A-A	A-C	A-R
81	Tunnel-Private-Group-ID	VLAN ID (1-4094). When 802.1X RADIUS VLAN assignment is enabled on a port, if the RADIUS server includes the tunnel attributes defined in RFC 2868 in the Access-Accept packet, the switch will automatically place the authenticated port in the specified VLAN. Reserved VLANs (such as for management) may not be specified. The attribute must be untagged (the Tag field must be 0).	0	0-1	0	0
79	EAP-Message	Encapsulated EAP packets from the supplicant to the authentication server (Radius) and vice-versa. The authenticator relays the decoded packet to both devices.	1+	1+	1+	1+
80	Message-Authenticator	Always present whenever an EAP-Message attribute is also included. Used to integrity-protect a packet.	1	1	1	1
87	NAS-Port-ID	Name assigned to the authenticator port, e.g. Server1_Port3	1	0	0	0

Legend: RADIUS Packet Types: A-R (Access-Request), A-A (Access-Accept), A-C (Access-Challenge), A-R (Access-Reject)

RADIUS Attribute Support:

- 0 This attribute MUST NOT be present in a packet.
- 0+ Zero or more instances of this attribute MAY be present in a packet.
- 0-1 Zero or one instance of this attribute MAY be present in a packet.
- 1 Exactly one instance of this attribute MUST be present in a packet.
- 1+ One or more of these attributes MUST be present.

EAPoL Configuration Guidelines

When configuring EAPoL, consider the following guidelines:

- The 802.1X port-based authentication is currently supported only in point-to-point configurations, that is, with a single supplicant connected to an 802.1X-enabled switch port.
- When 802.1X is enabled, a port has to be in the authorized state before any other Layer 2 feature can be operationally enabled. For example, the STG state of a port is operationally disabled while the port is in the unauthorized state.
- The 802.1X supplicant capability is not supported. Therefore, none of its ports can successfully connect to an 802.1X-enabled port of another device, such as another switch, that acts as an authenticator, unless access control on the remote port is disabled or is configured in forced-authorized mode. For example, if a G8296 is connected to another G8296, and if 802.1X is enabled on both switches, the two connected ports must be configured in force-authorized mode.
- Unsupported 802.1X attributes include Service-Type, Session-Timeout, and Termination-Action.
- RADIUS accounting service for 802.1X-authenticated devices or users is not currently supported.
- Configuration changes performed using SNMP and the standard 802.1X MIB will take effect immediately.

Chapter 8. Access Control Lists

Access Control Lists (ACLs) are filters that permit or deny traffic for security purposes. They can also be used with QoS to classify and segment traffic to provide different levels of service to different traffic types. Each filter defines the conditions that must match for inclusion in the filter, and also the actions that are performed when a match is made.

Lenovo Network Operating System 8.2 supports the following ACLs:

- IPv4 ACLs

Up to 256 ACLs are supported for networks that use IPv4 addressing. IPv4 ACLs are configured using the following ISCLI command path:

```
RS8296(config)# access-control list <IPv4 ACL number> ?
```

- IPv6 ACLs

Up to 128 ACLs are supported for networks that use IPv6 addressing. IPv6 ACLs are configured using the following ISCLI command path:

```
RS8296(config)# access-control list6 <IPv6 ACL number> ?
```

- VLAN Maps (VMaps)

Up to VLAN Maps are supported for attaching filters to VLANs rather than ports. See [“VLAN Maps” on page 112](#) for details.

- Management ACLs (MACLs)

Up to 256 MACLs are supported for filtering traffic toward CPU. MACLs are configured using the following ISCLI command path:

```
RS8296(config)# access-control macl <MACL number> ?
```

Summary of Packet Classifiers

ACLs allow you to classify packets according to a variety of content in the packet header (such as the source address, destination address, source port number, destination port number, and others). Once classified, packet flows can be identified for more processing.

IPv4 ACLs, IPv6 ACLs, and VMaps allow you to classify packets based on the following packet attributes:

- Ethernet header options (for IPv4 ACLs and VMaps only)
 - Source MAC address
 - Destination MAC address
 - VLAN number and mask
 - Ethernet type (ARP, IP, IPv6, MPLS, RARP, etc.)
 - Ethernet Priority (the IEEE 802.1p Priority)
- IPv4 header options (for IPv4 ACLs and VMaps only)
 - Source IPv4 address and subnet mask
 - Destination IPv4 address and subnet mask
 - Type of Service value
 - IP protocol number or name as shown in [Table 12](#):

Table 12. *Well-Known Protocol Types*

Number	Protocol Name
1	icmp
2	igmp
6	tcp
17	udp
89	ospf
112	vrrp

- IPv6 header options (for IPv6 ACLs only)
 - Source IPv6 address and prefix length
 - Destination IPv6 address and prefix length
 - Next Header value
 - Flow Label value
 - Traffic Class value

- TCP/UDP header options (for all ACLs)
 - TCP/UDP application source port and mask as shown in [Table 13](#)
 - TCP/UDP application destination port as shown in [Table 13](#)

Table 13. *Well-Known Application Ports*

Port	TCP/UDP Application	Port	TCP/UDP Application	Port	TCP/UDP Application
20	ftp-data	79	finger	179	bgp
21	ftp	80	http	194	irc
22	ssh	109	pop2	220	imap3
23	telnet	110	pop3	389	ldap
25	smtp	111	sunrpc	443	https
37	time	119	nntp	520	rip
42	name	123	ntp	554	rtsp
43	whois	143	imap	1645/1812	Radius
53	domain	144	news	1813	Radius
69	tftp	161	snmp	1985	Accounting
70	gopher	162	snmptrap		hsrp

- TCP/UDP flag value as shown in [Table 14](#)

Table 14. *Well-Known TCP flag values*

Flag	Value
URG	0x0020
ACK	0x0010
PSH	0x0008
RST	0x0004
SYN	0x0002
FIN	0x0001

- Packet format (for IPv4 ACLs and VMaps only)
 - Ethernet format (eth2, SNAP, LLC)
 - Ethernet tagging format
 - IP format (IPv4, IPv6)

Summary of ACL Actions

Once classified using ACLs, the identified packet flows can be processed differently. For each ACL, an *action* can be assigned. The action determines how the switch treats packets that match the classifiers assigned to the ACL. G8296 ACL actions include the following:

- Pass or Drop the packet
- Re-mark the packet with a new DiffServ Code Point (DSCP)
- Re-mark the 802.1p field
- Set the COS queue

Assigning Individual ACLs to a Port

Once you configure an ACL, you must assign the ACL to the appropriate ports. Each port can accept multiple ACLs, and each ACL can be applied for multiple ports. ACLs can be assigned individually.

To assign an individual ACLs to a port, use the following IP Interface Mode commands:

```
RS8296(config)# interface port <port>  
RS8296(config-if)# access-control list <IPv4 ACL number>  
RS8296(config-if)# access-control list6 <IPv6 ACL number>
```

When multiple ACLs are assigned to a port, higher-priority ACLs are considered first, and their action takes precedence over lower-priority ACLs. ACL order of precedence is discussed in the next section.

ACL Order of Precedence

If no assigned ACL matches the port traffic, no ACL action is applied.

ACL Metering and Re-Marking

You can define a profile for the aggregate traffic flowing through the G8296 by configuring a QoS meter (if desired) and assigning ACLs to ports.

Note: When you add ACLs to a port, make sure they are ordered correctly in terms of precedence (see [“ACL Order of Precedence” on page 107](#)).

Actions taken by an ACL are called *In-Profile* actions. You can configure additional In-Profile and Out-of-Profile actions on a port. Data traffic can be metered, and re-marked to ensure that the traffic flow provides certain levels of service in terms of bandwidth for different types of network traffic.

Metering

QoS metering provides different levels of service to data streams through user-configurable parameters. A meter is used to measure the traffic stream against a traffic profile which you create. Thus, creating meters yields In-Profile and Out-of-Profile traffic for each ACL, as follows:

- **In-Profile** If there is no meter configured or if the packet conforms to the meter, the packet is classified as In-Profile.
- **Out-of-Profile** If a meter is configured and the packet does not conform to the meter (exceeds the committed rate or maximum burst rate of the meter), the packet is classified as Out-of-Profile.

Using meters, you set a Committed Rate in Kbps . All traffic within this Committed Rate is In-Profile. Additionally, you can set a Maximum Burst Size that specifies an allowed data burst larger than the Committed Rate for a brief period. These parameters define the In-Profile traffic.

Meters keep the sorted packets within certain parameters. You can configure a meter on an ACL, and perform actions on metered traffic, such as packet re-marking.

Re-Marking

Re-marking allows for the treatment of packets to be reset based on new network specifications or desired levels of service. You can configure the ACL to re-mark a packet as follows:

- Change the DSCP value of a packet, used to specify the service level that traffic receives.
- Change the 802.1p priority of a packet.

ACL Port Mirroring

For IPv4 ACLs and VMaps, packets that match the filter can be mirrored to another switch port for network diagnosis and monitoring.

The source port for the mirrored packets cannot be a portchannel, but may be a member of a portchannel.

The destination port to which packets are mirrored must be a physical port.

The action (permit, drop, etc.) of the ACL or VMap must be configured before assigning it to a port.

Use the following commands to add mirroring to an ACL:

- For IPv4 ACLs:

```
RS8296(config)# access-control list <ACL number> mirror port <destination  
port>
```

The ACL must be also assigned to its target ports as usual (see [“Assigning Individual ACLs to a Port”](#) on page 106).

- For VMaps (see [“VLAN Maps”](#) on page 112):

```
RS8296(config)# access-control vmap <VMap number> mirror port <monitor  
destination port>
```

See the configuration example on [page 113](#).

Viewing ACL Statistics

ACL statistics display how many packets have “hit” (matched) each ACL. Use ACL statistics to check filter performance or to debug the ACL filter configuration.

You must enable statistics for each ACL that you wish to monitor:

```
RS8296(config)# access-control list <ACL number> statistics
```

ACL Configuration Examples

ACL Example 1

Use this configuration to block traffic to a specific host. All traffic that ingresses on port 1 is denied if it is destined for the host at IP address 100.10.1.1

1. Configure an Access Control List.

```
RS8296(config)# access-control list 1 ipv4 destination-ip-address  
100.10.1.1  
RS8296(config)# access-control list 1 action deny
```

2. Add ACL 1 to port 1.

```
RS8296(config)# interface port 1  
RS8296(config-if)# access-control list 1  
RS8296(config-if)# exit
```

ACL Example 2

Use this configuration to block traffic from a network destined for a specific host address. All traffic that ingresses in port 2 with source IP from class 100.10.1.0/24 and destination IP 200.20.2.2 is denied.

1. Configure an Access Control List.

```
RS8296(config)# access-control list 2 ipv4 source-ip-address 100.10.1.0  
255.255.255.0  
RS8296(config)# access-control list 2 ipv4 destination-ip-address  
200.20.2.2 255.255.255.255  
RS8296(config)# access-control list 2 action deny
```

2. Add ACL 2 to port 2.

```
RS8296(config)# interface port 2  
RS8296(config-if)# access-control list 2  
RS8296(config-if)# exit
```

ACL Example 3

Use this configuration to block traffic from a specific IPv6 source address. All traffic that ingresses in port 2 with source IP from class 2001:0:0:5:0:0:2/128 is denied.

1. Configure an Access Control List.

```
RS8296(config)# access-control list6 3 ipv6 source-address  
                2001:0:0:5:0:0:2 128  
RS8296(config)# access-control list6 3 action deny
```

2. Add ACL 2 to port 2.

```
RS8296(config)# interface port 2  
RS8296(config-if)# access-control list6 3  
RS8296(config-if)# exit
```

ACL Example 4

Use this configuration to deny all ARP packets that ingress a port.

1. Configure an Access Control List.

```
RS8296(config)# access-control list 2 ethernet ethernet-type arp  
RS8296(config)# access-control list 2 action deny
```

2. Add ACL 2 to port EXT2.

```
RS8296(config)# interface port 2  
RS8296(config-if)# access-control list 2  
RS8296(config-if)# exit
```

ACL Example 5

Use the following configuration to permit access to hosts with destination MAC address that matches 11:05:00:10:00:00 FF:F5:FF:FF:FF:FF and deny access to all other hosts.

1. Configure Access Control Lists.

```
RS8296(config)# access-control list 30 ethernet destination-mac-address  
                11:05:00:10:00:00 FF:F5:FF:FF:FF:FF  
RS8296(config)# access-control list 30 action permit  
RS8296(config)# access-control list 100 ethernet destination-mac-address  
                00:00:00:00:00:00 00:00:00:00:00:00  
RS8296(config)# access-control list 100 action deny
```

2. Add ACLs to a port.

```
RS8296(config)# interface port 2  
RS8296(config-if)# access-control list 30  
RS8296(config-if)# access-control list 100  
RS8296(config-if)# exit
```

VLAN Maps

A VLAN map (VMap) is an ACL that can be assigned to a VLAN or VM group rather than to a switch port as with IPv4 ACLs. This is particularly useful in a virtualized environment where traffic filtering and metering policies must follow virtual machines (VMs) as they migrate between hypervisors.

The G8296 supports up to VMaps.

Individual VMap filters are configured in the same fashion as IPv4 ACLs, except that VLANs cannot be specified as a filtering criteria (unnecessary, since the VMap are assigned to a specific VLAN or associated with a VM group VLAN).

VMaps are configured using the following ISCLI configuration command path:

```
RS8296(config)# access-control vmap <VMap ID> ?
  action          Set filter action
  ethernet        Ethernet header options
  ipv4            IP version 4 header options
  meter           ACL metering configuration
  packet-format   Set to filter specific packet format types
  re-mark         ACL re-mark configuration
  statistics      Enable access control list statistics
  tcp-udp         TCP and UDP filtering options
```

Once a VMap filter is created, it can be assigned or removed using the following configuration commands:

- For regular VLAN, use config-vlan mode:

```
RS8296(config)# vlan <VLAN ID>
RS8296(config-vlan)# [no] vmap <VMap ID> [serverports | non-serverports]
```

- For a VM group (see [“VM Group Types” on page 242](#)), use the global configuration mode:

```
RS8296(config)# [no] virt vmgroup <ID> vmap <VMap ID>
[serverports | non-serverports]
```

Note: Each VMap can be assigned to only one VLAN or VM group. However, each VLAN or VM group may have multiple VMaps assigned to it.

When the optional `serverports` or `non-serverports` parameter is specified, the action to add or remove the VMap is applied for either the switch server ports (`serverports`) or uplink ports (`non-serverports`). If omitted, the operation will be applied to all ports in the associated VLAN or VM group.

VMap Example

In this example, EtherType 2 traffic from VLAN 3 server ports is mirrored to a network monitor on port 4.

```
RS8296(config)# access-control vmap 21 packet-format ethernet
ethernet-type2
RS8296(config)# access-control vmap 21 mirror port 4
RS8296(config)# access-control vmap 21 action permit
RS8296(config)# vlan 3
RS8296(config-vlan)# vmap 21 serverports
```

Management ACLs

Management ACLs (MACLs) filter inbound traffic i.e. traffic toward the CPU. MACLs are applied switch-wide. Traffic can be filtered based on the following:

- IPv4 source address
- IPv4 destination address
- IPv4 protocols
- TCP/UDP destination or source port

Lower MACL numbers have higher priority.

Following is an example MACL configuration based on a destination IP address and a TCP-UDP destination port:

```
RS8296(config)# access-control mac1 1 ipv4 destination-ip-address 1.1.1.1
255.255.255.0
RS8296(config)# access-control mac1 1 tcp-udp destination-port 111 0xffff
RS8296(config)# access-control mac1 1 statistics
RS8296(config)# access-control mac1 1 action permit
RS8296(config)# access-control mac1 1 enable
```

Use the following command to view the MACL configuration:

```
RS8296(config)# show access-control mac1 1

MACL 1 profile : Enabled
IPv4
- DST IP      : 1.1.1.1/255.255.255.0
TCP/UDP
- DST Port    : 111/0xffff
Action       : Permit
Statistics   : Enabled
```

Using Storm Control Filters

Excessive transmission of broadcast or multicast traffic can result in a network storm. A network storm can overwhelm your network with constant broadcast or multicast traffic, and degrade network performance. Common symptoms of a network storm are denial-of-service (DoS) attacks, slow network response times, and network operations timing out.

The G8296 provides filters that can limit the number of the following packet types transmitted by switch ports:

- Broadcast packets
- Multicast packets
- Unknown unicast packets (destination lookup failure)

Unicast packets whose destination MAC address is not in the Forwarding Database are *unknown unicasts*. When an unknown unicast is encountered, the switch handles it like a broadcast packet and floods it to all other ports in the VLAN (broadcast domain). A high rate of unknown unicast traffic can have the same negative effects as a broadcast storm.

Configuring Storm Control

Configure broadcast filters on each port that requires broadcast storm control. Set a threshold that defines the total number of broadcast packets transmitted (), in packets per second. When the threshold is reached, no more packets of the specified type are transmitted.

To filter broadcast packets on a port, use the following commands:

```
RS8296(config)# interface port 1
RS8296(config-if)# storm-control broadcast level rate <packets per second>
```

To filter multicast packets on a port, use the following commands:

```
RS8296(config-if)# storm-control multicast level rate <packets per second>
```

To filter unknown unicast packets on a port, use the following commands:

```
RS8296(config-if)# storm-control unicast level rate <packets per second>
RS8296(config-if)# exit
```


Part 3: Switch Basics

This section discusses basic switching functions:

- VLANs
- Port Trunking
- Spanning Tree Protocols (Spanning Tree Groups, Rapid Spanning Tree Protocol, and Multiple Spanning Tree Protocol)
- Virtual Link Aggregation Groups
- Quality of Service

Chapter 9. VLANs

This chapter describes network design and topology considerations for using Virtual Local Area Networks (VLANs). VLANs commonly are used to split up groups of network users into manageable broadcast domains, to create logical segmentation of workgroups, and to enforce security policies among logical segments. The following topics are discussed in this chapter:

- [“VLANs and Port VLAN ID Numbers” on page 120](#)
- [“VLAN Tagging/Trunk Mode” on page 122](#)
- [“VLAN Topologies and Design Considerations” on page 126](#)
This section discusses how you can connect users and segments to a host that supports many logical segments or subnets by using the flexibility of the multiple VLAN system.
- [“Private VLANs” on page 129](#)

Note: VLANs can be configured from the Command Line Interface (see “VLAN Configuration” as well as “Port Configuration” in the *Command Reference*).

VLANs Overview

Setting up virtual LANs (VLANs) is a way to segment networks to increase network flexibility without changing the physical network topology. With network segmentation, each switch port connects to a segment that is a single broadcast domain. When a switch port is configured to be a member of a VLAN, it is added to a group of ports (workgroup) that belong to one broadcast domain.

Ports are grouped into broadcast domains by assigning them to the same VLAN. Frames received in one VLAN can only be forwarded within that VLAN, and multicast, broadcast, and unknown unicast frames are flooded only to ports in the same VLAN.

The RackSwitch G8296 (G8296) supports jumbo frames with a Maximum Transmission Unit (MTU) of bytes. Within each frame, 18 bytes are reserved for the Ethernet header and CRC trailer. The remaining space in the frame comprise the packet, which includes the payload and any additional overhead, such as 802.1q or VLAN tags. Jumbo frame support is automatic: it is enabled by default, requires no manual configuration, and cannot be manually disabled.

VLANs and Port VLAN ID Numbers

VLAN Numbers

The G8296 supports up to 4095 VLANs per switch. Each can be identified with any number between 1 and 4094. VLAN 1 is the default VLAN for the data ports. VLAN 4095 is used by the management network, which includes the management ports.

Use the following command to view VLAN information:

```
RS G8296#show vlan
```

VLAN	Name	Status	Ports	
1	Default VLAN	ena	1-87 88 89-96	
4095	Mgmt VLAN	ena	MGT	
Primary	Secondary	Type	Ports	vPorts

PVID/Native VLAN Numbers

Each port in the switch has a configurable default VLAN number, known as its *PVID*. By default, the PVID for all non-management ports is set to 1, which correlates to the default VLAN ID. The PVID for each port can be configured to any VLAN number between 1 and 4094.

Use the following command to view PVIDs:

Use the following command to set the port PVID/Native VLAN:

```
Access Mode Port

RS8296(config)# interface port <port number>
RS8296(config-if)# switchport access vlan <VLAN ID>

For Trunk Mode Port

RS8296(config)# interface port <port number>
RS8296(config-if)# switchport trunk native vlan <VLAN ID>
```

Each port on the switch can belong to one or more VLANs, and each VLAN can have any number of switch ports in its membership. Any port that belongs to multiple VLANs, however, must have VLAN *tagging*/trunk mode enabled (see [“VLAN Tagging/Trunk Mode”](#) on page 122).

VLAN Tagging/Trunk Mode

Lenovo Network Operating System software supports 802.1Q VLAN *tagging*, providing standards-based VLAN support for Ethernet systems.

Tagging places the VLAN identifier in the frame header of a packet, allowing each port to belong to multiple VLANs. When you add a port to multiple VLANs, you also must enable tagging on that port.

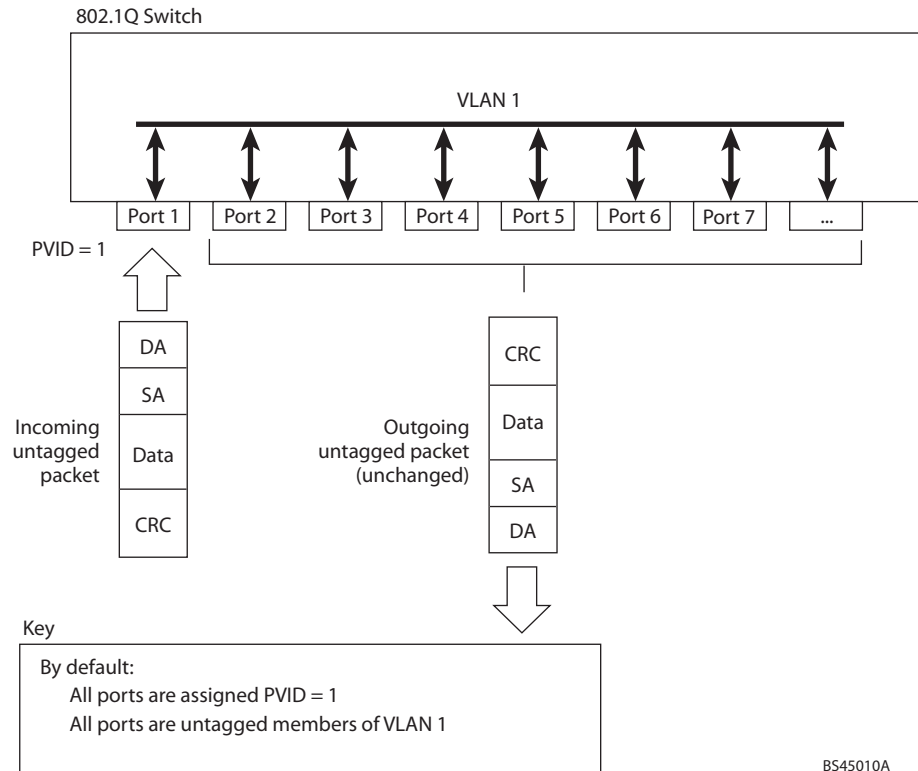
Since tagging fundamentally changes the format of frames transmitted on a tagged port, you must carefully plan network designs to prevent tagged frames from being transmitted to devices that do not support 802.1Q VLAN tags, or devices where tagging is not enabled.

Important terms used with the 802.1Q tagging feature are:

- VLAN identifier (VID)—the 12-bit portion of the VLAN tag in the frame header that identifies an explicit VLAN.
- Port VLAN identifier (PVID)—a classification mechanism that associates a port with a specific VLAN. For example, a port with a PVID of 3 (PVID =3) assigns all untagged frames received on this port to VLAN 3. Any untagged frames received by the switch are classified with the PVID of the receiving port.
- Tagged frame—a frame that carries VLAN tagging information in the header. This VLAN tagging information is a 32-bit field (VLAN tag) in the frame header that identifies the frame as belonging to a specific VLAN. Untagged frames are marked (tagged) with this classification as they leave the switch through a port that is configured as a tagged port.
- Untagged frame— a frame that does not carry any VLAN tagging information in the frame header.
- Untagged member—a port that has been configured as an untagged member of a specific VLAN. When an untagged frame exits the switch through an untagged member port, the frame header remains unchanged. When a tagged frame exits the switch through an untagged member port, the tag is stripped and the tagged frame is changed to an untagged frame.
- Tagged member—a port that has been configured as a tagged member of a specific VLAN. When an untagged frame exits the switch through a tagged member port, the frame header is modified to include the 32-bit tag associated with the PVID. When a tagged frame exits the switch through a tagged member port, the frame header remains unchanged (original VID remains).

Note: If a 802.1Q tagged frame is received by a port that has VLAN-tagging disabled and the port VLAN ID (PVID) is different than the VLAN ID of the packet, then the frame is dropped at the ingress port.

Figure 2. Default VLAN settings



Note: The port numbers specified in these illustrations may not directly correspond to the physical port configuration of your switch model.

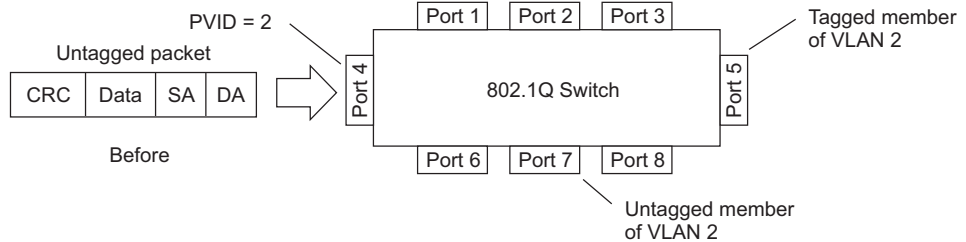
When a VLAN is configured, ports are added as members of the VLAN, and the ports are defined as either *tagged* or *untagged* (see [Figure 3](#) through [Figure 6](#)).

The default configuration settings for the G8296 has all ports set as untagged members of VLAN 1 with all ports configured as PVID = 1. In the default configuration example shown in [Figure 2](#), all incoming packets are assigned to VLAN 1 by the default port VLAN identifier (PVID = 1).

[Figure 3](#) through [Figure 6](#) illustrate generic examples of VLAN tagging. In [Figure 3](#), untagged incoming packets are assigned directly to VLAN 2 (PVID = 2). Port 5 is configured as a *tagged* member of VLAN 2, and port 7 is configured as an *untagged* member of VLAN 2.

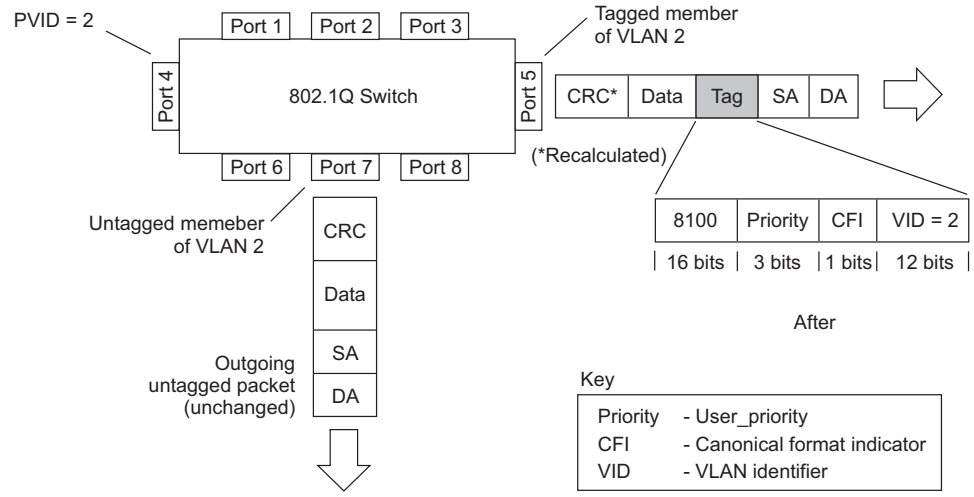
Note: The port assignments in the following figures are not meant to match the G8296.

Figure 3. Port-based VLAN assignment



As shown in [Figure 4](#), the untagged packet is marked (tagged) as it leaves the switch through port 5, which is configured as a tagged member of VLAN 2. The untagged packet remains unchanged as it leaves the switch through port 7, which is configured as an untagged member of VLAN 2.

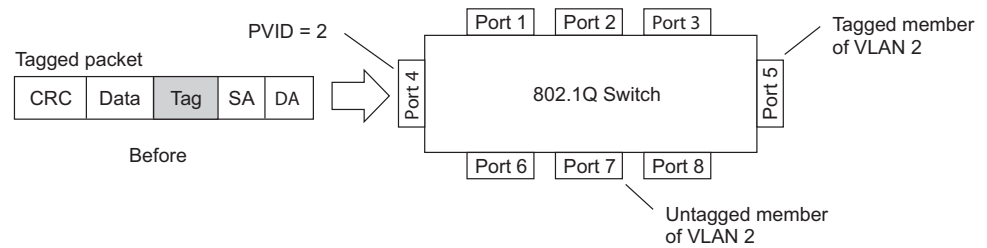
Figure 4. 802.1Q tagging (after port-based VLAN assignment)



BS45012A

In [Figure 5](#), tagged incoming packets are assigned directly to VLAN 2 because of the tag assignment in the packet. Port 5 is configured as a *tagged* member of VLAN 2, and port 7 is configured as an *untagged* member of VLAN 2.

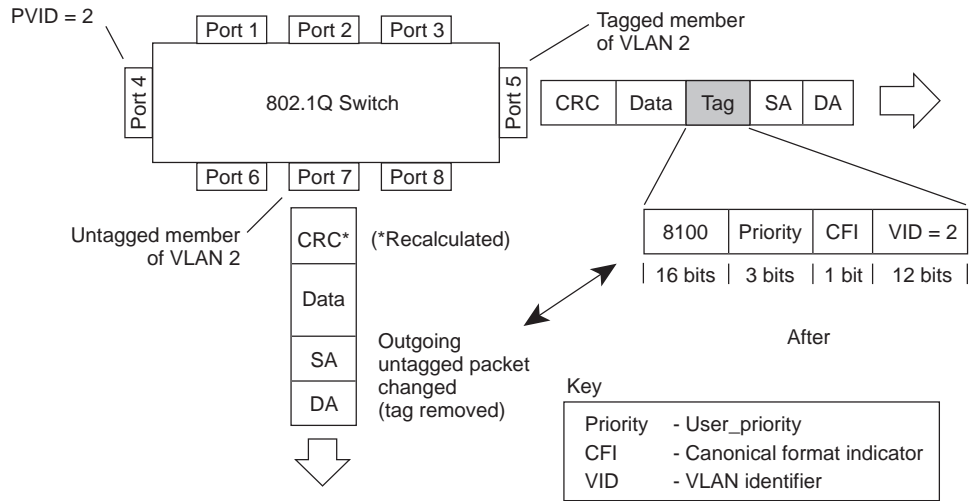
Figure 5. 802.1Q tag assignment



BS45013A

As shown in [Figure 6](#), the tagged packet remains unchanged as it leaves the switch through port 5, which is configured as a tagged member of VLAN 2. However, the tagged packet is stripped (untagged) as it leaves the switch through port 7, which is configured as an untagged member of VLAN 2.

Figure 6. 802.1Q tagging (after 802.1Q tag assignment)



BS45014A

VLAN Topologies and Design Considerations

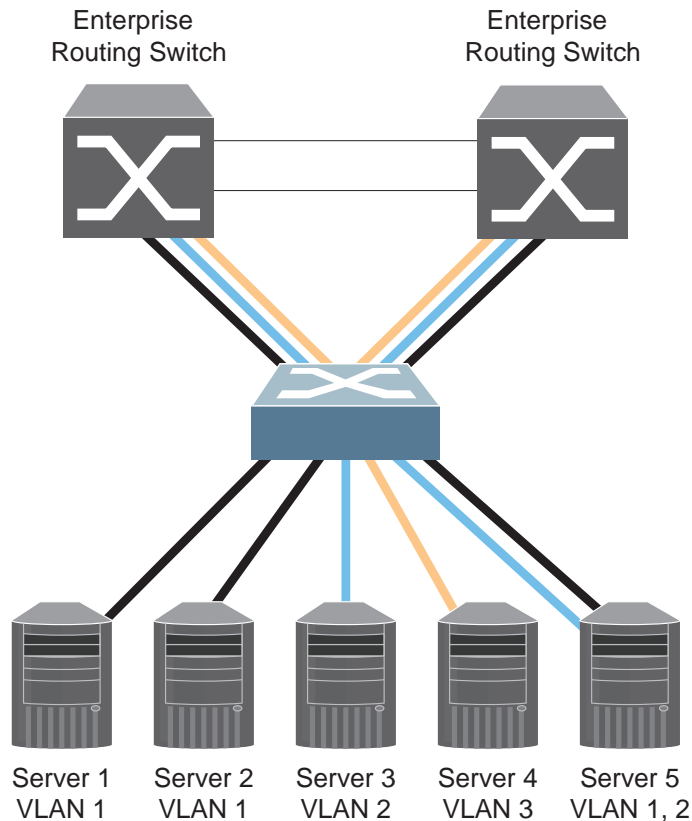
Note the following when working with VLAN topologies:

- By default, the G8296 software is configured so that tagging/trunk mode is disabled on all ports.
- By default, the G8296 software is configured so that all data ports are members of VLAN 1.
- When using Spanning Tree, STG 2-128 may contain only one VLAN unless Multiple Spanning-Tree Protocol (MSTP) mode is used. With MSTP mode, STG 1 to 32 can include multiple VLANs.
- All ports involved in both trunking and port mirroring must have the same VLAN configuration. If a port is on a trunk with a mirroring port, the VLAN configuration cannot be changed. For more information trunk groups, see [“Ports and Trunking” on page 131](#) and [“Port Mirroring” on page 577](#).

Multiple VLANs with Tagging/Trunk Mode Adapters

[Figure 7](#) illustrates a network topology described in [Note: on page 127](#) and the configuration example on [page 128](#).

Figure 7. Multiple VLANs with VLAN-Tagged Gigabit Adapters



The features of this VLAN are described in the following table.

Table 15. *Multiple VLANs Example*

Component	Description
G8296 switch	This switch is configured with three VLANs that represent three different IP subnets. Five ports are connected downstream to servers. Two ports are connected upstream to routing switches. Uplink ports are members of all three VLANs, with VLAN tagging/trunk mode enabled.
Server 1	This server is a member of VLAN 1 and has presence in only one IP subnet. The associated switch port is only a member of VLAN 1, so tagging/trunk mode is disabled.
Server 2	This server is a member of VLAN 1 and has presence in only one IP subnet. The associated switch port is only a member of VLAN 1, so tagging/trunk mode is disabled.
Server 3	This server belongs to VLAN 2, and it is logically in the same IP subnet as Server 5. The associated switch port has tagging/trunk mode disabled.
Server 4	A member of VLAN 3, this server can communicate only with other servers via a router. The associated switch port has tagging/trunk mode disabled.
Server 5	A member of VLAN 1 and VLAN 2, this server can communicate only with Server 1, Server 2, and Server 3. The associated switch port has tagging/trunk mode enabled.
Enterprise Routing switches	These switches must have all three VLANs (VLAN 1, 2, 3) configured. They can communicate with Server 1, Server 2, and Server 5 via VLAN 1. They can communicate with Server 3 and Server 5 via VLAN 2. They can communicate with Server 4 via VLAN 3. Tagging/trunk mode on switch ports is enabled.

Note: VLAN tagging/trunk mode is required only on ports that are connected to other switches or on ports that connect to tag-capable end-stations, such as servers with VLAN-tagging/trunk mode adapters.

To configure a specific VLAN on a trunk port, the following conditions must be met:

- The port must be in trunk mode.
- The VLAN must be in the trunk's allowed VLAN range. By default, the range includes all VLANs.
- The VLAN must be un-reserved.
- The VLAN must be created.

The order in which the conditions above are met is not relevant. However, all conditions must be met collectively. When the final condition is met, the VLAN is enabled on the port. When the first condition is broken, the VLAN is disabled.

VLAN Configuration Example

Use the following procedure to configure the example network shown in [Figure 7 on page 126](#).

1. Enable VLAN tagging/trunk mode on server ports that support multiple VLANs.

```
RS8296(config)# interface port 5
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 2
RS8296(config-if)# exit
```

2. Enable tagging/trunk mode on uplink ports that support multiple VLANs.

```
RS8296(config)# interface port 19
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 2,3
RS8296(config-if)# exit
RS8296(config)# interface port 20
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 2,3
RS8296(config-if)# exit
```

3. Configure server ports that belong to a single VLAN.

```
RS8296(config)# interface port 4
RS8296(config-if)# switchport mode access
RS8296(config-if)# switchport access vlan 2
RS8296(config-if)# exit
```

By default, all ports are members of VLAN 1, so configure only those ports that belong to other VLANs.

Private VLANs

Private VLANs provide Layer 2 isolation between the ports within the same broadcast domain. Private VLANs can control traffic within a VLAN domain, and provide port-based security for host servers.

Networking OS supports Private VLAN configuration as described in RFC 5517.

Use Private VLANs to partition a VLAN domain into sub-domains. Each sub-domain is comprised of one primary VLAN and one or more secondary VLANs, as follows:

- Primary VLAN—carries unidirectional traffic downstream from promiscuous ports. Each Private VLAN configuration has only one primary VLAN. All ports in the Private VLAN are members of the primary VLAN.
- Secondary VLAN—Secondary VLANs are internal to a private VLAN domain, and are defined as follows:
 - Isolated VLAN—carries unidirectional traffic upstream from the host servers toward ports in the primary VLAN and the gateway. Each Private VLAN configuration can contain only one isolated VLAN.
 - Community VLAN—carries upstream traffic from ports in the community VLAN to other ports in the same community, and to ports in the primary VLAN and the gateway. Each Private VLAN configuration can contain multiple community VLANs.

After you define the primary VLAN and one or more secondary VLANs, you map the secondary VLAN(s) to the primary VLAN.

Private VLAN Ports

Private VLAN ports are defined as follows:

- Promiscuous—A promiscuous port is a port that belongs to the primary VLAN. The promiscuous port can communicate with all the interfaces, including ports in the secondary VLANs (Isolated VLAN and Community VLANs).
- Isolated—An isolated port is a host port that belongs to an isolated VLAN. Each isolated port has complete layer 2 separation from other ports within the same private VLAN (including other isolated ports), except for the promiscuous ports.
 - Traffic sent to an isolated port is blocked by the Private VLAN, except the traffic from promiscuous ports.
 - Traffic received from an isolated port is forwarded only to promiscuous ports.
- Community—A community port is a host port that belongs to a community VLAN. Community ports can communicate with other ports in the same community VLAN, and with promiscuous ports. These interfaces are isolated at layer 2 from all other interfaces in other communities and from isolated ports within the Private VLAN.

Configuration Guidelines

The following guidelines apply when configuring Private VLANs:

- The default VLAN 1 cannot be a Private VLAN.
- IGMP Snooping must be disabled on Private VLANs.
- All VLANs that comprise the Private VLAN must belong to the same Spanning Tree Group.

Configuration Example

Follow this procedure to configure a Private VLAN.

1. Select a VLAN and define the Private VLAN type as primary.

```
RS8296(config)# vlan 700
RS8296(config-vlan)# private-vlan primary
RS8296(config-vlan)# exit
```

2. Configure a promiscuous port for VLAN 700.

```
RS8296(config)# interface port 1
RS8296(config-if)# switchport mode private-vlan
RS8296(config-if)# switchport private-vlan mapping 700
RS8296(config-if)# exit
```

3. Configure two secondary VLANs: isolated VLAN and community VLAN.

```
RS8296(config)# vlan 701
RS8296(config-vlan)# private-vlan isolated
RS8296(config-vlan)# exit
RS8296(config)# vlan 702
RS8296(config-vlan)# private-vlan community
RS8296(config-vlan)# exit
```

4. Map secondary VLANs to primary VLAN.

```
RS8296(config)# vlan 700-702
RS8296(config-vlan)# stg 1
RS8296(config-vlan)# exit
RS8296(config)# vlan 700
RS8296(config-vlan)# private-vlan association 701,702
RS8296(config-vlan)# exit
```

5. Configure host ports for secondary VLANs.

```
RS8296(config)# interface port 2
RS8296(config-if)# switchport mode private-vlan
RS8296(config-if)# switchport private-vlan host-association 700 701
RS8296(config-if)# exit

RS8296(config)# interface port 3
RS8296(config-if)# switchport mode private-vlan
RS8296(config-if)# switchport private-vlan host-association 700 702
RS8296(config-if)# exit
```

6. Verify the configuration.

```
RS8296(config)# show vlan private-vlan
```

Primary	Secondary	Type	Ports	vPorts
700	701	isolated	1 2	empty
700	702	community	1 3	empty

Chapter 10. Ports and Trunking

Trunk groups can provide super-bandwidth, multi-link connections between the RackSwitch G8296 (G8296) and other trunk-capable devices. A trunk group is a group of ports that act together, combining their bandwidth to create a single, larger virtual link. This chapter provides configuration background and examples for trunking multiple ports together:

- [“Configuring QSFP+ Ports” on page 132](#)
- [“Trunking Overview” on page 134”](#)
- [“Configuring a Static Port Trunk” on page 136](#)
- [“Configurable Trunk Hash Algorithm” on page 142](#)
- [“Link Aggregation Control Protocol” on page 138](#)

Configuring QSFP+ Ports

QSFP+ ports support both 10GbE and 40GbE, as shown in [Table 16](#).

Table 16. *QSFP+ Port Numbering*

Physical Port Number	40GbE mode	10GbE mode
Port 87	Port 87	Ports 87/1-87/4
Port 88	Port 88	Ports 88/1-88/4
Port 89	Port 89	
Port 90	Port 90	
Port 91	Port 91	
Port 92	Port 92	
Port 93	Port 93	
Port 94	Port 94	
Port 95	Port 95	
Port 96	Port 96	

Use the following procedure to change the QSFP+ port mode.

1. Display the current port mode for the QSFP+ ports.

```
# show boot qsf-p-port-modes
QSFP ports booted configuration:
  Port 87 - 40G Mode
  Port 88 - 40G Mode
  Port 89 - 40G Mode
  Port 90 - 40G Mode
  Port 91 - 40G Mode
  Port 92 - 40G Mode
  Port 93 - 40G Mode
  Port 94 - 40G Mode
  Port 95 - 40G Mode
  Port 96 - 40G Mode

QSFP ports saved configuration:
  Port 87 - 40G Mode
  Port 88 - 40G Mode
```

2. Change the port mode to 10GbE. Select the physical port number.

```
RS8296(config)#no boot qsf-p-40Gports 87
```

3. Verify the change.

```
# show boot qsfp-port-modes
QSFN ports booted configuration:
Port 87 - 40G Mode
Port 88 - 40G Mode
Port 89 - 40G Mode
Port 90 - 40G Mode
Port 91 - 40G Mode
Port 92 - 40G Mode
Port 93 - 40G Mode
Port 94 - 40G Mode
Port 95 - 40G Mode
Port 96 - 40G Mode

QSFN ports saved configuration:
Port 87/1, 87/2, 87/3, 87/4 - 10G Mode
Port 88 - 40G Mode
```

4. Reset the switch.

```
RS8296(config)# reload
```

Use the following command to reset a port to 40GbE mode.

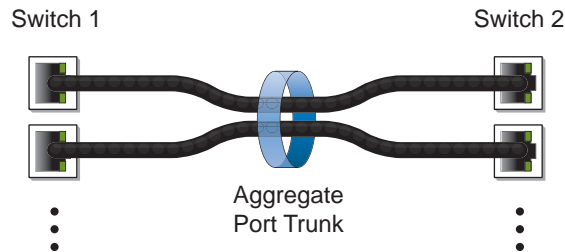
```
RS8296(config)# boot qsfp-40Gports 87/1
```

Trunking Overview

When using port trunk groups between two switches, as shown in [Figure 8](#), you can create a virtual link between the switches, operating with combined throughput levels that depends on how many physical ports are included.

Each G8296 supports up to 102 trunk groups. Two trunk types are available: static trunk groups (portchannel), and dynamic LACP trunk groups. Each type can contain up to 32 member ports, depending on the port type and availability.

Figure 8. Port Trunk Group



Trunk groups are also useful for connecting a G8296 to third-party devices that support link aggregation, such as Cisco routers and switches with EtherChannel technology (*not* ISL trunking technology) and Sun's Quad Fast Ethernet Adapter. Trunk Group technology is compatible with these devices when they are configured manually.

Trunk traffic is statistically distributed among the ports in a trunk group, based on a variety of configurable options.

Also, since each trunk group is comprised of multiple physical links, the trunk group is inherently fault tolerant. As long as one connection between the switches is available, the trunk remains active and statistical load balancing is maintained whenever a port in a trunk group is lost or returned to service.

Static Trunks

Static Trunk Requirements

When you create and enable a static trunk, the trunk members (switch ports) take on certain settings necessary for correct operation of the trunking feature.

Before you configure your trunk, you must consider these settings, along with specific configuration rules, as follows:

1. Read the configuration rules provided in the section, [“Static Trunk Group Configuration Rules” on page 135](#).
2. Determine which switch ports (up to 32) are to become *trunk members* (the specific ports making up the trunk).
3. Ensure that the chosen switch ports are set to **enabled**. Trunk member ports must have the same VLAN and Spanning Tree configuration.
4. Consider how the existing Spanning Tree will react to the new trunk configuration. See [Chapter 11, “Spanning Tree Protocols,”](#) for Spanning Tree Group configuration guidelines.
5. Consider how existing VLANs will be affected by the addition of a trunk.

Static Trunk Group Configuration Rules

The trunking feature operates according to specific configuration rules. When creating trunks, consider the following rules that determine how a trunk group reacts in any network topology:

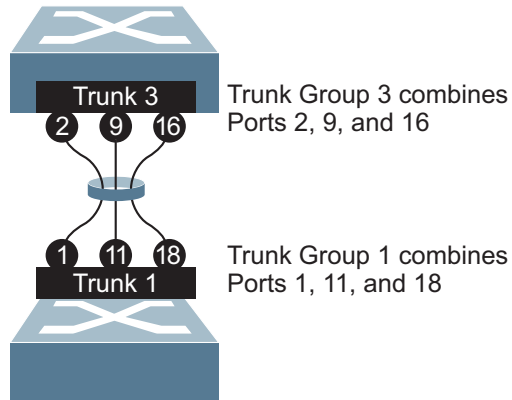
- All trunks must originate from one logical device, and lead to one logical destination device. Usually, a trunk connects two physical devices together with multiple links. However, in some networks, a single logical device may include multiple physical devices or when using VLAGs (see [“Virtual Link Aggregation Groups” on page 167](#)). In such cases, links in a trunk are allowed to connect to multiple physical devices because they act as one logical device.
- Any physical switch port can belong to only one trunk group.
- Trunking from third-party devices must comply with Cisco® EtherChannel® technology.
- All ports in a trunk must have the same link configuration (speed, duplex, flow control), the same VLAN properties, and the same Spanning Tree, storm control, and ACL configuration. It is recommended that the ports in a trunk be members of the same VLAN.
- Each trunk inherits its port configuration (speed, flow control, tagging) from the first member port. As additional ports are added to the trunk, their settings must be changed to match the trunk configuration.
- When a port leaves a trunk, its configuration parameters are retained.
- You cannot configure a trunk member as a monitor port in a port-mirroring configuration.

- Trunks cannot be monitored by a monitor port; however, trunk members can be monitored.

Configuring a Static Port Trunk

In the following example, three ports are trunked between two switches.

Figure 9. Port Trunk Group Configuration Example



Prior to configuring each switch in this example, you must connect to the appropriate switches as the administrator.

Note: For details about accessing and using any of the commands described in this example, see the *RackSwitch G8296 ISCLI Reference*.

1. Follow these steps on the G8296:
 - a. Define a trunk group.

```
RS8296(config)# portchannel 3 port 2,9,16
RS8296(config)# portchannel 3 enable
```

- b. Verify the configuration.

```
# show portchannel information
```

Examine the resulting information. If any settings are incorrect, make appropriate changes.

2. Repeat the process on the other switch.

```
RS8296(config)# portchannel 1 port 1,11,18
RS8296(config)# portchannel 1 enable
```

3. Connect the switch ports that will be members in the trunk group.

Trunk group 3 (on the G8296) is now connected to trunk group 1 (on the other switch).

Note: In this example, two G8296 switches are used. If a third-party device supporting link aggregation is used (such as Cisco routers and switches with EtherChannel technology or Sun's Quad Fast Ethernet Adapter), trunk groups on the third-party device must be configured manually. Connection problems could arise when using automatic trunk group negotiation on the third-party device.

4. Examine the trunking information on each switch.

```
# show portchannel information
PortChannel 3: Enabled
Protocol-Static
port state:
  2: STG 1 forwarding
  9: STG 1 forwarding
 16: STG 1 forwarding
```

Information about each port in each configured trunk group is displayed. Make sure that trunk groups consist of the expected ports and that each port is in the expected state.

The following restrictions apply:

- Any physical switch port can belong to only one trunk group.
- Up to 32 ports can belong to the same trunk group.
- All ports in static trunks must have the same link configuration (speed, duplex, flow control).
- Trunking from third-party devices must comply with Cisco® EtherChannel® technology.

Link Aggregation Control Protocol

LACP Overview

Link Aggregation Control Protocol (LACP) is an IEEE 802.3ad standard for grouping several physical ports into one logical port (known as a dynamic trunk group or Link Aggregation group) with any device that supports the standard. Please refer to IEEE 802.3ad-2002 for a full description of the standard.

The 802.3ad standard allows standard Ethernet links to form a single Layer 2 link using the Link Aggregation Control Protocol (LACP). Link aggregation is a method of grouping physical link segments of the same media type and speed in full duplex, and treating them as if they were part of a single, logical link segment. If a link in a LACP trunk group fails, traffic is reassigned dynamically to the remaining link(s) of the dynamic trunk group.

Note: LACP implementation in the Lenovo Network Operating System does not support the Churn machine, an option used to detect if the port is operable within a bounded time period between the actor and the partner. Only the Marker Responder is implemented, and there is no marker protocol generator.

A port's Link Aggregation Identifier (LAG ID) determines how the port can be aggregated. The Link Aggregation ID (LAG ID) is constructed mainly from the partner switch's *system ID* and the port's *admin key*, as follows:

- **System ID:** an integer value based on the partner switch's MAC address and the system priority assigned in the CLI.
- **Admin key:** a port's Admin key is an integer value (1-65535) that you can configure in the CLI. Each switch port that participates in the same LACP trunk group must have the same *admin key* value. The Admin key is *local significant*, which means the partner switch does not need to use the same Admin key value.

For example, consider two switches, an Actor (the G8296) and a Partner (another switch), as shown in [Table 17](#).

Table 17. Actor vs. Partner LACP configuration

Actor Switch	Partner Switch	
Port 7 (admin key = 100)	Port 1 (admin key = 50)	Primary trunk group
Port 8 (admin key = 100)	Port 2 (admin key = 50)	Primary trunk group
Port 9 (admin key = 100)	Port 3 (admin key = 70)	Secondary trunk group
Port 10 (admin key = 100)	Port 4 (admin key = 70)	Secondary trunk group

In the configuration shown in [Table 17](#), Actor switch ports 7 and 8 aggregate to form an LACP trunk group with Partner switch ports 1 and 2. Only ports with the same LAG ID are aggregated in the trunk group. Actor switch ports 9 and 10 are not aggregated in the same trunk group, because although they have the same admin key on the Actor switch, their LAG IDs are different (due to a different Partner switch admin key configuration). Instead, they form a secondary trunk group with Partner switch ports 3 and 4.

LACP automatically determines which member links can be aggregated and then aggregates them. It provides for the controlled addition and removal of physical links for the link aggregation.

Static LACP Trunks

To prevent switch ports with the same admin key from forming multiple trunk groups, you can configure the LACP trunk as static. In a static LACP trunk, ports with the same admin key, but with different LAG IDs, compete to get aggregated in a trunk group. The LAG ID for the trunk group is decided based on the first port that is aggregated in the group. Ports with this LAG ID get aggregated and the other ports are placed in suspended mode. As per the configuration shown in [Table 17](#), if port 7 gets aggregated first, then the LAG ID of port 7 would be the LAG ID of the trunk. Port 8 will join the trunk while ports 9 and 10 would be placed in suspended mode. When in suspended mode, a port transmits only LACP data units (LACPDU) and discards all other traffic.

A port may also be placed in suspended mode for the following reasons:

- When LACP is configured on the port but it stops receiving LACPDUs from the partner switch.
- When the port has a different LAG ID because of the partner switch MAC or port LACP key being different. For example: when a switch is connected to two partners.

Static LACP trunks are configured by associating the LACP admin key to a portchannel ID within a dedicated LACP portchannel group range:

```
RS8296(config)# portchannel <103-204> lacp key <adminkey of the LAG>
```

LACP Port Modes

Only 32 ports can actively participate in the LAG at a given time.

Each port on the switch can have one of the following LACP modes.

- off (default)
The user can configure this port in to a regular static trunk group.
- active
The port is capable of forming an LACP trunk. This port sends LACPDU packets to partner system ports.
- passive
The port is capable of forming an LACP trunk. This port only responds to the LACPDU packets sent from an LACP *active* port.

Each active LACP port transmits LACP data units (LACPDUs), while each passive LACP port listens for LACPDUs. During LACP negotiation, the admin key is exchanged. The LACP trunk group is enabled as long as the information matches at both ends of the link. If the admin key value changes for a port at either end of the link, that port's association with the LACP trunk group is lost.

When the system is initialized, all ports by default are in LACP *off* mode and are assigned unique *admin keys*. To make a group of ports aggregatable, you assign them all the same *admin key*. You must set the port's LACP mode to *active* to activate LACP negotiation. You can set other port's LACP mode to *passive*, to reduce the amount of LACPDU traffic at the initial trunk-forming stage.

Use the following command to check whether the ports are trunked:

```
RS8296 # show lACP information
```

LACP Individual

Ports assigned with an LACP admin key are prevented by default from forming individual links if they cannot join an LACP trunk group. To override this behavior, use the following commands:

```
RS8296(config) # interface port <port no.>  
RS8296(config-if) # no lACP suspend-individual
```

This allows the selected ports to be treated as normal link-up ports, which may forward data traffic according to STP, Hot Links or other applications, if they do not receive any LACPDUs.

To configure the LACP individual setting for all the ports in a static LACP trunk, use the following commands:

```
RS8296(config)# interface portchannel lACP <adminkey of the LAG>  
RS8296(config-PortChannel)# [no] lACP suspend-individual
```

or:

```
RS8296(config)# interface portchannel <static trunk ID of the LAG>  
RS8296(config-PortChannel)# [no] lACP suspend-individual
```

Note: By default, all ports are configured with the command **lACP suspend-individual**.

LACP Minimum Links Option

For dynamic trunks that require a guaranteed amount of bandwidth to be considered useful, you can specify the minimum number of links for the trunk. If the specified minimum number of ports are not available, the trunk link will not be established. If an active LACP trunk loses one or more component links, the trunk will be placed in the down state if the number of links falls to less than the specified minimum. By default, the minimum number of links is 1, meaning that LACP trunks will remain operational as long as at least one link is available.

The LACP minimum links setting can be configured as follows:

- Via interface configuration mode:

```
RS8296(config)# interface port <port number or range>
RS8296(config-if)# port-channel min-links <minimum links>
RS8296(config-if)# exit
```

- Or via portchannel configuration mode:

```
RS8296(config)# interface portchannel lacp <LACP key>
RS8296(config-PortChannel)# port-channel min-links <minimum links>
RS8296(config-if)# exit
```

Configuring LACP

Use the following procedure to configure LACP for ports 7, 8, 9 and 10 to participate in link aggregation.

1. Configure port parameters. All ports that participate in the LACP trunk group must have the same settings, including VLAN membership.
2. Select the port range and define the admin key. Only ports with the same admin key can form an LACP trunk group.

```
RS8296(config)# interface port 7-10
RS8296(config-if)# lacp key 100
```

3. Set the LACP mode.

```
RS8296(config-if)# lacp mode active
```

4. Allow member ports to individually participate in normal data traffic if no LACPDUs are received.

```
RS8296(config-if)# no lacp suspend-individual
RS8296(config-if)# exit
```

5. Set the link aggregation as static, by associating it with trunk ID 65:

```
RS8296(config)# portchannel 105 lacp key 100
```

Configurable Trunk Hash Algorithm

Traffic in a trunk group is statistically distributed among member ports using a *hash* process where various address and attribute bits from each transmitted frame are recombined to specify the particular trunk port the frame will use.

The switch can be configured to use a variety of hashing options. To achieve the most even traffic distribution, select options that exhibit a wide range of values for your particular network. Avoid hashing on information that is not usually present in the expected traffic, or which does not vary.

The G8296 supports the following hashing options:

- Layer 2 source MAC address

```
RS8296(config)# portchannel thash l2thash 12-source-mac-address
```

- Layer 2 destination MAC address

```
RS8296(config)# portchannel thash l2thash 12-destination-mac-address
```

- Layer 2 source and destination MAC address

```
RS8296(config)# portchannel thash l2thash 12-source-destination-mac
```

- Layer 3 IPv4/IPv6 source IP address

```
RS8296(config)# portchannel thash l3thash 13-source-ip-address
```

- Layer 3 IPv4/IPv6 destination IP address

```
RS8296(config)# portchannel thash l3thash 13-destination-ip-address
```

- Layer 3 source and destination IPv4/IPv6 address (the default)

```
RS8296(config)# portchannel thash l3thash 13-source-destination-ip
```

- Layer 2 hash configuration

```
RS8296(config)# portchannel thash l3thash 13-use-l2-hash
```

- Layer 4 port hash

```
RS8296(config)# portchannel thash l4port
```

- Ingress port hash

```
RS8296(config)# portchannel thash ingress
```

The G8296 supports the following FCoE hashing options:

```
RS8296(config)# portchannel thash fcoe cntag-id
RS8296(config)# portchannel thash fcoe destination-id
RS8296(config)# portchannel thash fcoe fabric-id
RS8296(config)# portchannel thash fcoe originator-id
RS8296(config)# portchannel thash fcoe responder-id
RS8296(config)# portchannel thash fcoe source-id
```

Chapter 11. Spanning Tree Protocols

When multiple paths exist between two points on a network, Spanning Tree Protocol (STP), or one of its enhanced variants, can prevent broadcast loops and ensure that the RackSwitch G8296 (G8296) uses only the most efficient network path.

This chapter covers the following topics:

- [“Spanning Tree Protocol Modes” on page 146](#)
- [“Global STP Control” on page 147](#)
- [“PVRST Mode” on page 147](#)
- [“Rapid Spanning Tree Protocol” on page 160](#)
- [“Multiple Spanning Tree Protocol” on page 162](#)
- [“Port Type and Link Type” on page 165](#)

Spanning Tree Protocol Modes

Lenovo Network Operating System 8.2 supports the following STP modes:

- Rapid Spanning Tree Protocol (RSTP)

IEEE 802.1D (2004) RSTP allows devices to detect and eliminate logical loops in a bridged or switched network. When multiple paths exist, STP configures the network so that only the most efficient path is used. If that path fails, STP automatically configures the best alternative active path on the network to sustain network operations. RSTP is an enhanced version of IEEE 802.1D (1998) STP, providing more rapid convergence of the Spanning Tree network path states on STG 1.

See [“Rapid Spanning Tree Protocol” on page 160](#) for details.

- Per-VLAN Rapid Spanning Tree (PVRST)

PVRST mode is based on RSTP to provide rapid Spanning Tree convergence, but supports instances of Spanning Tree, allowing one STG per VLAN. PVRST mode is compatible with Cisco R-PVST/R-PVST+ mode.

PVRST is the default Spanning Tree mode on the G8296. See [“PVRST Mode” on page 147](#) for details.

- Multiple Spanning Tree Protocol (MSTP)

IEEE 802.1Q (2003) MSTP provides both rapid convergence and load balancing in a VLAN environment. MSTP allows multiple STGs, with multiple VLANs in each.

See [“Multiple Spanning Tree Protocol” on page 162](#) for details.

Global STP Control

By default, the Spanning Tree feature is globally enabled on the switch, and is set for PVRST mode. Spanning Tree (and thus any currently configured STP mode) can be globally disabled using the following command:

```
RS8296(config)# spanning-tree mode disable
```

Spanning Tree can be re-enabled by specifying the STP mode:

```
RS8296(config)# spanning-tree mode {pvrst|rstp|mst}
```

where the command options represent the following modes:

- rstp: RSTP mode
- pvrst: PVRST mode
- mst: MSTP mode

PVRST Mode

Note: Per-VLAN Rapid Spanning Tree (PVRST) is enabled by default on the G8296.

Using STP, network devices detect and eliminate logical loops in a bridged or switched network. When multiple paths exist, Spanning Tree configures the network so that a switch uses only the most efficient path. If that path fails, Spanning Tree automatically sets up another active path on the network to sustain network operations.

N/OS PVRST mode is based on IEEE 802.1w RSTP. Like RSTP, PVRST mode provides rapid Spanning Tree convergence. However, PVRST mode is enhanced for multiple instances of Spanning Tree. In PVRST mode, each VLAN may be automatically or manually assigned to one of 127 available STGs. Each STG acts as an independent, simultaneous instance of STP. PVRST uses IEEE 802.1Q tagging to differentiate STP BPDUs and is compatible with Cisco R-PVST/R-PVST+ modes.

The relationship between ports, trunk groups, VLANs, and Spanning Trees is shown in [Table 18](#).

Table 18. *Ports, Trunk Groups, and VLANs*

Switch Element	Belongs To
Port	Trunk group or one or more VLANs
Trunk group	One or more VLANs
VLAN (non-default)	<ul style="list-style-type: none">● PVRST: One VLAN per STG● RSTP: All VLANs are in STG 1● MSTP: Multiple VLANs per STG

Port States

The port state controls the forwarding and learning processes of Spanning Tree. In PVRST, the port state has been consolidated to the following: discarding, learning, and forwarding.

Due to the sequence involved in these STP states, considerable delays may occur while paths are being resolved. To mitigate delays, ports defined as *edge* ports (“Port Type and Link Type” on page 165) may bypass the discarding and learning states, and enter directly into the forwarding state.

Bridge Protocol Data Units

Bridge Protocol Data Units Overview

To create a Spanning Tree, the switch generates a configuration Bridge Protocol Data Unit (BPDU), which it then forwards out of its ports. All switches in the Layer 2 network participating in the Spanning Tree gather information about other switches in the network through an exchange of BPDUs.

A bridge sends BPDU packets at a configurable regular interval (2 seconds by default). The BPDU is used to establish a path, much like a hello packet in IP routing. BPDUs contain information about the transmitting bridge and its ports, including bridge MAC addresses, bridge priority, port priority, and path cost. If the ports are in trunk mode/tagged, each port sends out a special BPDU containing the tagged information.

The generic action of a switch on receiving a BPDU is to compare the received BPDU to its own BPDU that it will transmit. If the priority of the received BPDU is better than its own priority, it will replace its BPDU with the received BPDU. Then, the switch adds its own bridge ID number and increments the path cost of the BPDU. The switch uses this information to block any necessary ports.

Note: If STP is globally disabled, BPDUs from external devices will transit the switch transparently. If STP is globally enabled, for ports where STP is turned off, inbound BPDUs will instead be discarded.

Determining the Path for Forwarding BPDUs

When determining which port to use for forwarding and which port to block, the G8296 uses information in the BPDU, including each bridge ID. A technique based on the “lowest root cost” is then computed to determine the most efficient path for forwarding.

Bridge Priority

The bridge priority parameter controls which bridge on the network is the STG root bridge. To make one switch become the root bridge, configure the bridge priority lower than all other switches and bridges on your network. The lower the value, the higher the bridge priority. Use the following command to configure the bridge priority:

```
RS8296(config)# spanning-tree stp <STG> bridge priority <0-65535>
```

Port Priority

The port priority helps determine which bridge port becomes the root port or the designated port. The case for the root port is when two switches are connected using a minimum of two links with the same path-cost. The case for the designated port is in a network topology that has multiple bridge ports with the same path-cost connected to a single segment, the port with the lowest port priority becomes the designated port for the segment. Use the following command to configure the port priority:

```
RS8296(config-if)# spanning-tree stp <STG> priority <port priority>
```

where *priority value* is a number from 0 to 240, in increments of 16 (such as 0, 16, 32, and so on). If the specified priority value is not evenly divisible by 16, the value will be automatically rounded down to the nearest valid increment whenever manually changed in the configuration, or whenever a configuration file from a release prior to N/OS 6.5 is loaded.

Root Guard

The root guard feature provides a way to enforce the root bridge placement in the network. It keeps a new device from becoming root and thereby forcing STP re-convergence. If a root-guard enabled port detects a root device, that port will be placed in a blocked state.

You can configure the root guard at the port level using the following commands:

```
RS8296(config)# interface port <port number>  
RS8296(config-if)# spanning-tree guard root
```

The default state is “none”, i.e. disabled.

Loop Guard

In general, STP resolves redundant network topologies into loop-free topologies. The loop guard feature performs additional checking to detect loops that might not be found using Spanning Tree. STP loop guard ensures that a non-designated port does not become a designated port.

To globally enable loop guard, enter the following command:

```
RS8296(config)# spanning-tree loopguard
```

Note: The global loop guard command will be effective on a port only if the port-level loop guard command is set to default as shown below:

```
RS8296(config)# interface port <port number>  
RS8296(config-if)# no spanning-tree guard
```

To enable loop guard at the port level, enter the following command:

```
RS8296(config)# interface port <port number>  
RS8296(config-if)# spanning-tree guard loop
```

The default state is “none”, meaning disabled.

Port Path Cost

The port path cost assigns lower values to high-bandwidth ports, such as 10 Gigabit Ethernet, to encourage their use. The objective is to use the fastest links so that the route with the lowest cost is chosen. A value of 0 (the default) indicates that the default cost will be computed for an auto-negotiated link or trunk speed.

Use the following command to modify the port path cost:

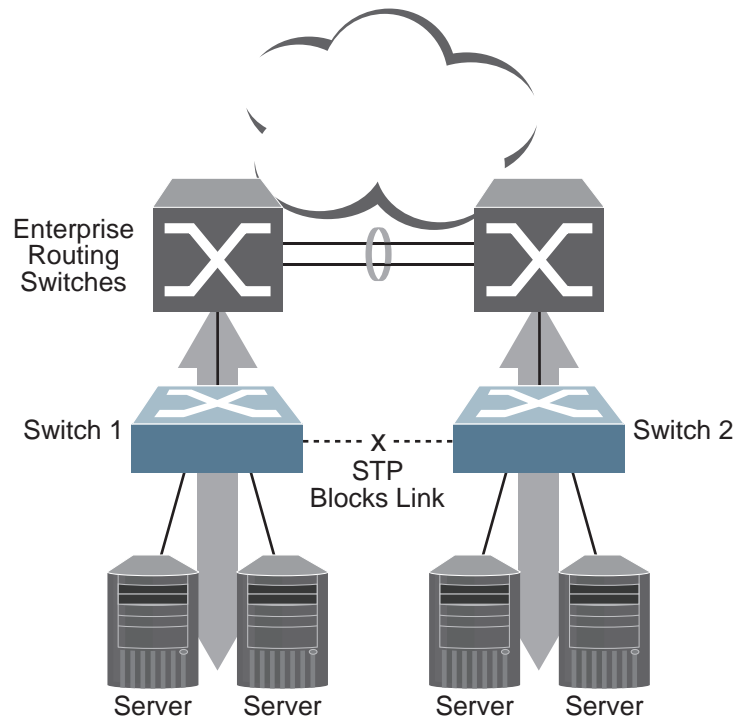
```
RS8296(config)# interface port <port number>  
RS8296(config-if)# spanning-tree stp <STG> path-cost <path cost value>  
RS8296(config-if)# exit
```

The port path cost can be a value from 1 to 200000000. Specify 0 for automatic path cost.

Simple STP Configuration

Figure 10 depicts a simple topology using a switch-to-switch link between two G8296 1 and 2.

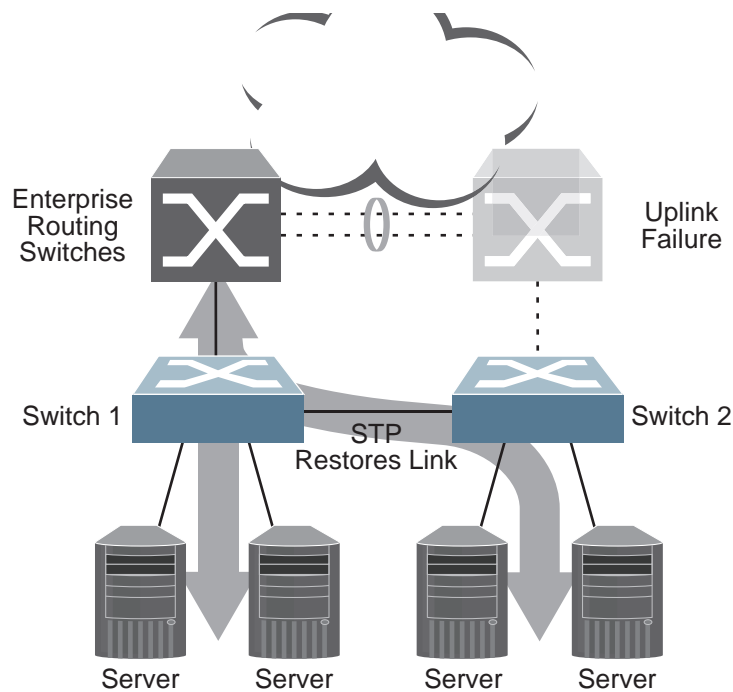
Figure 10. Spanning Tree Blocking a Switch-to-Switch Link



To prevent a network loop among the switches, STP must block one of the links between them. In this case, it is desired that STP block the link between the Lenovo switches, and not one of the G8296 uplinks or the Enterprise switch trunk.

During operation, if one G8296 experiences an uplink failure, STP will activate the Lenovo switch-to-switch link so that server traffic on the affected G8296 may pass through to the active uplink on the other G8296, as shown in Figure 11.

Figure 11. Spanning Tree Restoring the Switch-to-Switch Link



In this example, port 10 on each G8296 is used for the switch-to-switch link. To ensure that the G8296 switch-to-switch link is blocked during normal operation, the port path cost is set to a higher value than other paths in the network. To configure the port path cost on the switch-to-switch links in this example, use the following commands on each G8296.

```
RS8296(config)# interface port 10
RS8296(config-if)# spanning-tree stp 1 path-cost 60000
RS8296(config-if)# exit
```


Per-VLAN Spanning Tree Groups

PVRST mode supports a maximum of 128 STGs, with each STG acting as an independent, simultaneous instance of STP.

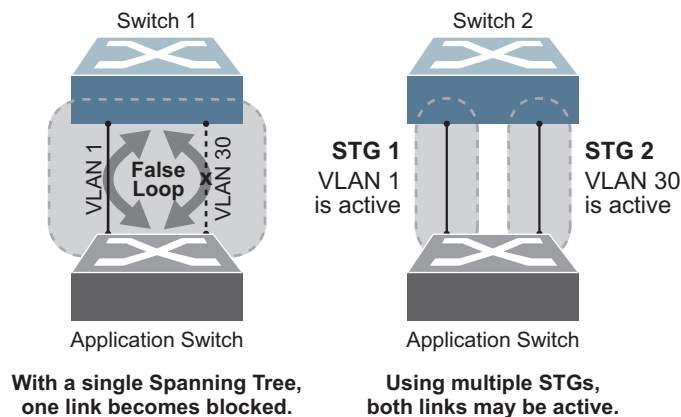
Multiple STGs provide multiple data paths which can be used for load-balancing and redundancy. To enable load balancing between two G8296s using multiple STGs, configure each path with a different VLAN and then assign each VLAN to a separate STG. Since each STG is independent, they each send their own IEEE 802.1Q tagged Bridge Protocol Data Units (BPDUs).

Each STG behaves as a bridge group and forms a loop-free topology. The default STG 1 may contain multiple VLANs (typically until they can be assigned to another STG). STGs 2-127 may contain only one VLAN each.

Using Multiple STGs to Eliminate False Loops

Figure 12 shows a simple example of why multiple STGs are needed. In the figure, two ports on a G8296 are connected to two ports on an application switch. Each of the links is configured for a different VLAN, preventing a network loop. However, in the first network, since a single instance of Spanning Tree is running on all the ports of the G8296, a physical loop is assumed to exist, and one of the VLANs is blocked, impacting connectivity even though no actual loop exists.

Figure 12. Using Multiple Instances of Spanning Tree Group



In the second network, the problem of improper link blocking is resolved when the VLANs are placed into different Spanning Tree Groups (STGs). Since each STG has its own independent instance of Spanning Tree, each STG is responsible only for the loops within its own VLAN. This eliminates the false loop, and allows both VLANs to forward packets between the switches at the same time.

VLANs and STG Assignment

In PVRST mode, up to 128 STGs are supported. Ports cannot be added directly to an STG. Instead, ports must be added as members of a VLAN, and the VLAN must then be assigned to the STG.

STG 1 is the default STG. Although VLANs can be added to or deleted from default STG 1, the STG itself cannot be deleted from the system. By default, STG 1 is enabled and includes VLAN 1, which by default includes all switch ports (except for management VLANs and management ports).

STG 128 is reserved for switch management. By default, STG 128 is disabled, but includes management VLAN 4095 and the management ports (MGMT-A and MGMT-B).

By default, all other STGs (STG 2 through 127) are enabled, though they initially include no member VLANs. VLANs must be assigned to STGs. By default, this is done automatically using VLAN Automatic STG Assignment (VASA), though it can also be done manually (see [“Manually Assigning STGs” on page 154](#)).

When VASA is enabled (as by default), each time a new VLAN is configured, the switch will automatically assign that new VLAN to its own STG. Conversely, when a VLAN is deleted, if its STG is not associated with any other VLAN, the STG is returned to the available pool.

The specific STG number to which the VLAN is assigned is based on the VLAN number itself. For low VLAN numbers (1 through 127), the switch will attempt to assign the VLAN to its matching STG number. For higher numbered VLANs, the STG assignment is based on a simple modulus calculation; the attempted STG number will “wrap around,” starting back at the top of STG list each time the end of the list is reached. However, if the attempted STG is already in use, the switch will select the next available STG. If an empty STG is not available when creating a new VLAN, the VLAN is automatically assigned to default STG 1.

If ports are tagged, each tagged port sends out a special BPDU containing the tagged information. Also, when a tagged port belongs to more than one STG, the egress BPDUs are tagged to distinguish the BPDUs of one STG from those of another STG.

VASA is enabled by default, but can be disabled or re-enabled using the following commands:

```
RS8296(config)# [no] spanning-tree stg-auto
```

If VASA is disabled, when you create a new VLAN, that VLAN automatically belongs to default STG 1. To place the VLAN in a different STG, assign it manually.

VASA applies only to PVRST mode and is ignored in RSTP and MSTP modes.

Manually Assigning STGs

The administrator may manually assign VLANs to specific STGs, whether or not VASA is enabled.

1. If no VLANs exist (other than default VLAN 1), see [“Guidelines for Creating VLANs” on page 155](#) for information about creating VLANs and assigning ports to them.

2. Assign the VLAN to an STG using one of the following methods:
 - o From the global configuration mode:

```
RS8296(config)# spanning-tree stp <STG number> vlan <VLAN>
```

- o Or from within the VLAN configuration mode:

```
RS8296(config)# vlan <VLAN numbers>
RS8296(config-vlan)# stg <STG number>
RS8296(config-vlan)# exit
```

When a VLAN is assigned to a new STG, the VLAN is automatically removed from its prior STG.

Note: For proper operation with switches that use Cisco PVST+, it is recommended that you create a separate STG for each VLAN.

Guidelines for Creating VLANs

Follow these guidelines when creating VLANs:

- When you create a new VLAN, if VASA is enabled (the default), that VLAN is automatically assigned its own STG. If VASA is disabled, the VLAN automatically belongs to STG 1, the default STG. To place the VLAN in a different STG, see [“Manually Assigning STGs” on page 154](#). The VLAN is automatically removed from its old STG before being placed into the new STG.
- Each VLANs must be contained *within* a single STG; a VLAN cannot span multiple STGs. By confining VLANs within a single STG, you avoid problems with Spanning Tree blocking ports and causing a loss of connectivity within the VLAN. When a VLAN spans multiple switches, it is recommended that the VLAN remain within the same STG (be assigned the same STG ID) across all the switches.
- If ports are tagged, all trunked ports can belong to multiple STGs.
- A port cannot be directly added to an STG. The port must first be added to a VLAN, and that VLAN added to the desired STG.

Rules for VLAN Tagged/Trunk Mode Ports

The following rules apply to VLAN tagged ports:

- Tagged/trunk mode ports can belong to more than one STG, but untagged/access mode ports can belong to only one STG.
- When a tagged/trunk mode port belongs to more than one STG, the egress BPDUs are tagged to distinguish the BPDUs of one STG from those of another STG.

Adding and Removing Ports from STGs

The following rules apply when you add ports to or remove ports from STGs:

- When you add a port to a VLAN that belongs to an STG, the port is also added to that STG. However, if the port you are adding is an untagged port and is already a member of another STG, that port will be removed from its current STG and added to the new STG. An untagged port cannot belong to more than one STG.

For example: Assume that VLAN 1 belongs to STG 1, and that port 1 is untagged and does not belong to any STG. When you add port 1 to VLAN 1, port 1 will automatically become part of STG 1.

However, if port 5 is untagged and is a member of VLAN 3 in STG 2, then adding port 5 to VLAN 1 in STG 1 will change the port PVID from 3 to 1:

"Port 5 is an UNTAGGED/Access Mode port and its PVID/Native-VLAN changed from 3 to 1.

- When you remove a port from VLAN that belongs to an STG, that port will also be removed from the STG. However, if that port belongs to another VLAN in the same STG, the port remains in the STG.

As an example, assume that port 2 belongs to only VLAN 2, and that VLAN 2 belongs to STG 2. When you remove port 2 from VLAN 2, the port is moved to default VLAN 1 and is removed from STG 2.

However, if port 2 belongs to both VLAN 1 and VLAN 2, and both VLANs belong to STG 2, removing port 2 from VLAN 2 does not remove port 2 from STG 2, because the port is still a member of VLAN 1, which is still a member of STG 1.

- An STG cannot be deleted, only disabled. If you disable the STG while it still contains VLAN members, Spanning Tree will be off on all ports belonging to that VLAN.

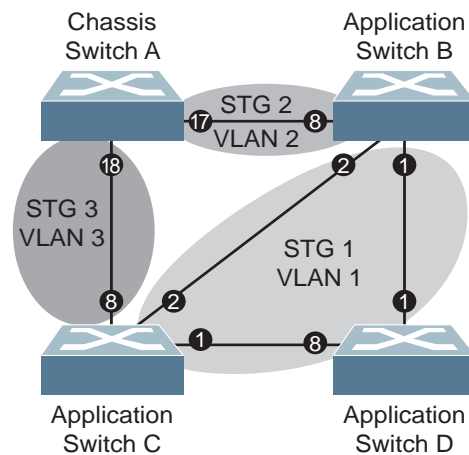
The relationship between port, trunk groups, VLANs, and Spanning Trees is shown in [Table 18 on page 147](#).

The Switch-Centric Model

PVRST is switch-centric: STGs are enforced only on the switch where they are configured. PVRST allows only one VLAN per STG, except for the default STG 1 to which multiple VLANs can be assigned. The STG ID is not transmitted in the Spanning Tree BPDU. Each Spanning Tree decision is based entirely on the configuration of the particular switch.

For example, in [Figure 13](#), each switch is responsible for the proper configuration of its own ports, VLANs, and STGs. Switch A identifies its own port 17 as part of VLAN 2 on STG 2, and the Switch B identifies its own port 8 as part of VLAN 2 on STG 2.

Figure 13. Implementing Multiple Spanning Tree Groups



The VLAN participation for each Spanning Tree Group in [Figure 13 on page 157](#) is as follows:

- VLAN 1 Participation

Assuming Switch B to be the root bridge, Switch B transmits the BPDU for STG 1 on ports 1 and 2. Switch C receives the BPDU on port 2, and Switch D receives the BPDU on port 1. Because there is a network loop between the switches in VLAN 1, either Switch D will block port 8 or Switch C will block port 1, depending on the information provided in the BPDU.

- VLAN 2 Participation

Switch B, the root bridge, generates a BPDU for STG 2 from port 8. Switch A receives this BPDU on port 17, which is assigned to VLAN 2, STG 2. Because switch B has no additional ports participating in STG 2, this BPDU is not forwarded to any additional ports and Switch B remains the designated root.

- VLAN 3 Participation

For VLAN 3, Switch A or Switch C may be the root bridge. If Switch A is the root bridge for VLAN 3, STG 3, then Switch A transmits the BPDU from port 18. Switch C receives this BPDU on port 8 and is identified as participating in VLAN 3, STG 3. Since Switch C has no additional ports participating in STG 3, this BPDU is not forwarded to any additional ports and Switch A remains the designated root.

Configuring Multiple STGs

This configuration shows how to configure the three instances of STGs on the switches A, B, C, and D illustrated in [Figure 13 on page 157](#).

Because VASA is enabled by default, each new VLAN is automatically assigned its own STG.

1. Set the Spanning Tree mode on each switch to PVRST.

```
RS8296(config)# spanning-tree mode pvrst
```

Note: PVRST is the default mode on the G8296. This step is not required unless the STP mode has been previously changed, and is shown here merely as an example of manual configuration.

2. Configure the following on Switch A:

Enable VLAN 2 and VLAN 3.

```
RS8296(config)# vlan 2
RS8296(config-vlan)# exit
RS8296(config)# vlan 3
RS8296(config-vlan)# exit
```

If VASA is disabled, enter the following commands:

```
RS8296(config)# spanning-tree stp 2 vlan 2
RS8296(config)# spanning-tree stp 3 vlan 3
```

Add port 17 to VLAN 2, port 18 to VLAN 3.

```
RS8296(config)# interface port 17
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 2
RS8296(config-if)# exit

RS8296(config)# interface port 18
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 3
RS8296(config-if)# exit
```

VLAN 2 and VLAN 3 are removed from STG 1.

Note: In PVRST mode, each instance of STG is enabled by default.

3. Configure the following on Switch B:

Add port 8 to VLAN 2. Ports 1 and 2 are by default in VLAN 1 assigned to STG 1.

```
RS8296(config)# vlan 2
RS8296(config-vlan)# stg 2
RS8296(config-vlan)# exit
RS8296(config)# interface port 8
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 2
RS8296(config-if)# exit
```

If VASA is disabled, enter the following command:

```
RS8296(config)# spanning-tree stp 2 vlan 2
```

VLAN 2 is automatically removed from STG 1. By default VLAN 1 remains in STG 1.

4. Configure the following on application switch C:

Add port 8 to VLAN 3. Ports 1 and 2 are by default in VLAN 1 assigned to STG 1.

```
RS8296(config)# vlan 3
RS8296(config-vlan)# stg 3
RS8296(config-vlan)# exit
RS8296(config)# interface port 8
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 3
RS8296(config-if)# exit
```

If VASA is disabled, enter the following command:
RS8296(config)# **spanning-tree stp 3 vlan 3**

VLAN 3 is automatically removed from STG 1. By default VLAN 1 remains in STG 1.

5. Switch D does not require any special configuration for multiple Spanning Trees. Switch D uses default STG 1 only.

Rapid Spanning Tree Protocol

RSTP provides rapid convergence of the Spanning Tree and provides the fast re-configuration critical for networks carrying delay-sensitive traffic such as voice and video. RSTP significantly reduces the time to reconfigure the active topology of the network when changes occur to the physical topology or its configuration parameters. RSTP reduces the bridged-LAN topology to a single Spanning Tree.

RSTP was originally defined in IEEE 802.1w (2001) and was later incorporated into IEEE 802.1D (2004), superseding the original STP standard.

RSTP parameters apply only to Spanning Tree Group (STG) 1. The PVRST mode STGs 2-128 are not used when the switch is placed in RSTP mode.

RSTP is compatible with devices that run IEEE 802.1D (1998) Spanning Tree Protocol. If the switch detects IEEE 802.1D (1998) BPDUs, it responds with IEEE 802.1D (1998)-compatible data units. RSTP is not compatible with Per-VLAN Rapid Spanning Tree (PVRST) protocol.

Port States

RSTP port state controls are the same as for PVRST: discarding, learning, and forwarding.

Due to the sequence involved in these STP states, considerable delays may occur while paths are being resolved. To mitigate delays, ports defined as *edge/portfast* ports ([“Port Type and Link Type” on page 165](#)) may bypass the discarding and learning states, and enter directly into the forwarding state.

RSTP Configuration Guidelines

This section provides important information about configuring RSTP. When RSTP is turned on, the following occurs:

- STP parameters apply only to STG 1.
- Only STG 1 is available. All other STGs are turned off.
- All VLANs, including management VLANs, are moved to STG 1.

RSTP Configuration Example

This section provides steps to configure RSTP.

1. Configure port and VLAN membership on the switch.
2. Set the Spanning Tree mode to Rapid Spanning Tree.

```
RS8296(config)# spanning-tree mode rstp
```

3. Configure RSTP parameters.

```
RS8296(config)# spanning-tree stp 1 bridge priority 8192
RS8296(config)# spanning-tree stp 1 bridge hello-time 5
RS8296(config)# spanning-tree stp 1 bridge forward-delay 20
RS8296(config)# spanning-tree stp 1 bridge maximum-age 30
RS8296(config)# no spanning-tree stp 1 enable
```


4. Configure port parameters:

```
RS8296(config)# interface port 3
RS8296(config-if)# spanning-tree stp 1 priority 240
RS8296(config-if)# spanning-tree stp 1 path-cost 500
RS8296(config-if)# no spanning-tree stp 1 enable
RS8296(config-if)# exit
```

Multiple Spanning Tree Protocol

Multiple Spanning Tree Protocol (MSTP) extends Rapid Spanning Tree Protocol (RSTP), allowing multiple Spanning Tree Groups (STGs) which may each include multiple VLANs. MSTP was originally defined in IEEE 802.1s (2002) and was later included in IEEE 802.1Q (2003).

In MSTP mode, the G8296 supports up to 32 instances of Spanning Tree, corresponding to STGs 1-32, with each STG acting as an independent, simultaneous instance of RSTP.

MSTP allows frames assigned to different VLANs to follow separate paths, with each path based on an independent Spanning Tree instance. This approach provides multiple forwarding paths for data traffic, thereby enabling load-balancing, and reducing the number of Spanning Tree instances required to support a large number of VLANs.

Due to Spanning Tree's sequence of discarding, learning, and forwarding, lengthy delays may occur while paths are being resolved. Ports defined as *edge/portfast* ports ("[Port Type and Link Type](#)" on page 165) bypass the Discarding and Learning states, and enter directly into the Forwarding state.

Note: In MSTP mode, Spanning Tree for the management ports is turned off by default.

MSTP Region

A group of interconnected bridges that share the same attributes is called an MST region. Each bridge within the region must share the following attributes:

- Alphanumeric name
- Revision number
- VLAN-to STG mapping scheme

MSTP provides rapid re-configuration, scalability and control due to the support of regions, and multiple Spanning-Tree instances support within each region.

Common Internal Spanning Tree

The Common Internal Spanning Tree (CIST) or MST0 provides a common form of Spanning Tree Protocol, with one Spanning-Tree instance that can be used throughout the MSTP region. CIST allows the switch to interoperate with legacy equipment, including devices that run IEEE 802.1D (1998) STP.

CIST allows the MSTP region to act as a virtual bridge to other bridges outside of the region, and provides a single Spanning-Tree instance to interact with them.

CIST port configuration includes Hello time, path-cost, and interface priority. These parameters do not affect Spanning Tree Groups 1-32. They apply only when the CIST is used.

MSTP Configuration Guidelines

This section provides important information about configuring Multiple Spanning Tree Groups:

- When MSTP is turned on, the switch moves VLAN 1 and VLANs 33-128 to the CIST. When MSTP is turned off, the switch moves VLAN 1 and VLANs 33-128 from the CIST to STG 1.
- When you enable MSTP, you must configure the Region Name. A default version number of 1 is configured automatically.
- Each bridge in the region must have the same name, revision number, and VLAN mapping.

MSTP Configuration Examples

Example 1

This section provides steps to configure MSTP on the G8296.

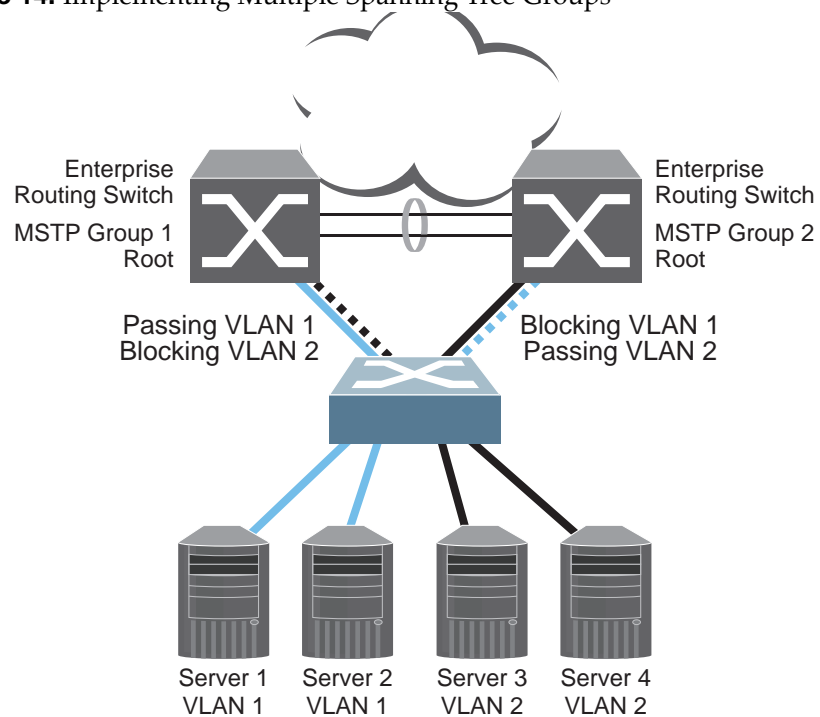
1. Configure port and VLAN membership on the switch.
2. Configure Multiple Spanning Tree region parameters, and set the mode to MSTP.
3. Map VLANs to MSTP instances:

```
RS8296(config)# spanning-tree mst configuration( Enter MST configuration mode)
RS8296(config-mst)# instance <instance ID> vlan <vlan number or range>
```

Example 2

This configuration shows how to configure MSTP Groups on the switch, as shown in [Figure 13](#).

Figure 14. Implementing Multiple Spanning Tree Groups



This example shows how multiple Spanning Trees can provide redundancy without wasting any uplink ports. In this example, the server ports are split between two separate VLANs. Both VLANs belong to two different MSTP groups. The Spanning Tree *priority* values are configured so that each routing switch is the root for a different MSTP instance. All of the uplinks are active, with each uplink port backing up the other.

1. Configure port membership and define the STGs for VLAN 1. Enable tagging on uplink ports that share VLANs. Port 19 and port 20 connect to the Enterprise Routing switches.

```
RS8296(config)# interface port 19, 20
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# exit
```

2. Configure MSTP: Spanning Tree mode, region name, and version.

```
RS8296(config)# spanning-tree mst configuration
RS8296(config-mst)# name MyRegion (Define the Region name)
RS8296(config-mst)# revision 100 (Define the Revision level)
RS8296(config-mst)# exit
RS8296(config)# spanning-tree mode mst (Set mode to Multiple Spanning Trees)
```

3. Map VLANs to MSTP instances:

```
RS8296(config)# spanning-tree mst configuration
RS8296(config-mst)# instance 1 vlan 1
RS8296(config-mst)# instance 2 vlan 2
```

4. Add server ports 1 and 2 to VLAN 1. Add uplink ports 19 and port 20 to VLAN 1.

```
RS8296(config)# interface port 1,2,19,20
RS8296(config-if)# switchport trunk allowed vlan add 1
RS8296(config-if)# exit
```

5. Configure port membership and define the STGs for VLAN 2. Add server ports 3, 4, and 5 to VLAN 2. Add uplink ports 19 and 20 to VLAN 2. Assign VLAN 2 to STG 2.

```
RS8296(config)# interface port 3,4,5,19,20
RS8296(config-if)# switchport trunk allowed vlan add 2
RS8296(config-if)# exit
```

Note: Each STG is enabled by default.

Port Type and Link Type

Edge/Portfast Port

A port that does not connect to a bridge is called an *edge port*. Since edge ports are assumed to be connected to non-STP devices (such as directly to hosts or servers), they are placed in the forwarding state as soon as the link is up.

Edge ports send BPDUs to upstream STP devices like normal STP ports, but do not receive BPDUs. If a port with `edge` enabled does receive a BPDU, it immediately begins working as a normal (non-edge) port, and participates fully in Spanning Tree.

Use the following commands to define or clear a port as an edge port:

```
RS8296(config)# interface port <port>
RS8296(config-if)# [no] spanning-tree portfast
RS8296(config-if)# exit
```

Link Type

The link type determines how the port behaves in regard to Rapid Spanning Tree. Use the following commands to define the link type for the port:

```
RS8296(config)# interface port <port>
RS8296(config-if)# [no] spanning-tree link-type <type>
RS8296(config-if)# exit
```

where *type* corresponds to the duplex mode of the port, as follows:

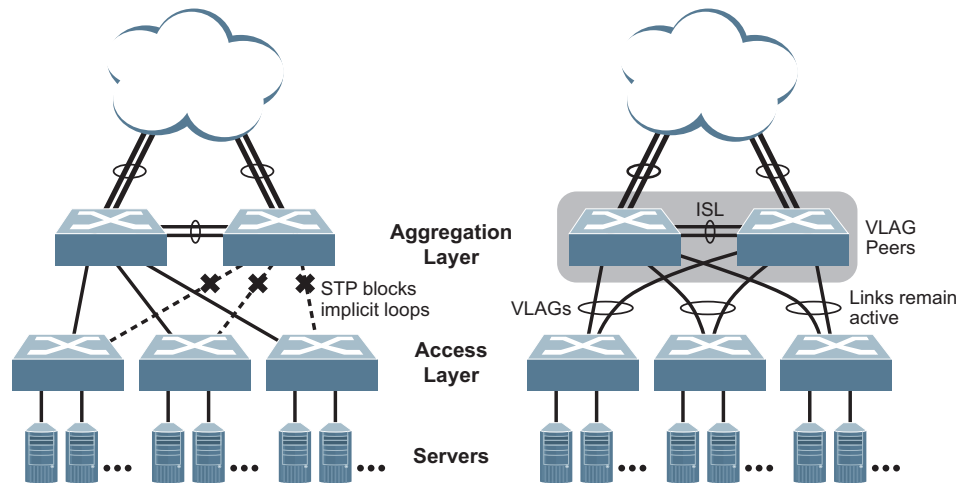
- `p2p` A full-duplex link to another device (point-to-point)
- `shared` A half-duplex link is a shared segment and can contain more than one device.
- `auto` The switch dynamically configures the link type.

Note: Any STP port in full-duplex mode can be manually configured as a shared port when connected to a non-STP-aware shared device (such as a typical Layer 2 switch) used to interconnect multiple STP-aware devices.

Chapter 12. Virtual Link Aggregation Groups

In many data center environments, downstream servers or switches connect to upstream devices which consolidate traffic. For example, see [Figure 15](#).

Figure 15. Typical Data Center Switching Layers with STP vs. VLAG



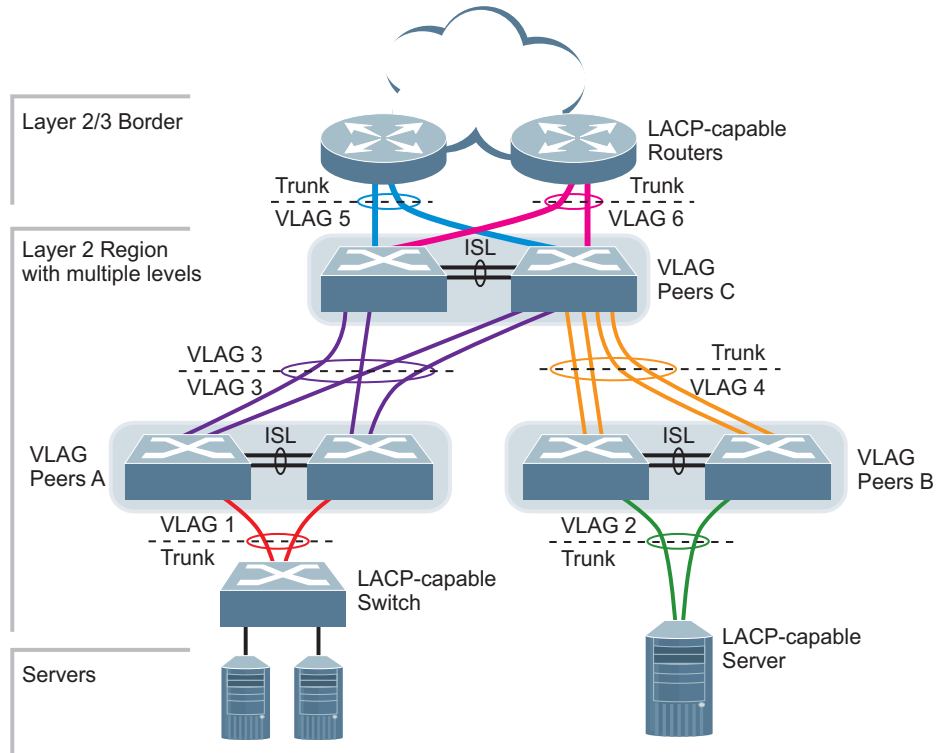
As shown in the example, a switch in the access layer may be connected to more than one switch in the aggregation layer to provide for network redundancy. Typically, Spanning Tree Protocol (RSTP, PVRST, or MSTP—see [“Spanning Tree Protocols” on page 145](#)) is used to prevent broadcast loops, blocking redundant uplink paths. This has the unwanted consequence of reducing the available bandwidth between the layers by as much as 50%. In addition, STP may be slow to resolve topology changes that occur during a link failure, and can result in considerable MAC address flooding.

Using Virtual Link Aggregation Groups (VLAGs), the redundant uplinks remain active, utilizing all available bandwidth.

Two switches are paired into VLAG peers, and act as a single virtual entity for the purpose of establishing a multi-port trunk. Ports from both peers can be grouped into a VLAG and connected to the same LAG-capable target device. From the perspective of the target device, the ports connected to the VLAG peers appear to be a single trunk connecting to a single logical device. The target device uses the configured Tier ID to identify the VLAG peers as this single logical device. It is important that you use a unique Tier ID for each VLAG pair you configure. The VLAG-capable switches synchronize their logical view of the access layer port structure and internally prevent implicit loops. The VLAG topology also responds more quickly to link failure and does not result in unnecessary MAC flooding.

VLAGs are also useful in multi-layer environments for both uplink and downlink redundancy to any regular LAG-capable device. For example:

Figure 16. VLAG Application with Multiple Layers



Wherever ports from *both* peered switches are trunked to another device, the trunked ports must be configured as a VLAG. For example, VLAGs 1 and 3 must be configured for both VLAG Peer A switches. VLAGs 2 and 4 must be configured for both VLAG Peer B switches. VLAGs 3, 5, and 6 must be configured on both VLAG Peer C switches. Other devices connecting to the VLAG peers are configured using regular static or dynamic trunks.

Note: Do not configure a VLAG for connecting only one switch in the peer set to another device or peer set. For instance, in VLAG Peer C, a regular trunk is employed for the downlink connection to VLAG Peer B because only one of the VLAG Peer C switches is involved.

VLAG Capacities

Servers or switches that connect to the VLAG peers using a multi-port VLAG are considered VLAG clients. VLAG clients are not required to be VLAG-capable. The ports participating in the VLAG are configured as regular port trunks on the VLAG client end.

On the VLAG peers, the VLAGs are configured similarly to regular port trunks, using many of the same features and rules. See [“Ports and Trunking” on page 131](#) for general information concerning all port trunks.

Each VLAG begins as a regular port trunk on each VLAG-peer switch. The VLAG may be either a static trunk group (portchannel) or dynamic LACP trunk group, and consumes one slot from the overall port trunk capacity pool. The trunk type must match that used on VLAG client devices. Additional configuration is then required to implement the VLAG on both VLAG peer switches.

You may configure up to 102 trunk groups on the switch, with all types (regular or VLAG, static or LACP) sharing the same pool.

The maximum number of configurable VLAG instances is as follows:

- **With STP off:** Maximum of 52 VLAG instances
- **With STP on:**
 - **PVRST/MSTP with one VLAG instance per VLAN/STG:** Maximum of 52 VLAG instances
 - **PVRST/MSTP with one VLAG instance belonging to multiple VLANs/STGs:** Maximum of 20 VLAG instances

Each trunk type can contain up to 32 member ports, depending on the port type and availability.

VLAGs versus Port Trunks

Though similar to regular port trunks in many regards, VLAGs differ from regular port trunks in a number of important ways:

- A VLAG can consist of multiple ports on two VLAG peers, which are connected to one logical client device such as a server, switch, or another VLAG device.
- The participating ports on the client device are configured as a regular port trunk.
- The VLAG peers must be the same model, and run the same software version.
- VLAG peers require a dedicated inter-switch link (ISL) for synchronization. The ports used to create the ISL must have the following properties:
 - ISL ports must have VLAN tagging turned on.
 - ISL ports must be configured for all VLAG VLANs.
 - ISL ports must be placed into a regular port trunk group (dynamic or static).
 - A minimum of two ports on each switch are recommended for ISL use.
- Dynamic routing protocols, such as OSPF, cannot terminate on VLAGs.

- Routing over VLAGs is not supported. However, IP forwarding between subnets served by VLAGs can be accomplished using VRRP.
- VLAGs are configured using additional commands.
- It is recommended that end-devices connected to VLAG switches use NICs with dual-homing. This increases traffic efficiency, reduces ISL load, and provides faster link failover.

Configuring VLAGs

When configuring VLAG or making changes to your VLAG configuration, consider the following VLAG behavior:

- When adding a static Mrouter on VLAG links, ensure that you also add it on the ISL link to avoid VLAG link failure. If the VLAG link fails, traffic cannot be recovered through the ISL.
- When you enable VLAG on the switch, if a MSTP region mismatch is detected with the VLAG peer, the ISL will shut down. In such a scenario, correct the region on the VLAG peer and manually enable the ISL.
- If you have enabled VLAG on the switch, and you need to change the STP mode, ensure that you first disable VLAG and then change the STP mode.
- When VLAG is enabled, you may see two root ports on the secondary VLAG switch. One of these will be the actual root port for the secondary VLAG switch and the other will be a root port synced with the primary VLAG switch.
- The LACP key used must be unique for each VLAG in the entire topology.
- The STG to VLAN mapping on both VLAG peers must be identical.

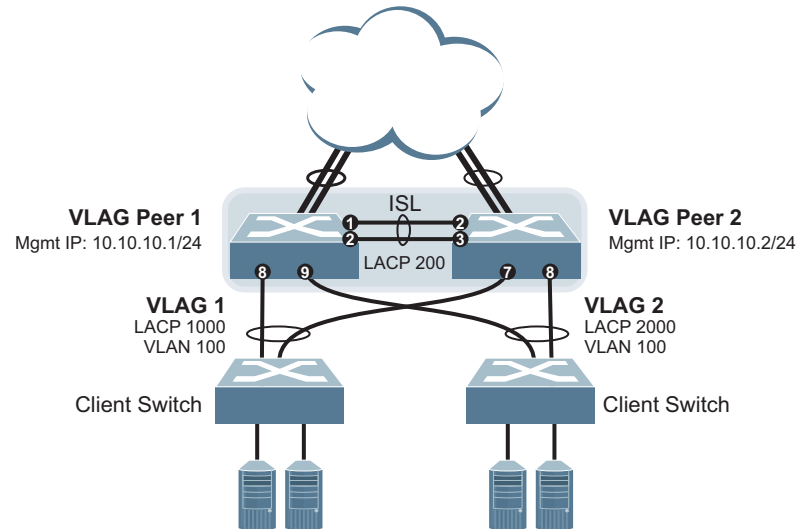
The following parameters must be identically configured on the VLAG ports of both the VLAG peers:

- VLANs
- Native VLAN tagging
- STP mode
- BPDU Guard setting
- STP port setting
- MAC aging timers
- Static MAC entries
- ACL configuration parameters
- QoS configuration parameters

Basic VLAG Configuration

Figure 18 shows an example configuration where two VLAG peers are used for aggregating traffic from downstream devices.

Figure 18. Basic VLAGs



In this example, each client switch is connected to both VLAG peers. On each client switch, the ports connecting to the VLAG peers are configured as a dynamic LACP port trunk. The VLAG peer switches share a dedicated ISL for synchronizing VLAG information. On the individual VLAG peers, each port leading to a specific client switch (and part of the client switch's port trunk) is configured as a VLAG.

In the following example configuration, only the configuration for VLAG 1 on VLAG Peer 1 is shown. VLAG Peer 2 and all other VLAGs are configured in a similar fashion.

Configure the ISL

The ISL connecting the VLAG peers is shared by all their VLAGs. The ISL needs to be configured only once on each VLAG peer.

1. Configure STP if required. Use PVRST or MSTP mode only:

```
RS8296(config)# spanning-tree mode pvrst
```

2. Configure the ISL ports and place them into a port trunk group:

```
RS8296(config)# interface port 1-2
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# lacp mode active
RS8296(config-if)# lacp key 200
RS8296(config-if)# exit
RS8296(config)# vlag isl adminkey 200
```

Note: In this case, a dynamic trunk group is shown. A static trunk (portchannel) could be configured instead.

3. Configure VLAG Tier ID. This is used to identify the VLAG switch in a multi-tier environment.

```
RS8296(config)# vlag tier-id 10
```

4. Configure the ISL for the VLAG peer.

Make sure you configure the VLAG peer (VLAG Peer 2) using the same ISL trunk type (dynamic or static), the same VLAN, and the same STP mode and tier ID used on VLAG Peer 1.

5. Enable VLAG globally.

```
RS8296(config)# vlag enable
```

Configure the VLAG

1. Configure the VLAN for VLAG 1. Make sure members include the ISL and VLAG 1 ports.

```
RS8296(config)# vlan 100
RS8296(config-vlan)# exit
RS8296(config)# interface port 1-2,8
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 100
RS8296(config-if)# exit
```

Note: In MSTP mode, VLANs are automatically mapped to CIST.

2. Place the VLAG 1 port(s) in a port trunk group:

```
RS8296(config)# interface port 8
RS8296(config-if)# lacp mode active
RS8296(config-if)# lacp key 1000
RS8296(config-if)# exit
```

3. Assign the trunk to the VLAG:

```
RS8296(config)# vlag adminkey 1000 enable
```

4. Continue by configuring all required VLAGs on VLAG Peer 1, and then repeat the configuration for VLAG Peer 2.

For each corresponding VLAG on the peer, the port trunk type (dynamic or static), VLAN, and STP mode and ID must be the same as on VLAG Peer 1.

5. Verify the completed configuration:

```
# show vlag
```

VLAG Configuration - VLANs Mapped to MSTI

Follow the steps below to configure VLAG in environments where the VLANs are mapped to Multiple Spanning Tree Instance (MSTI).

Configure the ISL

The ISL connecting the VLAG peers is shared by all their VLAGs. The ISL needs to be configured only once on each VLAG peer.

1. Configure STP:

```
RS8296(config)# spanning-tree mode mst
```

2. Configure the ISL ports and place them into a port trunk group:

```
RS8296(config)# interface port 1-2
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# lacp mode active
RS8296(config-if)# lacp key 200
RS8296(config-if)# exit
RS8296(config)# vlag isl adminkey 200
```

Note: In this case, a dynamic trunk group is shown. A static trunk (portchannel) could be configured instead.

3. Configure VLAG Tier ID. This is used to identify the VLAG switch in a multi-tier environment.

```
RS8296(config)# vlag tier-id 10
```

4. Configure the ISL for the VLAG peer.

Make sure you configure the VLAG peer (VLAG Peer 2) using the same ISL trunk type (dynamic or static), the same VLAN, and the same STP mode and tier ID used on VLAG Peer 1.

5. Enable VLAG globally.

```
RS8296(config)# vlag enable
```

Configure the VLAG

1. Configure the VLAN for VLAG 1.

```
RS8296(config)# vlan 100
RS8296(config-vlan)# exit
```

Note: The VLAN is automatically mapped to CIST.

2. Shut down ISL ports and VLAG ports. This ensures that traffic passes through to other switches.

```
RS8296(config)# interface port 1-2,8
RS8296(config-if)# shutdown
```

3. Map the VLAN to an MSTI.

```
RS8296(config)# spanning-tree mst 1 vlan 100
```

4. Disable VLAG.

```
RS8296(config)# no vlag enable
```

5. Add VLAG ports to the VLAN.

```
RS8296(config)# interface port 1-2,8
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan 100
RS8296(config-if)# exit
```

6. Enable VLAG.

```
RS8296(config)# vlag enable
```

7. Place the VLAG 1 port(s) in a port trunk group:

```
RS8296(config)# interface port 8
RS8296(config-if)# lacp mode active
RS8296(config-if)# lacp key 1000
RS8296(config-if)# exit
```

8. Assign the trunk to the VLAG:

```
RS8296(config)# vlag adminkey 1000 enable
```

9. Continue by configuring all required VLAGs on VLAG Peer 1, then follow the steps for configuring VLAG Peer 2.

10. Configure the VLAN for VLAG 2.

```
RS8296(config)# vlan 100
RS8296(config-vlan)# exit
```

Note: The VLAN is automatically mapped to CIST.

11. Map the VLAN to an MSTI.

```
RS8296(config)# spanning-tree mst 1 vlan 100
```

Note: At this point, traffic may be momentarily disrupted due to STP recalculation.

12. Disable VLAG.

```
RS8296(config)# no vlag enable
```

Note: At this point, traffic may be momentarily disrupted due to STP recalculation.

13. Add VLAG ports to the VLAN.

```
RS8296(config)# interface port 1-2,8
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan 100
RS8296(config-if)# exit
```

14. Enable VLAG.

```
RS8296(config)# vlag enable
```

Note: At this point, traffic may be momentarily disrupted due to STP recalculation.

On VLAG Peer 1:

15. Enable ISL ports and VLAG ports:

```
RS8296(config)# interface port 1-2,8
RS8296(config-if)# no shutdown
```

On VLAG Peer2:

16. Place the VLAG 2 port(s) in a port trunk group:

```
RS8296(config)# interface port 8
RS8296(config-if)# lacp mode active
RS8296(config-if)# lacp key 1000
RS8296(config-if)# exit
```

17. Assign the trunk to the VLAG:

```
RS8296(config)# vlag adminkey 1000 enable
```

18. Verify the completed configuration:

```
RS8296# show vlag
```

Configuring Health Check

We strongly recommend that you configure the G8296 to check the health status of its VLAG peer. Although the operational status of the VLAG peer is generally determined via the ISL connection, configuring a network health check provides an alternate means to check peer status in case the ISL links fail. Use an independent link between the VLAG switches to configure health check.

Note: Configuring health check on an ISL VLAN interface or on a VLAG data port may impact the accuracy of the health check status.

1. Configure a management interface for the switch.

Note: If the switch does not have a dedicated management interface, configure a VLAN for the health check interface. The health check interface can be configured with an IPv4 or IPv6 address:

```
RS8296(config)# interface ip 128
RS8296(config-ip-if)# ip address 10.10.10.1 255.255.255.0
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
```

Note: Configure a similar interface on VLAG Peer 2. For example, use IP address 10.10.10.2.

2. Specify the IPv4 or IPv6 address of the VLAG Peer:

```
RS8296(config)# vlag h1thchk peer-ip 10.10.10.2
```

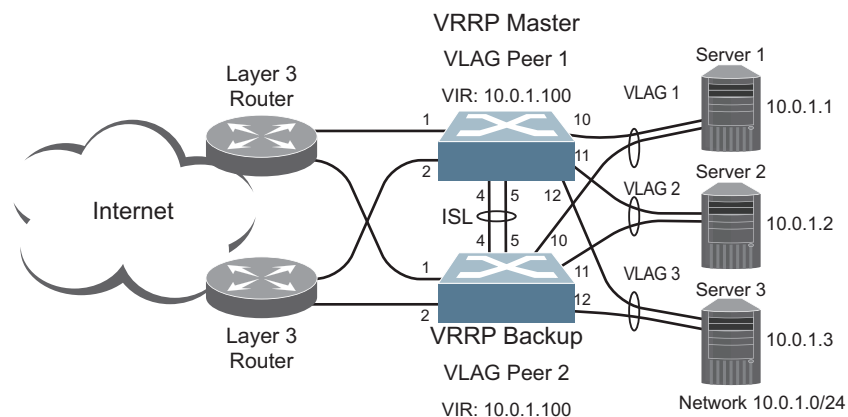
Note: For VLAG Peer 2, the management interface would be configured as 10.10.10.2, and the health check would be configured for 10.10.10.1, pointing back to VLAG Peer 1.

VLAGs with VRRP

Note: In a multi-layer environment, configure VRRP separately for each layer. We recommend that you configure VRRP only on the tier with uplinks. See [“Configuring VLAGs in Multiple Layers” on page 184](#).

VRRP (see [“Virtual Router Redundancy Protocol” on page 499](#)) can be used in conjunction with VLAGs and LACP-capable devices to provide seamless redundancy.

Figure 19. Active-Active Configuration using VRRP and VLAGs



Task 1: Configure VLAG Peer 1

Note: Before enabling VLAG, you must configure the VLAG tier ID and ISL portchannel.

1. Configure VLAG tier ID and enable VLAG globally.

```
RS8296(config)# vlag tier-id 10
RS8296(config)# vlag enable
```

2. Configure appropriate routing.

```
RS8296(config)# router ospf
RS8296(config-router-ospf)# area 1 area-id 0.0.0.1
RS8296(config-router-ospf)# enable
RS8296(config-router-ospf)# exit
```

Although OSPF is used in this example, static routing could also be deployed. For more information, see [“OSPF” on page 443](#) or [“Basic IP Routing” on page 325](#).

3. Configure a server-facing interface.

```
RS8296(config)# interface ip 3
RS8296(config-ip-if)# ip address 10.0.1.10 255.255.255.0
RS8296(config-ip-if)# vlan 100
RS8296(config-ip-if)# exit
```

4. Turn on VRRP and configure the Virtual Interface Router.

```
RS8296(config)# router vrrp
RS8296(config-vrrp)# enable
RS8296(config-vrrp)# virtual-router 1 virtual-router-id 1
RS8296(config-vrrp)# virtual-router 1 interface 3
RS8296(config-vrrp)# virtual-router 1 address 10.0.1.100
RS8296(config-vrrp)# virtual-router 1 enable
```

5. Set the priority of Virtual Router 1 to 101, so that it becomes the Master.

```
RS8296(config-vrrp)# virtual-router 1 priority 101
RS8296(config-vrrp)# exit
```

6. Configure the ISL ports and place them into a port trunk group:

```
RS8296(config)# interface port 4-5
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# lacp mode active
RS8296(config-if)# lacp key 2000
RS8296(config-if)# exit
```

Note: In this case, a dynamic trunk group is shown. A static trunk (portchannel) could be configured instead.

7. Configure the upstream ports.

```
RS8296(config)# interface port 1
RS8296(config-if)# switchport access vlan 10
RS8296(config-if)# exit
RS8296(config)# interface port 2
RS8296(config-if)# switchport access vlan 20
RS8296(config-if)# exit
```

8. Configure the server ports.

```
RS8296(config)# interface port 10
RS8296(config-if)# switchport access vlan 100
RS8296(config-if)# exit
RS8296(config)# interface port 11
RS8296(config-if)# switchport access vlan 100
RS8296(config-if)# exit
RS8296(config)# interface port 12
RS8296(config-if)# switchport access vlan 100
RS8296(config-if)# exit
```

9. Configure all VLANs including VLANs for the VLAGs.

```
RS8296(config)# vlan 10
RS8296(config-vlan)# exit
RS8296(config)# interface port 1
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 10
RS8296(config-if)# exit

RS8296(config)# vlan 20
RS8296(config-vlan)# exit
RS8296(config)# interface port 2
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 20
RS8296(config-if)# exit

RS8296(config)# vlan 100
RS8296(config-vlan)# exit
RS8296(config)# interface port 4-5, 10-12
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 100
RS8296(config-if)# exit
```

10. Configure Internet-facing interfaces.

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip address 172.1.1.10 255.255.255.0
RS8296(config-ip-if)# vlan 10
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# ip ospf area 1
RS8296(config-ip-if)# ip ospf enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip address 172.1.3.10 255.255.255.0
RS8296(config-ip-if)# vlan 20
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# ip ospf area 1
RS8296(config-ip-if)# ip ospf enable
RS8296(config-ip-if)# exit
```

11. Place the VLAG port(s) in their port trunk groups.

```
RS8296(config)# interface port 10
RS8296(config-if)# lacp mode active
RS8296(config-if)# lacp key 1000
RS8296(config-if)# exit
RS8296(config)# interface port 11
RS8296(config-if)# lacp mode active
RS8296(config-if)# lacp key 1100
RS8296(config-if)# exit
RS8296(config)# interface port 12
RS8296(config-if)# lacp mode active
RS8296(config-if)# lacp key 1200
RS8296(config-if)# exit
```

12. Assign the trunks to the VLAGs:

```
RS8296(config)# vlag adminkey 1000 enable
RS8296(config)# vlag adminkey 1100 enable
RS8296(config)# vlag adminkey 1200 enable
```

13. Verify the completed configuration:

```
# show vlag
```

Task 2: Configure VLAG Peer 2

The VLAG peer (VLAG Peer 2) must be configured using the same ISL trunk type (dynamic or static), the same VLAN, and the same STP mode and Tier ID used on VLAG Switch 1.

For each corresponding VLAG on the peer, the port trunk type (dynamic or static), VLAN, and STP mode and ID must be the same as on VLAG Switch 1.

1. Configure VLAG tier ID and enable VLAG globally.

```
RS8296(config)# vlag tier-id 10
RS8296(config)# vlag enable
```

2. Configure appropriate routing.

```
RS8296(config)# router ospf
RS8296(config-router-ospf)# area 1 area-id 0.0.0.1
RS8296(config-router-ospf)# enable
RS8296(config-router-ospf)# exit
```

Although OSPF is used in this example, static routing could also be deployed.

3. Configure a server-facing interface.

```
RS8296(config)# interface ip 3
RS8296(config-ip-if)# ip address 10.0.1.11 255.255.255.0
RS8296(config-ip-if)# vlan 100
RS8296(config-ip-if)# exit
```

4. Turn on VRRP and configure the Virtual Interface Router.

```
RS8296(config)# router vrrp
RS8296(config-vrrp)# enable
RS8296(config-vrrp)# virtual-router 1 virtual-router-id 1
RS8296(config-vrrp)# virtual-router 1 interface 3
RS8296(config-vrrp)# virtual-router 1 address 10.0.1.100
RS8296(config-vrrp)# virtual-router 1 enable
```

5. Configure the ISL ports and place them into a port trunk group:

```
RS8296(config)# interface port 4-5
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# lacp mode active
RS8296(config-if)# lacp key 2000
RS8296(config-if)# exit
```

6. Configure the upstream ports.

```
RS8296(config)# interface port 1
RS8296(config-if)# switchport access vlan 30
RS8296(config-if)# exit
RS8296(config)# interface port 2
RS8296(config-if)# switchport access vlan 40
RS8296(config-if)# exit
```

7. Configure the server ports.

```
RS8296(config)# interface port 10
RS8296(config-if)# switchport access vlan 100
RS8296(config-if)# exit
RS8296(config)# interface port 11
RS8296(config-if)# switchport access vlan 100
RS8296(config-if)# exit
RS8296(config)# interface port 12
RS8296(config-if)# switchport access vlan 100
RS8296(config-if)# exit
```

8. Configure all VLANs including VLANs for the VLAGs.

```
RS8296(config)# vlan 30
RS8296(config-vlan)# exit
RS8296(config)# interface port 1
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 30
RS8296(config-if)# exit

RS8296(config)# vlan 40
RS8296(config-vlan)# exit
RS8296(config)# interface port 2
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 40
RS8296(config-if)# exit

RS8296(config)# vlan 100
RS8296(config-vlan)# exit
RS8296(config)# interface port 4-5,10-12
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 100
RS8296(config-if)# exit
```

9. Configure Internet-facing interfaces.

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip address 172.1.2.11 255.255.255.0
RS8296(config-ip-if)# vlan 30
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# ip ospf area 1
RS8296(config-ip-if)# ip ospf enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip address 172.1.4.12 255.255.255.0
RS8296(config-ip-if)# vlan 40
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# ip ospf area 1
RS8296(config-ip-if)# ip ospf enable
RS8296(config-ip-if)# exit
```

10. Place the VLAG port(s) in their port trunk groups.

```
RS8296(config)# interface port 10
RS8296(config-if)# lACP mode active
RS8296(config-if)# lACP key 1000
RS8296(config-if)# exit
RS8296(config)# interface port 11
RS8296(config-if)# lACP mode active
RS8296(config-if)# lACP key 1100
RS8296(config-if)# exit
RS8296(config)# interface port 12
RS8296(config-if)# lACP mode active
RS8296(config-if)# lACP key 1200
RS8296(config-if)# exit
```

11. Assign the trunks to the VLAGs:

```
RS8296(config)# vlag adminkey 1000 enable
RS8296(config)# vlag adminkey 1100 enable
RS8296(config)# vlag adminkey 1200 enable
```

12. Verify the completed configuration:

```
# show vlag
```

Configuring VLAGs in Multiple Layers

Figure 20. VLAG in Multiple Layers

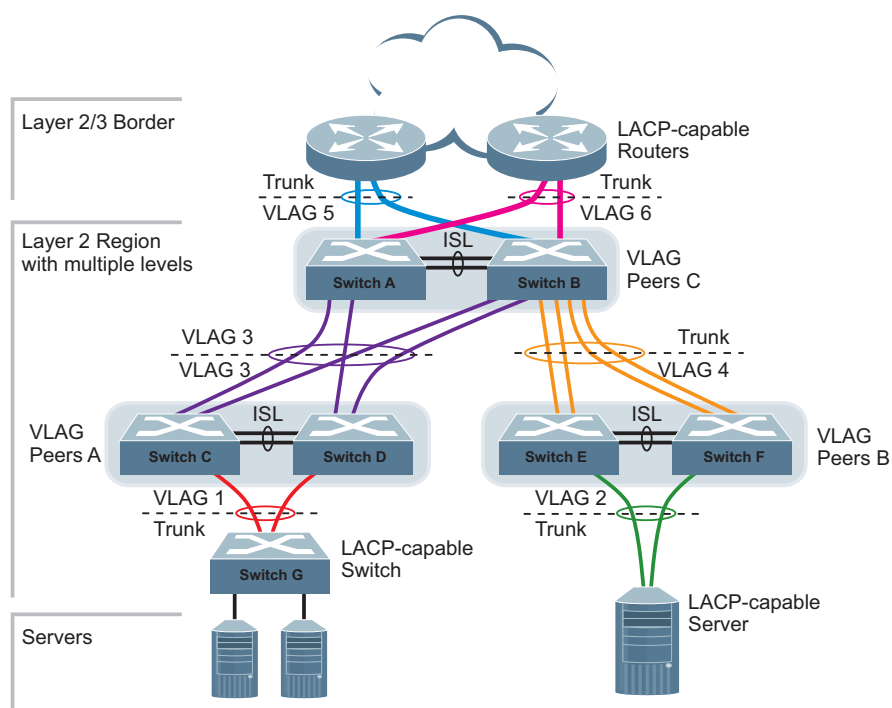


Figure 20 shows an example of VLAG being used in a multi-layer environment. Following are the configuration steps for the topology.

Task 1: Configure Layer 2/3 border switches.

Configure ports on border switch as follows:

```
RS8296(config)# interface port 1,2
RS8296(config-if)# lacp key 100
RS8296(config-if)# lacp mode active
RS8296(config-if)# exit
```

Repeat these commands for the second border switch.

Task 2: Configure switches in the Layer 2 region.

Consider the following:

- ISL ports on switches A and B - ports 1, 2
- Ports connecting to Layer 2/3 - ports 5, 6
- Ports on switches A and B connecting to switches C and D: ports 10, 11
- Ports on switch B connecting to switch E: ports 15, 16
- Ports on switch B connecting to switch F: ports 17, 18

1. Configure VLAG tier ID and enable VLAG globally.

```
RS8296(config)# vlag tier-id 10
RS8296(config)# vlag enable
```

2. Configure ISL ports on Switch A.

```
RS8296(config)# interface port 1,2
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# lacp key 200
RS8296(config-if)# lacp mode active
RS8296(config-if)# exit

RS8296(config)# vlag isl adminkey 200
RS8296(config-vlan)# exit
```

3. Configure port on Switch A connecting to Layer 2/3 router 1.

```
RS8296(config)# vlan 10
VLAN number 10 with name "VLAN 10" created
VLAN 10 was assigned to STG 10
RS8296(config-vlan)# exit
RS8296(config)# interface port 1,2,5
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 10
RS8296(config-if)# exit

RS8296(config)# interface port 5
RS8296(config-if)# lacp key 400
RS8296(config-if)# lacp mode active
RS8296(config-if)# exit

RS8296(config)# vlag adminkey 400 enable
```

Repeat these steps on Switch B for ports connecting to Layer 2/3 router 1.

4. Configure port on Switch A connecting to Layer 2/3 router 2.

```
RS8296(config)# vlan 20
VLAN number 20 with name "VLAN 20" created
VLAN 20 was assigned to STG 20
RS8296(config-vlan)# exit
RS8296(config)# interface port 1,2,6
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 20
RS8296(config-if)# exit

RS8296(config)# interface port 6
RS8296(config-if)# lACP key 500
RS8296(config-if)# lACP mode active
RS8296(config-if)# exit

RS8296(config)# vlag adminkey 500 enable
```

Repeat these commands on Switch B for ports connecting to Layer 2/3 router 2.

5. Configure ports on Switch A connecting to downstream VLAG switches C and D.

```
RS8296(config)# vlan 20
RS8296(config-vlan)# exit
RS8296(config)# interface port 10,11
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 20
RS8296(config-if)# lACP key 600
RS8296(config-if)# lACP mode active
RS8296(config-if)# exit

RS8296(config)# vlag adminkey 600 enable
```

Repeat these commands on Switch B for ports connecting to downstream VLAG switch C and D.

6. Configure ports on Switch B connecting to downstream switches E and F.

```
RS8296(config)# vlan 30
RS8296(config-vlan)# exit
RS8296(config)# interface port 15-18
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 30
RS8296(config-if)# lACP key 700
RS8296(config-if)# lACP mode active
RS8296(config-if)# exit
```

7. Configure ISL between switches C and D, and between E and F as shown in Step 1.
8. Configure the Switch G as shown in Step 2.

VLAG with PIM

Protocol Independent Multicast (PIM) is designed for efficiently routing multicast traffic across one or more IPv4 domains. PIM is used by multicast source stations, client receivers, and intermediary routers and switches, to build and maintain efficient multicast routing trees. PIM is protocol independent; It collects routing information using the existing unicast routing functions underlying the IPv4 network, but does not rely on any particular unicast protocol. For PIM to function, a Layer 3 routing protocol (such as BGP, OSPF, RIP, or static routes) must first be configured on the switch.

Lenovo Network Operating System supports PIM in Sparse Mode (PIM-SM) and Dense Mode (PIM-DM). For more details on PIM, see [Chapter 33, “Protocol Independent Multicast”](#) on page 473.

PIM, when configured in a VLAG topology, provides efficient multicast routing along with redundancy and failover. Only the primary VLAG switch forwards multicast data packets to avoid duplicate packets reaching the access layer switch. The secondary VLAG switch is available as backup and forwards packets only when the primary VLAG switch is not available and during failover.

See [Figure 18 on page 173](#) for a basic VLAG topology. For PIM to function in a VLAG topology, the following are required:

- IGMP (v1 or v2) must be configured on the VLAG switches.
- A Layer 3 routing protocol (such as BGP, OSPF, RIP, or static routes) must be globally enabled and on VLAG-associated IP interfaces for multicast routing.
- The VLAG switches must be connected to upstream multicast routers.
- The Rendezvous Point (RP) and/or the Bootstrap router (BSR) must be configured on the upstream router.
- Flooding must be disabled on the VLAG switches or in the VLAN associated with the VLAG ports.
- ISL ports must be members of VLANs that have VLAG ports as members.

For PIM configuration steps and commands, see [“PIM Configuration Examples”](#) on page 483.

Traffic Forwarding

In a VLAG with PIM topology, traffic forwarded by the upstream router is managed as follows:

- If the primary and secondary VLAG ports are up, the primary switch forwards traffic to the receiver. The secondary switch blocks the traffic. Multicast entries are created on both the VLAG switches: primary VLAG switch with forward state; secondary VLAG switch with pruned state.
- If the primary VLAG port fails, the secondary VLAG switch forwards traffic to the receiver. Multicast entries are created on both the VLAG switches: primary VLAG switch with forward state; secondary VLAG switch with VLAG pruned state.

- If the secondary VLAG port fails, the primary VLAG switch forwards traffic to the receiver. Multicast entries are created on both the VLAG switches: primary VLAG switch with forward state; secondary VLAG switch with pruned state.
- If the primary VLAG switch is down, the secondary VLAG switch forwards traffic to the receiver. When the primary VLAG switch boots up again, it becomes the secondary VLAG switch and blocks traffic to the receiver. The VLAG switch that was secondary initially becomes the primary and continues forwarding traffic to the receiver.
- If the secondary VLAG switch is down, the primary VLAG switch forwards traffic to the receiver. When the secondary VLAG switch is up, it blocks traffic. The primary switch forwards traffic to the receiver.
- If the uplink to the primary VLAG switch is down, the secondary VLAG switch forwards traffic to the receiver and to the primary VLAG switch over the ISL. The primary VLAG switch blocks traffic to the receiver so the receiver does not get double traffic. Both the VLAG switches will have multicast entries in forward state.
- If the uplink to the secondary VLAG switch is down, the primary VLAG switch forwards traffic to the receiver and to the secondary VLAG switch over the ISL. The secondary VLAG switch blocks traffic to the receiver so the receiver does not get double traffic. Both the VLAG switches will have multicast entries in the forward state.

Health Check

In a VLAG with PIM topology, you must configure health check. See [“Configuring Health Check” on page 177](#).

When health check is configured, and the ISL is down, the primary VLAG switch forwards traffic to the receiver. The secondary VLAG switch ports will be errdisable state and will block traffic to the receiver.

Chapter 13. Quality of Service

Quality of Service features allow you to allocate network resources to mission-critical applications at the expense of applications that are less sensitive to such factors as time delays or network congestion. You can configure your network to prioritize specific types of traffic, ensuring that each type receives the appropriate Quality of Service (QoS) level.

The following topics are discussed in this section:

- [“QoS Overview” on page 190](#)
- [“Using ACL Filters” on page 191](#)
- [“Using DSCP Values to Provide QoS” on page 193](#)
- [“Using 802.1p Priority to Provide QoS” on page 199](#)
- [“Queuing and Scheduling” on page 200](#)
- [“Control Plane Protection” on page 200](#)
- [“WRED with ECN” on page 201](#)

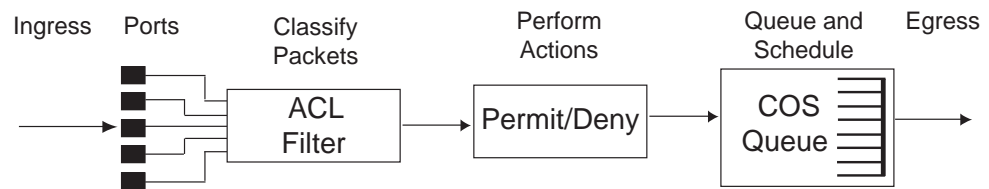
QoS Overview

QoS helps you allocate guaranteed bandwidth to the critical applications, and limit bandwidth for less critical applications. Applications such as video and voice must have a certain amount of bandwidth to work correctly; using QoS, you can provide that bandwidth when necessary. Also, you can put a high priority on applications that are sensitive to timing out or that cannot tolerate delay, by assigning their traffic to a high-priority queue.

By assigning QoS levels to traffic flows on your network, you can ensure that network resources are allocated where they are needed most. QoS features allow you to prioritize network traffic, thereby providing better service for selected applications.

Figure 21 shows the basic QoS model used by the switch.

Figure 21. QoS Model



The basic QoS model works as follows:

- Classify traffic:
 - Read DSCP value.
 - Read 802.1p priority value.
 - Match ACL filter parameters.
- Perform actions:
 - Define bandwidth and burst parameters
 - Select actions to perform on in-profile and out-of-profile traffic
 - Deny packets
 - Permit packets
 - Mark DSCP or 802.1p Priority
 - Set COS queue (with or without re-marking)
- Queue and schedule traffic:
 - Place packets in one of the COS queues.
 - Schedule transmission based on the COS queue.

Using ACL Filters

Access Control Lists (ACLs) are filters that allow you to classify and segment traffic, so you can provide different levels of service to different traffic types. Each filter defines the conditions that must match for inclusion in the filter, and also the actions that are performed when a match is made.

Lenovo Network Operating System 8.2 supports up to 256 ACLs.

The G8296 allows you to classify packets based on various parameters. For example:

- Ethernet: source MAC, destination MAC, VLAN number/mask, Ethernet type, priority.
- IPv4: Source IP address/mask, destination address/mask, type of service, IP protocol number.
- TCP/UDP: Source port, destination port, TCP flag.
- Packet format

For ACL details, see [“Access Control Lists” on page 103](#).

Summary of ACL Actions

Actions determine how the traffic is treated. The G8296 QoS actions include the following:

- Pass or Drop
- Re-mark a new DiffServ Code Point (DSCP)
- Re-mark the 802.1p field
- Set the COS queue

ACL Metering and Re-Marking

You can define a profile for the aggregate traffic flowing through the G8296 by configuring a QoS meter (if desired) and assigning ACLs to ports. When you add ACLs to a port, make sure they are ordered correctly in terms of precedence.

Actions taken by an ACL are called *In-Profile* actions. You can configure additional In-Profile and Out-of-Profile actions on a port. Data traffic can be metered, and re-marked to ensure that the traffic flow provides certain levels of service in terms of bandwidth for different types of network traffic.

Metering

QoS metering provides different levels of service to data streams through user-configurable parameters. A meter is used to measure the traffic stream against a traffic profile, which you create. Thus, creating meters yields In-Profile and Out-of-Profile traffic for each ACL, as follows:

- **In-Profile**—If there is no meter configured or if the packet conforms to the meter, the packet is classified as In-Profile.
- **Out-of-Profile**—If a meter is configured and the packet does not conform to the meter (exceeds the committed rate or maximum burst rate of the meter), the packet is classified as Out-of-Profile.

Using meters, you set a Committed Rate in Kbps (multiples of 64 Mbps). All traffic within this Committed Rate is In-Profile. Additionally, you set a Maximum Burst Size that specifies an allowed data burst larger than the Committed Rate for a brief period. These parameters define the In-Profile traffic.

Meters keep the sorted packets within certain parameters. You can configure a meter on an ACL, and perform actions on metered traffic, such as packet re-marking.

Re-Marking

Re-marking allows for the treatment of packets to be reset based on new network specifications or desired levels of service. You can configure the ACL to re-mark a packet as follows:

- Change the DSCP value of a packet, used to specify the service level traffic receives.
- Change the 802.1p priority of a packet.

Using DSCP Values to Provide QoS

The switch uses the Differentiated Services (DiffServ) architecture to provide QoS functions. DiffServ is described in IETF RFCs 2474 and 2475.

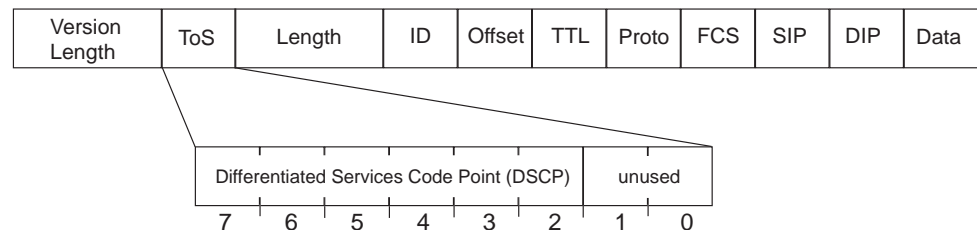
The six most significant bits in the TOS byte of the IP header are defined as DiffServ Code Points (DSCP). Packets are marked with a certain value depending on the type of treatment the packet must receive in the network device. DSCP is a measure of the Quality of Service (QoS) level of the packet.

The switch can classify traffic by reading the DiffServ Code Point (DSCP) or IEEE 802.1p priority value, or by using filters to match specific criteria. When network traffic attributes match those specified in a traffic pattern, the policy instructs the switch to perform specified actions on each packet that passes through it. The packets are assigned to different Class of Service (COS) queues and scheduled for transmission.

Differentiated Services Concepts

To differentiate between traffic flows, packets can be classified by their DSCP value. The Differentiated Services (DS) field in the IP header is an octet, and the first six bits, called the DS Code Point (DSCP), can provide QoS functions. Each packet carries its own QoS state in the DSCP. There are 64 possible DSCP values (0-63).

Figure 22. Layer 3 IPv4 packet



The switch can perform the following actions to the DSCP:

- Read the DSCP value of ingress packets.
- Re-mark the DSCP value to a new value
- Map the DSCP value to a Class of Service queue (COSq).

The switch can use the DSCP value to direct traffic prioritization.

With DiffServ, you can establish policies to direct traffic. A policy is a traffic-controlling mechanism that monitors the characteristics of the traffic, (for example, its source, destination, and protocol) and performs a controlling action on the traffic when certain characteristics are matched.

Trusted/Untrusted Ports

By default, all ports on the G8296 are trusted. To configure untrusted ports, re-mark the DSCP value of the incoming packet to a lower DSCP value using the following commands:

```
RS8296(config)# interface port 1
RS8296(config-if)# dscp-marking
RS8296(config-if)# exit
RS8296(config)# qos dscp dscp-mapping <DSCP value (0-63)> <new value>
RS8296(config)# qos dscp re-marking
```

Per Hop Behavior

The DSCP value determines the Per Hop Behavior (PHB) of each packet. The PHB is the forwarding treatment given to packets at each hop. QoS policies are built by applying a set of rules to packets, based on the DSCP value, as they hop through the network.

The default settings are based on the following standard PHBs, as defined in the IEEE standards:

- Expedited Forwarding (EF)—This PHB has the highest egress priority and lowest drop precedence level. EF traffic is forwarded ahead of all other traffic. EF PHB is described in RFC 2598.
- Assured Forwarding (AF)—This PHB contains four service levels, each with a different drop precedence, as shown in the following table. Routers use drop precedence to determine which packets to discard last when the network becomes congested. AF PHB is described in RFC 2597.

Drop Precedence	Class 1	Class 2	Class 3	Class 4
Low	AF11 (DSCP 10)	AF21 (DSCP 18)	AF31 (DSCP 26)	AF41 (DSCP 34)
Medium	AF12 (DSCP 12)	AF22 (DSCP 20)	AF32 (DSCP 28)	AF42 (DSCP 36)
High	AF13 (DSCP 14)	AF23 (DSCP 22)	AF33 (DSCP 30)	AF43 (DSCP 38)

- Class Selector (CS)—This PHB has eight priority classes, with CS7 representing the highest priority, and CS0 representing the lowest priority, as shown in the following table. CS PHB is described in RFC 2474.

Priority	Class Selector	DSCP
Highest	CS7	56
	CS6	48
	CS5	40
	CS4	32
	CS3	24
	CS2	16
	CS1	8
Lowest	CS0	0

QoS Levels

Table 19 shows the default service levels provided by the switch, listed from highest to lowest importance:

Table 19. *Default QoS Service Levels*

Service Level	Default PHB	802.1p Priority
Critical	CS7	7
Network Control	CS6	6
Premium	EF, CS5	5
Platinum	AF41, AF42, AF43, CS4	4
Gold	AF31, AF32, AF33, CS3	3
Silver	AF21, AF22, AF23, CS2	2
Bronze	AF11, AF12, AF13, CS1	1
Standard	DF, CS0	0

DSCP Re-Marking and Mapping

The switch can use the DSCP value of ingress packets to re-mark the DSCP to a new value, and to set an 802.1p priority value. Use the following command to view the default settings.

```
RS8296# show qos dscp
Current DSCP Remarking Configuration: OFF
```

DSCP	New DSCP	New 802.1p Prio
0	0	0
1	1	0
2	2	0
3	3	0
4	4	0
5	5	0
6	6	0
7	7	0
8	8	1
9	9	0
10	10	1
...		
54	54	0
55	55	0
56	56	7
57	57	0
58	58	0
59	59	0
60	60	0
61	61	0
62	62	0
63	63	0

Use the following command to turn on DSCP re-marking globally:

```
RS8296(config)# qos dscp re-marking
```

Then you must enable DSCP re-marking on any port that you wish to perform this function (Interface Port mode).

Note: If an ACL meter is configured for DSCP re-marking, the meter function takes precedence over QoS re-marking.

DSCP Re-Marking Configuration Examples

Example 1

The following example includes the basic steps for re-marking DSCP value and mapping DSCP value to 802.1p.

1. Turn DSCP re-marking on globally, and define the DSCP-DSCP-802.1p mapping. You can use the default mapping.

```
RS8296(config)# qos dscp re-marking
RS8296(config)# qos dscp dscp-mapping <DSCP value (0-63)> <new value>
RS8296(config)# qos dscp dot1p-mapping <DSCP value (0-63)> <802.1p value>
```

2. Enable DSCP re-marking on a port.

```
RS8296(config)# interface port 1
RS8296(config-if)# qos dscp re-marking
RS8296(config-if)# exit
```

Example 2

The following example assigns strict priority to VoIP traffic and a lower priority to all other traffic.

1. Create an ACL to re-mark DSCP value and COS queue for all VoIP packets.

```
RS8296(config)# access-control list 2 tcp-udp source-port 5060 0xffff
RS8296(config)# access-control list 2 meter committed-rate 10000000
RS8296(config)# access-control list 2 meter enable
RS8296(config)# access-control list 2 re-mark in-profile dscp 56
RS8296(config)# access-control list 2 re-mark dot1p 7
RS8296(config)# access-control list 2 action permit
```

2. Create an ACL to set a low priority to all other traffic.

```
RS8296(config)# access-control list 3 action set-priority 1
RS8296(config)# access-control list 3 action permit
```

3. Apply the ACLs to a port and enable DSCP marking.

```
RS8296(config)# interface port 5
RS8296(config-if)# access-control list 2
RS8296(config-if)# access-control list 3 ethernet source-mac-address
00:00:00:00:00:00 00:00:00:00:00:00
RS8296(config-if)# dscp-marking
RS8296(config-if)# exit
```

4. Enable DSCP re-marking globally.

```
RS8296(config)# qos dscp re-marking
```

5. Assign the DSCP re-mark value.

```
RS8296(config)# qos dscp dscp-mapping 40 9
RS8296(config)# qos dscp dscp-mapping 46 9
```

6. Assign strict priority to VoIP COS queue.

```
RS8296(config)# qos transmit-queue weight-cos 7 0
```

7. Map priority value to COS queue for non-VoIP traffic.

```
RS8296(config)# qos transmit-queue mapping 1 1
```

8. Assign weight to the non-VoIP COS queue.

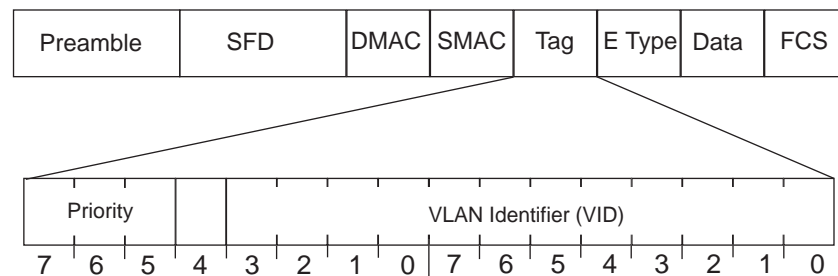
```
RS8296(config)# qos transmit-queue weight-cos 1 2
```

Using 802.1p Priority to Provide QoS

The G8296 provides Quality of Service functions based on the priority bits in a packet's VLAN header. (The priority bits are defined by the 802.1p standard within the IEEE 802.1Q VLAN header.) The 802.1p bits, if present in the packet, specify the priority to be given to packets during forwarding. Packets with a numerically higher (non-zero) priority are given forwarding preference over packets with lower priority value.

The IEEE 802.1p standard uses eight levels of priority (0-7). Priority 7 is assigned to highest priority network traffic, such as OSPF or RIP routing table updates, priorities 5-6 are assigned to delay-sensitive applications such as voice and video, and lower priorities are assigned to standard applications. A value of 0 (zero) indicates a "best effort" traffic prioritization, and this is the default when traffic priority has not been configured on your network. The switch can filter packets based on the 802.1p values.

Figure 23. Layer 2 802.1q/802.1p VLAN tagged packet



Ingress packets receive a priority value, as follows:

- **Tagged packets**—switch reads the 802.1p priority in the VLAN tag.
- **Untagged packets**—switch tags the packet and assigns an 802.1p priority value, based on the port's default 802.1p priority.

Egress packets are placed in a COS queue based on the priority value, and scheduled for transmission based on the.

To configure a port's default 802.1p priority value, use the following commands.

```
RS8296(config)# interface port 1
RS8296(config-if)# dot1p <802.1p value (0-7)>
RS8296(config-if)# exit
```

Queuing and Scheduling

The G8296 can be configured to have 8 output Class of Service (COS) queues per port, into which each packet is placed. Each packet's 802.1p priority determines its COS queue, except when an ACL action sets the COS queue of the packet.

You can configure the following attributes for COS queues:

- Map 802.1p priority value to a COS queue
- Define the scheduling weight of each COS queue

You can map 802.1p priority value to a COS queue, as follows:

```
RS8296(config)# qos transmit-queue mapping <802.1p priority value (0-7)> <COS queue (0-7)>
```

To set the COS queue scheduling weight, use the following command.

```
RS8296(config)# qos transmit-queue weight-cos <COSq number> <COSq weight (0-15)>
```

Control Plane Protection

Control plane receives packets that are required for the internal protocol state machines. This type of traffic is usually received at low rate. However, in some situations such as DOS attacks, the switch may receive this traffic at a high rate. If the control plane protocols are unable to process the high rate of traffic, the switch may become unstable.

The control plane receives packets that are channeled through protocol-specific packet queues. Multiple protocols can be channeled through a common packet queue. However, one protocol cannot be channeled through multiple packet queues. These packet queues are applicable only to the packets received by the software and does not impact the regular switching or routing traffic. Packet queue with a higher number has higher priority.

You can configure the bandwidth for each packet queue. Protocols that share a packet queue will also share the bandwidth.

The following commands configure the control plane protection (CoPP) feature:

```
RS8296(config)# qos protocol-packet-control packet-queue-map <0-43>
                  <protocol>                               (Configure a queue for a protocol)
RS8296(config)# qos protocol-packet-control rate-limit-packet-queue
                  <0-43> <1-10000>                       (Set the bandwidth for the queue,
                                                           in packets per second)
```

WRED with ECN

Weighted Random Early Detection (WRED) is a congestion avoidance algorithm that helps prevent a TCP collapse, where a congested port indiscriminately drops packets from all sessions. The transmitting hosts wait to retransmit resulting in a dramatic drop in throughput. Often times, this TCP collapse repeats in a cycle, which results in a saw-tooth pattern of throughput. WRED selectively drops packets before the queue gets full, allowing majority of the traffic to flow smoothly.

WRED discards packets based on the CoS queues. Packets marked with lower priorities are discarded first.

Explicit Congestion Notification (ECN) is an extension to WRED. For packets that are ECN-aware, the ECN bit is marked to signal impending congestion instead of dropping packets. The transmitting hosts then slow down sending packets.

How WRED/ECN work together

For implementing WRED, you must define a profile with minimum threshold, maximum threshold, and a maximum drop probability. The profiles can be defined on a port or a CoS.

For implementing ECN, you require ECN-specific field that has two bits—the ECN-capable Transport (ECT) bit and the CE (Congestion Experienced) bit—in the IP header. ECN is identified and defined by the values in these bits in the Differentiated Services field of IP Header. [Table 20](#) shows the combination values of the ECN bits.

Table 20. *ECN Bit Setting*

ECT Bit	CE Bit	Description
0	0	Not ECN-capable
0	1	Endpoints of the transport protocol are ECN-capable
1	0	Endpoints of the transport protocol are ECN-capable
1	1	Congestion experienced

WRED and ECN work together as follows:

- If the number of packets in the queue is less than the minimum threshold, packets are transmitted. This happens irrespective of the ECN bit setting, and on networks where only WRED (without ECN) is enabled.
- If the number of packets in the queue is between the minimum threshold and the maximum threshold, one of the following occurs:
 - If the ECN field on the packet indicates that the endpoints are ECN-capable and the WRED algorithm determines that the packet has likely been dropped based on the drop probability, the ECT and CE bits for the packet are changed to 1, and the packet is transmitted.
 - If the ECN field on the packet indicates that neither endpoint is ECN-capable, the packet may be dropped based on the WRED drop probability. This is true even in cases where only WRED (without ECN) is enabled.
 - If the ECN field on the packet indicates that the network is experiencing congestion, the packet is transmitted. No further marking is required.
- If the number of packets in the queue is greater than the maximum threshold, packets are dropped based on the drop probability. This is the identical treatment a packet receives when only WRED (without ECN) is enabled.

Configuring WRED/ECN

For configuring WRED, you must define a TCP profile and a non-TCP profile. WRED prioritizes TCP traffic over non-TCP traffic.

For configuring ECN, you must define a TCP profile. You don't need a non-TCP profile as ECN can be enabled only for TCP traffic.

If you do not configure the profiles, the profile thresholds are set to maximum value of 0xFFFF to avoid drops.

Note: WRED/ECN can be configured only on physical ports and not on trunks. WRED and ECN are applicable only to unicast traffic.

Consider the following guidelines for configuring WRED/ECN:

- Profiles can be configured globally or per port. Global profiles are applicable to all ports.
- Always enable the global profile before applying the port-level profile.

Note: You can enable the global profile and disable the port-level profile. However, you must not enable the port-level profile if the global profile is disabled.

- WRED settings are dependent on Memory Management Unit (MMU) Settings. If you change the MMU setting, it could impact WRED functionality.
- You cannot enable WRED if you have QoS buffer settings such as Converged Enhanced Ethernet (CEE), Priority-based Flow Control (PFC), or Enhanced Transmission Selection (ETS).
- The number of WRED profiles per-port must match the total number of COS Queues configured in the system.
- If you have configured a TCP profile and enabled ECN, ECN remarking happens only if all traffic experiencing congestion is TCP traffic.
- Configure a TCP profile only after enabling ECN on the interface.

- You can apply TCP and non-TCP profile configurations irrespective of ECN status (enabled/disabled).

WRED/ECN Configuration Example

Follow these steps to enable WRED/ECN and configure a global and/or port-level profile. If you configure global and port-level profile, WRED/ECN uses the port-level profile to make transmit/drop decisions when experiencing traffic congestion.

Configure Global Profile for WRED

1. Enable WRED globally.

```
RS8296(config)# qos random-detect enable
```

2. Enable a transmit queue.

```
RS8296(config)# qos random-detect transmit-queue 0 enable
```

3. Configure WRED thresholds (minimum, maximum, and drop rate) for TCP traffic.

```
RS8296(config)# qos random-detect transmit-queue 0 tcp min-threshold 1  
max-threshold 2 drop-rate 3
```

Note: Percentages are of Average Queue available in hardware and not percentages of traffic.

4. Configure WRED thresholds (minimum, maximum, and drop rate) for non-TCP traffic.

```
RS8296(config)# qos random-detect transmit-queue 0 non-tcp min-threshold  
4 max-threshold 5 drop-rate 6
```

5. Select the port.

```
RS8296(config)# interface port 1
```

6. Enable WRED for the port.

```
RS8296(config-if)# random-detect enable  
RS8296(config-if)# exit
```

Configure Port-level Profile for WRED

1. Enable WRED globally.

```
RS8296(config)# qos random-detect enable
```

2. Select the port.

```
RS8296(config)# interface port 1
```

3. Enable WRED for the port .

```
RS8296(config-if)# random-detect enable
```

4. Enable a transmit queue.

```
RS8296(config-if)# random-detect transmit-queue 0 enable
```

5. Configure WRED thresholds (minimum, maximum, and drop rate) for TCP traffic.

```
RS8296(config-if)# random-detect transmit-queue 0 tcp min-threshold 11  
max-threshold 22 drop-rate 33
```

Note: Percentages are of Average Queue available in hardware and not percentages of traffic.

6. Configure WRED thresholds (minimum, maximum, and drop rate) for non-TCP traffic.

```
RS8296(config-if)# random-detect transmit-queue 0 non-tcp min-threshold  
44 max-threshold 55 drop-rate 66  
RS8296(config-if)# exit
```

Configure Global Profile for ECN

1. Enable ECN globally.

```
RS8296(config)# qos random-detect ecn enable
```

2. Enable a transmit queue.

```
RS8296(config)# qos random-detect transmit-queue 0 enable
```

3. Configure ECN thresholds (minimum, maximum, and drop rate) for TCP traffic.

```
RS8296(config)# qos random-detect transmit-queue 0 tcp min-threshold 1  
max-threshold 2 drop-rate 3
```

Note: Percentages are of Average Queue available in hardware and not percentages of traffic.

4. Select the port.

```
RS8296(config)# interface port 1
```

5. Enable ECN for the port.

```
RS8296(config-if)# random-detect ecn enable  
RS8296(config-if)# exit
```

Configure Port-level Profile for ECN

1. Enable ECN globally.

```
RS8296(config)# qos random-detect ecn enable
```

2. Select the port.

```
RS8296(config)# interface port 1
```

3. Enable ECN for the port.

```
RS8296(config-if)# random-detect ecn enable
```

4. Enable a transmit queue.

```
RS8296(config-if)# random-detect transmit-queue 0 enable
```

5. Configure ECN thresholds (minimum, maximum, and drop rate) for TCP traffic.

```
RS8296(config-if)# random-detect transmit-queue 0 tcp min-threshold 11  
max-threshold 22 drop-rate 33  
RS8296(config-if)# exit
```

Note: Percentages are of Average Queue available in hardware and not percentages of traffic.

Verifying WRED/ECN

Use the following command to view global WRED/ECN information.

```
RS8296(config)# show qos random-detect  
Current wred and ecn configuration:  
Global ECN: Enable  
Global WRED: Enable  
TQ0: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-  
Ena 10 20 30 10 20 30  
TQ1: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-  
Dis 0 0 0 0 0 0  
TQ2: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-  
Dis 0 0 0 0 0 0  
TQ3: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-  
Dis 0 0 0 0 0 0  
TQ4: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-  
Dis 0 0 0 0 0 0  
TQ5: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-  
Dis 0 0 0 0 0 0  
TQ6: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-  
Dis 0 0 0 0 0 0  
TQ7: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-  
Dis 0 0 0 0 0 0
```

Use the following command to view port-level WRED/ECN information.

```
RS8296(config)# show interface port 1 random-detect
Port: 1
    ECN: Enable
    WRED: Enable
TQ0: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-
Dis      0      0      0      0      0      0
TQ1: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-
Ena      4      5      6      1      2      3
TQ2: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-
Dis      0      0      0      0      0      0
TQ3: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-
Dis      0      0      0      0      0      0
TQ4: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-
Dis      0      0      0      0      0      0
TQ5: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-
Dis      0      0      0      0      0      0
TQ6: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-
Dis      0      0      0      0      0      0
TQ7: -WRED-TcpMinThr-TcpMaxThr-TcpDrate-NonTcpMinThr-NonTcpMaxThr-NonTcpDrate-
Dis      0      0      0      0      0      0
```

Chapter 14. Precision Time Protocol

As defined in the IEEE 1588-2008 standard, Precision Time Protocol (PTP) is a precision clock synchronization protocol for networked measurement and control systems. PTP provides system-wide synchronization accuracy and precision in the sub-microsecond range with minimal network and local clock computing resources. The synchronization is achieved through the exchange of messages: General messages that carry data but need not be time stamped; Event messages that are time stamped and are critical for clock synchronization accuracy.

A PTP network consists of PTP-enabled devices such as switches or routers. These devices consist of the following types of clocks:

- **Master clock:** In a PTP domain, the clock with the most precise time is considered the master clock. A best master clock algorithm determines the highest quality clock in a network.
- **Ordinary clock:** An ordinary clock synchronizes its time with the Master clock. The ordinary clock has a bidirectional communication with the master clock. By receiving synchronization/delay response and sending delay request packets, the ordinary clock adjusts its time with the master clock.
- **Boundary clock:** A boundary clock connects to multiple networks. It synchronizes with the attached master clock and in turn acts as a master clock to all attached ordinary clocks. Boundary clocks help to reduce the effect of jitter in Ethernet-based networks.
- **Transparent clock:** A transparent clock listens for PTP packets and adjusts the correction field in the PTP event packets as they pass the PTP device.

RackSwitch G8296 supports the configuration of ordinary clock and transparent clock. It cannot be a master clock as the switch does not participate in the master clock selection process.

Note: Lenovo Network Operating System does not support IPv6 for PTP.

By default, PTP version 2 is installed on the switch but is globally disabled. Use the following command to globally enable PTP:

```
RS8296(config)# ptp {ordinary|transparent} enable
```

PTP is configured on switch ports. In case of trunk ports, the PTP configuration must be the same on all ports participating in the same trunk. The switch uses only one port from a trunk (typically the one used by a multicast protocol) to forward PTP packets.

By default, PTP is enabled on all the switch ports. To disable PTP on a port, use the following commands:

```
RS8296(config)# interface port <port number>  
RS8296(config-if)# no ptp
```

Note: PTP cannot be enabled on management ports.

PTP packets have a Control Plane Protection (CoPP) queue of 36. You can change this CoPP priority using the following command:

```
RS8296(config)# qos protocol-packet-control packet-queue-map <0-47>
<protocol>
```

You can modify the PTP queue rate using the following command:

```
RS8296(config)# qos protocol-packet-control rate-limit-packet-queue <0-47>
<1-10000>
```

Ordinary Clock Mode

When the RackSwitch G8296 is configured as an ordinary clock, it synchronizes its clock with the master clock. If the G8296 does not detect a master clock, it will not synchronize its clock with any other device. In this mode, the G8296's clock cannot be modified manually or using another time protocol such as Network Time Protocol (NTP).

As an ordinary clock, the G8296 synchronizes with a single PTP domain. The switch uses a delay-request mechanism to synchronize with the master clock. The switch uses a source IP address for the packets it generates. You can create a loopback interface for this purpose. By default, the switch uses the lowest interface in the VLAN from which the sync messages are received. To assign a loopback interface, use the following command:

```
RS8296(config)# ip ptp source-interface loopback <interface number>
```

Note: If there are no interfaces on the switch that belong to the VLAN from which the sync messages are received, then the ordinary clock will not function. An error message will be generated. You can view this message using the `RS8296# show ptp` command.

Transparent Clock Mode

When the G8296 is configured as a transparent clock, its time can be set manually or using any time protocol. You must configure PTPv2 for the transparent clock to function. The switch does not modify PTPv1 packets as they pass through the switch.

As a transparent clock, the G8296 supports syntonization (synchronization of clock frequency but not time) and synchronization with multiple domains.

Event packets received on all ports on the switch that have PTP enabled will be adjusted with the residence time. The switch sends all PTP packets to the multicast group address: 224.0.1.129. You can use Protocol Independent Multicast (PIM), Internet Group Management Protocol (IGMP), or any other multicast protocol to route the PTP packets.

Tracing PTP Packets

PTP packets can be traced on the PTP ports. These packets can be identified by their destination IP address and UDP ports. The following table includes the IEEE standard specification:

Table 21. *IEEE Standard PTP Messages*

Message	IP Address/UDP Port
PTP-primary: All PTP messages except peer delay mechanism messages	224.0.1.129
PTP-pdelay: Peer delay mechanism messages	224.0.0.107
Event Messages: Sync, delay request, peer delay request, peer delay response	319
General Messages: Announce, follow-up, delay response, peer delay response follow-up, management	320

Viewing PTP Information

The following table includes commands for viewing PTP information:

Table 22. *PTP Information Commands*

Command	Description
RS8296(config)# show ptp	Displays global PTP information
RS8296(config)# show interface port <port number>	Displays port information including port-specific PTP information
RS8296(config)# show ptp counters	Displays ingress and egress PTP counters

Part 4: Advanced Switching Features

Chapter 15. OpenFlow

This document describes how you can create an OpenFlow Switch instance on the RackSwitch G8296.

The following topics are discussed in this document:

- [“OpenFlow Overview” on page 214](#)
- [“Configuring OpenFlow” on page 228](#)

OpenFlow Overview

OpenFlow architecture consists of a control plane residing outside of the switch (typically on a server) and a data plane residing in the switch. The control plane is called OpenFlow controller. The data plane which resides in the switch consists of a set of flows which determine the forwarding of data packets.

The OpenFlow protocol is described in the OpenFlow Switch Specification 1.0.0 and OpenFlow Switch Specification 1.3.1.

An OpenFlow network consists of simple flow-based switches in the data path, with a remote controller to manage all switches in the OpenFlow network.

OpenFlow maintains a TCP channel for communication of flow management between the controller and the switch. All controller-switch communication takes place over the switch's management network.

Switch Profiles

The RackSwitch G8296 can be used for configuring OpenFlow and legacy switching features simultaneously. However, Layer 2 and Layer 3 switching features can be configured only on the ports that are not OpenFlow ports. Legacy switching ports and OpenFlow ports do not communicate with each other.

Alternately, the switch can be configured as an OpenFlow-only switch if you do not need to configure legacy switching features.

Based on your requirement, select the switch boot profile using the following commands:

- OpenFlow-only:

```
RS8296(config)# boot profile openflow
```

The switch will operate only in OpenFlow environment. None of the legacy switching features will be supported.

- OpenFlow and Legacy Switching:

```
RS8296(config)# boot profile default
```

Legacy switching features can be configured on the non-OpenFlow ports. By default, the switch boots in this profile.

Reload the switch to apply boot profile changes.

OpenFlow Versions

The RackSwitch G8296 can be configured to run either version 1.0 or version 1.3 of the OpenFlow standard. By default, the switch uses OpenFlow 1.0. To shift between versions, use the following commands:

- RS8296(config)# boot openflow v1

The switch will run OpenFlow 1.0 after reboot. This version does not support MPLS-based flows (See [“Flow Tables” on page 215](#)), groups, and static portchannels.

- RS8296(config)# boot openflow v13

The switch will run OpenFlow 1.3 after reboot. This version does not support emergency flow tables (See [“Emergency Mode” on page 223](#)).

OpenFlow 1.3 includes (but not limited to) the following key features:

- Layer 3 MPLS
- Static LAG
- MAC address/IP address masking
- Flexible Table Miss and Fail Secure
- 40Gb support
- Static CLI for Flow Programming
- OpenFlow 1.0 backward compatibility
- OpenFlow Groups

OpenFlow Instance

The G8296 supports up to two instances of the OpenFlow protocol. Each instance appears as a switch to the controller. Instances on the same switch can be connected to different virtual networks. Each instance maintains a separate TCP channel for communication of flow management between controller and switch. Each instance supports up to four controllers. However, only one controller per instance is active at any point in time.

Global OpenFlow configuration applies to all instances. OpenFlow parameters for an instance can be configured using the command:

```
RS8296(config)# openflow instance <instance ID>.
```

OpenFlow ports cannot be shared between instances.

Flow Tables

A set of a flow identification condition and an action towards a flow is called *flow entry*, and the database that stores the entries is called the flow table. A flow is defined as all the packets matching a flow entry in an OpenFlow flow table. Each flow entry includes:

- Qualifiers - These are header fields that are matched with a packet.
- Actions to be performed when a packet matches the qualifiers.

The controller decides which flows to admit and the path their packets will follow.

The switch classifies the flows as ACL-based, FDB-based, or MPLS-based. MPLS-based flows are supported only in version 1.3 of the OpenFlow standard. When the switch operates in *OpenFlow* boot profile (See [“Switch Profiles” on page 214](#)), a maximum of 2000 ACL-based flows, 4096 FDB multicast flows, 123904 FDB unicast flows, 1000 MPLS label pop flows, and 2000 MPLS label push flows are available. When the switch operates in *default* boot profile, a maximum of 1500 ACL-based flows, 4096 FDB multicast flows, 123904 FDB unicast flows, 1000 MPLS

label pop flows, and 2000 MPLS label push flows are available. The instances share these flows dynamically. To guarantee a specific number of flows to an instance, use the following commands:

OpenFlow boot profile:

```
RS8296(config)# openflow instance <instance ID>
RS8296(config-openflow-instance)# max-flow-acl <0-2000>
RS8296(config-openflow-instance)# max-flow-mcast-fdb <0-4096>
RS8296(config-openflow-instance)# max-flow-ucast-fdb <0-123904>
RS8296(config-openflow-instance)# max-flow-mpls-pop <0-1000>
RS8296(config-openflow-instance)# max-flow-mpls-push <0-2000>
```

Default boot profile:

```
RS8296(config)# openflow instance <instance ID>
RS8296(config-openflow-instance)# max-flow-acl <0-1500>
```

Note: When the switch operates in *default* boot profile, the number of FDB and MPLS flows in an instance cannot be guaranteed.

In version 1.0 of the OpenFlow standard, the G8296 supports two flow tables per switch instance; basic flow table and emergency flow table. Actions are applied to packets that match the flow entry. This is done in the data path.

This system identifies packets as a flow by matching parameters in the following fields:

- Ingress port
- Source MAC (SMAC)
- Destination MAC (DMAC)
- Ether Type
- VLAN TAG – Single VLAN tag – VLAN ID and Priority
- IP address (source IP and destination IP)
- IP Protocol
- MPLS label
- DSCP bits
- Layer 4 Port (TCP, UDP)
- ICMP code and type
- If EtherType is ARP, then the specified ARP type (request/reply) or SIP in the ARP payload can be used a to match a packet.

Once a packet arrives, the switch searches the flow table. When a flow entry is hit in the search, the packet is processed according to the action specified in the flow entry.

Static Flows

You can configure static flow entries for OpenFlow instances. The switch forwards traffic based on these entries even if it is not connected to a controller. Up to 1000 static ACL entries across all instances can be configured. An OpenFlow controller cannot modify or delete these entries. Static flow entries can replace entries installed by a controller. Static flow entries are not lost when the switch is reloaded.

Static flow entries are based on the following qualifiers, actions, and options:

Table 23. *Static Flow Entry Qualifiers in OpenFlow 1.0*

Qualifier	Description
ingress-port	port of the instance
src-mac	source MAC address
dst-mac	destination MAC address
vlan-id	VLAN identifier (tagged, untagged, or 0-4094)
vlan-priority	802.1p (0-7)
src-ip	source IP address
dst-ip	destination IP address
src-port	Layer 4 source port (0-65535)
dst-port	Layer 4 destination port (0-65535)
ether-type	"arp"/"0806" or "ip"/"0800" or (hex-value <= 65535)
protocol	"tcp" or "udp" or 0-255
tos	IP TOS (0-255)
type	"request" or "reply" (can be set only if ether type is ARP)
all	all qualifiers or any qualifier

Table 24. *Static Flow Entry Qualifiers in OpenFlow 1.3*

Qualifier	Description
in-port	switch input port
in-portchannel	switch input portchannel (only static trunk supported)
in-phy-port	switch physical input port, valid only when in-port is pecified
eth-src	source MAC address and mask
eth-dst	destination MAC address and mask
vlan-vid	VLAN identifier (0-4095 + 65535 (untagged))

Table 24. *Static Flow Entry Qualifiers in OpenFlow 1.3 (continued)*

Qualifier	Description
vlan-pcp	802.1p(0-7)
ipv4-src	source IPv4 address and mask
ipv4-dst	destination IPv4 address and mask
tcp-src	TCP source port (0-65535)
tcp-dst	TCP destination port (0-65535)
udp-src	UDP source port (0-65535)
udp-dst	UDP destination port (0-65535)
icmpv4-type	ICMPv4 type
icmpv4-code	ICMPv4 code
eth-type	"arp"/"0806" or "ip"/"0800" or (hex-value < =65535)
ip-proto	"tcp" or "udp" or 0-255
ip-dscp	IP DSCP (6 bits in ToS field)
arp-op	"request" or "reply" (can be set only if eth-type is ARP)
mpls-label	MPLS label
all	all qualifiers or any qualifier

Table 25. *Static Flow Entry Actions in OpenFlow 1.0*

Action	Description
out-put	"all", "in-port", "flood", "controller" or a valid port
set-src-mac	change source MAC address
set-dst-mac	change destination MAC address
strip-vlan-id	strip VLAN
set-vlan-id	set VLAN ID
set-vlan-priority	set 802.1p priority (0-7)
set-nw-tos	set IP TOS (0-255)
drop	drop the packet

Table 26. *Static Flow Entry Actions in OpenFlow 1.3*

Action	Description
output	physical ports reserved ports: all, controller, in_port
output-portchannel	Logical ports (only support static LAG)
set-eth-src	change source MAC address
set-eth-dst	change destination MAC address
push-vlan	push a new VLAN tag, 0x8100
pop-vlan	pop the VLAN tag
push-mpls	push a new MPLS tag, 0x8847
pop-mpls	pop MPLS tag, 0x0800 or 0x800 (IP Ethertype)
set-vlan-vid	set VLAN ID, <0-4094>
set-vlan-pcp	set 802.1p priority (0-7)
set-ip-dscp	Set IP DSCP (0-63), 6 highest bits in TOS
set-mpls-ttl	set MPLS TTL
set-mpls-label	Set MPLS label, <1-0xFFFFF>
copy-ttl-out	copy TTL outward - from next-to-outermost to outermost
copy-ttl-in	copy TTL inward - from outermost to next-to-outermost
drop	drop the packet

Note: Actions for static flows must be specified in the following order: set-mpls-ttl, copy-ttl-in, pop-mpls, pop-vlan, push-mpls, push-vlan, copy-ttl-out, set-field, output. The actions output, output-portchannel, set-eth-src, set-eth-dst, set-vlan-vid, set-vlan-pcp, set-ip-dscp, and set-mpls-label may be used in any order.

Table 27. *Static Flow Entry Options*

Option	Description
max-len	maximum length of flow to send to controller

Port Membership

When static flow entries are configured, port membership changes are handled as follows:

- If a port is the “in-port” or “out-port” in a static flow entry, the port membership cannot be changed.
- When a port membership changes, the ingress bitmap of static entries with in-port ANY will be updated.

- When a port membership changes, the egress bitmap of static entries with redirect output FLOOD/ANY will be updated.

Static Flow Examples

Following are example static flow entries for Openflow 1.0:

- Basic ACL flow:

```
RS8296(config-openflow-instance)# static-table add index 1 match
ingress-port=1 actions out-put=10 priority 12345
```

- Flow with multiple qualifiers and actions:

```
RS8296(config-openflow-instance)# static-table add index 2 match
vlan-id=1,dst-mac=00:00:00:00:00:01 actions
set-vlan-priority=3,out-put=20 priority 1000
```

- Flow with action: output to controller:

```
RS8296(config-openflow-instance)# static-table add index 3 match all
actions out-put=controller options max-len=65534 priority 1000
```

Static ACL flow entries can be deleted using the command:

```
RS8296(config-openflow-instance)# static-table remove index <index number>
```

Following are example static flow entries for OpenFlow 1.3:

- Basic ACL flow:

```
RS8296(config-openflow-instance)# static-table add index 1 match
in-port=41 actions output=50 priority 12345
```

- Flow with multiple qualifiers and actions:

```
RS8296(config-openflow-instance)# static-table add index 2 match
vlan-id=1,eth-dst=00:00:00:00:00:01 actions set-vlan-pcp=3,output=50
priority 1000
```

Static ACL flow entries for OpenFlow 1.3 can also be deleted using the command:

```
RS8296(config-openflow-instance)# static-table remove index <index number>
```

In OpenFlow version 1.0:

```
RS8296(config-openflow-instance)# show openflow table

Openflow instance 1 is currently disabled

Openflow Instance Id: 2

BASIC FLOW TABLE

CONTROLLER ADDED FLOWS

Flow:1
  Filter Based, priority:    0, hard-time-out:    0, idle-time-out:    0
  cookie: 0x0
  QUALIFIERS:
  ACTION: drop
  STATS:  packets=0, bytes=0

Flow:2
  Filter Based, priority:65535, hard-time-out:    0, idle-time-out:    0
  cookie: 0x0
  QUALIFIERS: ether-type:0x88cc
               protocol:  0, tos:  0
               src-ip:0.0.0.0/32, dst-ip:0.0.0.0/32
               src-port:   0, dst-port:   0
  ACTION: output:CONTROLLER [Max Len: 65535 / - bytes (C/S)]
  STATS:  packets=0, bytes=0

Flow:3
  Filter Based, priority:    2, hard-time-out:    0, idle-time-out:    0
  cookie: 0x0
  QUALIFIERS:
  ACTION: output:CONTROLLER [Max Len: 65535 / - bytes (C/S)]
  STATS:  packets=0, bytes=0
```

In OpenFlow version 1.3:

```
RS8296(config-openflow-instance)# show openflow table

Openflow instance 1 is currently disabled

Openflow Instance Id: 2

CONTROLLER added FLOWS

Flow 1, Filter Based, priority:1000, hard-time-out:0, idle-time-out:0
cookie:0x0
QUALIFIERS: in-port:PCH1
eth-dst:01:00:08:00:00:03
vlan-vid:1
Instruction: apply_action
ACTION: output:18, PCH1, PCH66
STATS: packets=0, bytes=0

Flow 2, Filter Based, priority:2, hard-time-out:0, idle-time-out:0
cookie:0x0
QUALIFIERS:
Instruction: write_action
ACTION: output:CONTROLLER [Max Len: 65535 / - bytes (C/S)]
STATS: packets=0, bytes=0

Flow 3, Filter Based, priority:65535, hard-time-out:0, idle-time-out:0
cookie:0x0
QUALIFIERS: eth-type:0x88cc
Instruction: write_action
ACTION: output:CONTROLLER [Max Len: 65535 / - bytes (C/S)]
STATS: packets=0, bytes=0

Flow 4, Filter Based, priority:0, hard-time-out:0, idle-time-out:0
cookie:0x0
QUALIFIERS:
Instruction: empty
ACTION: drop
STATS: packets=0, bytes=0
```

Table-Miss

In OpenFlow 1.3, if a match is not found for an arriving packet, the packet is either dropped or sent to the controller. If sent to the controller, the controller decides which action(s) to perform on all packets from the same flow. The decision is then sent to the switch and cached as an entry in the switch instance's flow table. If the controller decides to add the flow, it sends a flow add message to the switch. The switch then adds the flow in its flow table. The next arriving packet that belongs to the same flow is then forwarded at line-rate through the switch without consulting the controller.

By default, packets that don't match any of the existing flow entries are dropped. To set up an OpenFlow instance to send non-matching packets to the controller instead of dropping them, use the following commands:

```
RS8296(config)# openflow instance <instance ID>
RS8296(config-openflow-instance)# table-miss controller
```

Fail Secure Mode

In OpenFlow 1.3, if contact with all controllers is lost, the switch automatically enters a fail secure mode. In this mode, the switch continues to operate based on the existing flow entries until they expire (according to the configured flow timeout value), while packets and messages destined to the controllers are dropped. When a connection to a controller is reestablished, the controller has the option to either delete or keep any remaining flow entries.

Emergency Mode

Note: Emergency Mode is supported only in version 1.0 of the OpenFlow standard.

By default, Emergency mode is disabled. In this state, if the connection to the controller fails, the switch enters Fail Secure mode.

To enable Emergency mode, use the following command:

```
RS8296(config)# openflow instance <instance ID>  
RS8296(config-openflow-instance)# emergency
```

In Emergency mode enabled state, if the connection to the controller fails, the switch tries to establish connection with any of the other configured controllers. If it is unable to connect with any controller, it enters Emergency mode. It replaces the flow entries with the entries from the emergency flow table.

The switch stays in the Emergency mode for the time configured as the Emergency timeout interval (default value is 30 seconds), after which the switch tries to establish connection with any configured controller.

- If connection with a controller is established, the switch exits Emergency mode. Entries in the Emergency flow table are retained. If desired, the controller may delete all the emergency flow entries.
- If connection with a controller is not established, the switch stays in Emergency mode and continues to forward packets based on the Emergency flows. It retries to establish a connection with a controller every time the Emergency timeout interval expires.

Emergency mode can be activated or deactivated per instance. To activate Emergency mode on an instance, use the following command:

```
RS8296# openflow instance <instance ID> enter-emergency
```

To deactivate Emergency mode on an instance, use the following command:

```
RS8296# no openflow instance <instance ID> enter-emergency
```

Table 28 displays an example of emergency flows created:

Table 28. *Emergency Flows*

```
RS8296(config)# show openflow table

Openflow Instance Id: 1

BASIC FLOW TABLE

Flow:1
  FDB Based, priority: 1000, hard-time-out:    0
  QUALIFIERS: dst-mac:01-02-03-05-06-00, vlan-id: 100
  ACTION: out-port:21

Flow:2
  Filter Based, priority:32768, hard-time-out:    0, idle-time-out:    0
  QUALIFIERS: vlan-id: 100
             dst-mac:01-02-03-66-76-00
  ACTION: output:22
  STATS:  packets=0, bytes=0

EMERGENCY FLOW TABLE

Flow:1
  FDB Based, priority: 1000, hard-time-out:    0
  QUALIFIERS: dst-mac:01-02-03-66-06-00, vlan-id: 100
  ACTION: out-port:21

Flow:2
  Filter Based, priority:32768, hard-time-out:    0, idle-time-out:    0
  QUALIFIERS: vlan-id: 100
             dst-mac:01-02-03-66-06-00
  ACTION: output:22

Openflow Instance Id: 2

BASIC FLOW TABLE

Flow:1
  FDB Based, priority: 1000, hard-time-out:    0
  QUALIFIERS: dst-mac:01-55-03-11-96-00, vlan-id: 200
  ACTION: out-port:31

EMERGENCY FLOW TABLE

Flow:1
  FDB Based, priority: 1000, hard-time-out:    0
  QUALIFIERS: dst-mac:01-55-03-11-16-00, vlan-id: 200
  ACTION: out-port:31
```

OpenFlow Ports

When OpenFlow is enabled, all OpenFlow instance member ports become OpenFlow ports. OpenFlow ports have the following characteristics:

- Learning is turned off.
- Flood blocking is turned on.
- Spanning Tree Protocol (STP) is disabled.

The switch communicates with OpenFlow controllers through controller management ports or through dedicated out-of-band management ports on the switch. All OpenFlow ports, except controller management ports, must be members of VLAN 1. Controller management ports can be members of any VLAN except VLAN 1.

Note: When the switch operates in the *default* boot profile, we recommend that you use a non-OpenFlow port to connect the switch with the controller. Use the following command to view port information:

```
RS8296(config)# show interface information
```

For each OpenFlow instance, when you configure the controller IP address and port, the switch establishes a TCP connection with the controller for flow control and management. See [Step 3 on page 228](#). The switch supports up to four controllers per instance. The default controller port is 6633 and is reachable via out-of-band management port (port MGT) or in-band port. The controller management ports must not be members of an OpenFlow instance. You can use a controller to manage and control multiple instances.

Use the following command to configure a data port as a controller management port:

```
RS8296(config)# openflow mgmtport <port number>
```

Note: In *default* boot profile, when you disable OpenFlow, the OpenFlow ports become legacy switch ports and are added to the default VLAN 1.

OpenFlow Edge Ports

You can configure a port as an OpenFlow edge port. Edge ports are connected to either non-OpenFlow switches or servers. OpenFlow edge ports have the following characteristics:

- Flood blocking is turned on.
- MAC learning is turned on.

Use the following commands to configure a port as an edge port:

```
RS8296(config)# openflow instance <instance ID>  
RS8296(config-openflow-instance)# edgeport <port number>
```

Note: Edge ports are not OpenFlow standard ports. You must configure edge ports only if the controller supports it.

Link Aggregation

In OpenFlow 1.3, logical ports can be used to aggregate links. You can add a logical port to an OpenFlow instance using the following command:

```
RS8296(config)# openflow instance <instance ID>  
RS8296(config-openflow-instance)# member portchannel <Trunk ID>
```

Once added, the logical port inherits the OpenFlow data properties such as MAC learning turned off, flood blocking turned on, and STP disabled. The switch derives the logical port ID based on the configured trunk ID.

The trunk link stays active as long as at least one member port is up. The trunk link speed is an aggregation of the speed of individual member ports. If a port goes down, the trunk link speed lowers.

A portchannel can be added to the OpenFlow instance using the command:

```
RS8296(config-openflow-instance)# member portchannel1 <Trunk ID>
```

To add a portchannel to the edge ports list, use the following command:

```
RS8296(config-openflow-instance)# edgeport portchannel1 <Trunk ID>
```

Data Path ID

The data path ID—automatically computed—is a combination of two bytes of the instance ID and six bytes of the switch MAC address. Alternately, the data path ID can be manually configured using the following command. Each instance on the switch must have a unique data path ID:

```
RS8296(config)# openflow instance <instance ID>
RS8296(config-openflow-instance)# dpid <Data path ID>
                                     (Hex string starting with 0x)
```

Note: If the data path ID is changed, the switch instance closes the active connection and reconnects with the modified data path ID.

sFlow Compatibility

OpenFlow ports that are set for sFlow sampling will send incoming sample packets to both the sFlow server or analyzer and the OpenFlow controller as “packet-in”. All sFlow servers must be configured as reachable over a management port or a non-OpenFlow data port.

Unicast FDB flows support both ingress and egress sampling. ACL-based flows, MPLS-based flows, multicast FDB-based flows, packet-in, send-to-controller, and mirror-to-controller flows only support ingress sampling.

OpenFlow Groups

In OpenFlow 1.3, the RackSwitch G8296 supports OpenFlow groups based on the OpenFlow 1.3.1 specification. The following OpenFlow group features are supported:

- Group types: ALL and INDIRECT
- Up to 256 groups per instance
- Group operations: ADD, MODIFY and DELETE
- Support for group multipart messages: group counter statistics, group description and group features

- Flows: ACL, FDB and MPLS
- Port types: physical port, logical port (static portchannel)

Configuring OpenFlow

The RackSwitch G8296 is capable of operating both in normal switching environment (*default* boot profile) and in OpenFlow switch environment (*OpenFlow* boot profile).

Note: If you disable OpenFlow, you must reboot the switch in order to resume normal switch environment operations.

Perform the following steps to configure an OpenFlow switch instance.

1. Enable OpenFlow:

```
RS8296(config)# openflow enable
```

2. Create an OpenFlow switch instance and add data ports:

```
RS8296(config)# openflow instance <1-2>  
RS8296(config-openflow-instance)# member <port number or range>
```

In version 1.3 of the OpenFlow standard, you can also add static portchannels to the instance:

```
RS8296(config-openflow-instance)# member portchannel <1-72>
```

3. Configure a controller for the OpenFlow switch instance:

```
RS8296(config-openflow-instance)# controller <1-4> address <IP address>  
[mgt-port | data-port]  
RS8296(config-openflow-instance)# controller <1-4> port <1-65535>
```

4. Enable the OpenFlow switch instance:

```
RS8296(config-openflow-instance)# enable
```

The switch is ready to perform switching functions in an OpenFlow environment.

5. Verify OpenFlow configuration:

```
RS8296(config)# show openflow <instance ID> information
```

Configuration Example 1 - OpenFlow Boot Profile

The following example includes steps to configure an OpenFlow switch instance when the switch operates in *OpenFlow* boot profile.

Configure OpenFlow instance 1, which connects with one controller via in-band management port and another controller via out-of-band management port; and OpenFlow instance 2, which connects with two controllers via in-band management ports.

1. Configure IP interface 128 for out-of-band connection:

```
RS8296(config)# interface ip 128
RS8296(config-ip-if)# ip address 172.20.100.1 255.255.0.0 enable
RS8296(config-ip-if)# exit
```

2. Configure IP interface 1 for in-band connection:

```
RS8296(config)# vlan 3000
RS8296(config-vlan)# exit
RS8296(config)# interface port 5
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 3000
RS8296(config-if)# exit

RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip address 172.21.100.1 255.255.0.0 enable
RS8296(config-ip-if)# vlan 3000
RS8296(config-ip-if)# exit
```

3. Configure IP interface 2 for in-band connection:

```
RS8296(config)# vlan 4000
RS8296(config-vlan)# exit
RS8296(config)# interface port 6
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 4000
RS8296(config-if)# exit

RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip address 172.22.100.1 255.255.0.0 enable
RS8296(config-ip-if)# vlan 4000
RS8296(config-ip-if)# exit
```

4. Enable OpenFlow globally:

```
RS8296(config)# openflow enable
```

5. Configure OpenFlow in-band management ports:

```
RS8296(config)# openflow mgmtport 5,6 (Switch can connect with the controllers via
dataport 5 and 6, which are connected to the
controller networks)
```

Note: Step 5 is not required when the switch operates in *default* boot profile.

6. Create OpenFlow switch instance 1 and add data ports:

```
RS8296(config)# openflow instance 1 (Create OpenFlow instance 1)
RS8296(config-openflow-instance)# member 17,18,19-25 (Add data ports 17,18, and data
port range 19 through 25 as
members of OpenFlow instance 1)
```

7. Configure controller 1 IP address using out-of-band management port:

```
RS8296(config-openflow-instance)# controller 1 address 172.20.100.73  
mgt-port
```

(Switch connects with controller 1 via the out-of-band management port; default controller port is used in this example)

8. Configure controller 2 IP address using in-band management port:

```
RS8296(config-openflow-instance)# controller 2 address 172.21.100.73  
data-port
```

(Switch connects with controller 2 via the in-band management port configured in [Step 5 on page 229](#); default controller port is used in this example)

9. Enable OpenFlow instance 1:

```
RS8296(config-openflow-instance)# enable  
RS8296(config-openflow-instance)# exit
```

10. Create OpenFlow switch instance 2 and add data ports:

```
RS8296(config)# openflow instance 2 (Create OpenFlow instance 2)  
RS8296(config-openflow-instance)# member 26,27,28-34
```

(Add data ports 26,27, and data port range 28 through 34 as members of OpenFlow instance 2)

11. Configure controller 1 IP address using in-band management port:

```
RS8296(config-openflow-instance)# controller 1 address 172.21.100.73  
data-port
```

(Switch connects with controller 1 via the in-band management port configured in [Step 5](#); default controller port is used in this example)

12. Configure controller 2 IP address using in-band management port:

```
RS8296(config-openflow-instance)# controller 2 address 172.22.100.73  
data-port
```

(Switch connects with controller 2 via in-band management port configured in [Step 5 on page 229](#); default controller port is used in this example)

13. Enable OpenFlow instance 2:

```
RS8296(config-openflow-instance)# enable
```

14. View OpenFlow Configuration:

```
RS8296(config)# show running-config
```

Configuration Example 2 - *Default* Boot Profile

The following example includes steps to configure an OpenFlow switch instance when the switch operates in *Default* boot profile.

1. Configure IP interface 128 for out-of-band connection:

```
RS8296(config)# interface ip 128
RS8296(config-ip-if)# ip address 172.20.100.1 255.255.0.0 enable
RS8296(config-ip-if)# exit
```

2. Configure IP interface 1 for in-band connection:

```
RS8296(config)# vlan 3000
RS8296(config-vlan)# exit
RS8296(config)# interface port 6
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 3000
RS8296(config-if)# exit

RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip address 172.21.100.1 255.255.0.0 enable
RS8296(config-ip-if)# vlan 3000
RS8296(config-ip-if)# exit
```

3. Configure IP interface 2 for in-band connection:

```
RS8296(config)# vlan 4000
RS8296(config-vlan)# exit
RS8296(config)# interface port 6
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 4000
RS8296(config-if)# exit

RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip address 172.22.100.1 255.255.0.0 enable
RS8296(config-ip-if)# vlan 4000
RS8296(config-ip-if)# exit
```

4. Enable OpenFlow globally:

```
RS8296(config)# openflow enable
```

5. Create OpenFlow switch instance 1 and add data ports:

```
RS8296(config)# openflow instance 1 (Create OpenFlow instance 1)
RS8296(config-openflow-instance)# member 17,18,19-25 (Add data ports 17,18, and data port range 19 through 25 as members of OpenFlow instance 1)
```

6. Configure controller 1 IP addresses using out-of-band management port:

```
RS8296(config-openflow-instance)# controller 1 address 172.20.100.73
mgt-port

(Switch connects with controller 1 via the out-of-band management port; default controller port is used in this example)
```

7. Configure controller 2 IP address using in-band management port:

```
RS8296(config-openflow-instance)# controller 2 address 172.21.100.73  
data-port  
(Switch connects with controller 2 via the in-band management port; default controller port is used in this example)
```

8. Enable OpenFlow instance 1:

```
RS8296(config-openflow-instance)# enable  
RS8296(config-openflow-instance)# exit
```

9. Create OpenFlow switch instance 2 and add data ports:

```
RS8296(config)# openflow instance 2 (Create OpenFlow instance 2)  
RS8296(config-openflow-instance)# member 26,27,28-34  
Add data ports 26,27, and data port range 28 through 34 as members of OpenFlow instance 2)
```

10. Configure controller 1 IP address using in-band management port:

```
RS8296(config-openflow-instance)# controller 1 address 172.21.100.73  
data-port  
(Switch connects with controller 1 via the in-band management port; default controller port is used in this example)
```

11. Configure controller 2 IP address using in-band management port:

```
RS8296(config-openflow-instance)# controller 2 address 172.22.100.73  
data-port  
Switch connects with controller 2 via in-band management port; default controller port is used in this example)
```

12. Enable OpenFlow instance 2:

```
RS8296(config-openflow-instance)# enable  
RS8296(config-openflow-instance)# exit
```

13. View OpenFlow Configuration:

```
RS8296(config)# show running-config
```


Feature Limitations

When the switch is booted in the *OpenFlow* profile, it operates only in OpenFlow switch environment. None of the normal switching environment features are supported.

If the switch is booted in *default* profile, normal switching environment features can be configured on the non-OpenFlow ports. However, the following features are not supported:

- ACLs
- ECN
- FCoE
- IPMC
- IPv6
- MACL
- vNIC
- VMready

Chapter 16. Deployment Profiles

The Networking OS software for the RackSwitch G8296 can be configured to operate in different modes for different deployment scenarios. Each deployment profile sets different capacity levels for basic switch resources, such as the number of IP Multicast (IPMC) entries and ACL entries, to optimize the switch for different types of networks.

This chapter covers the following topics

- [“Available Profiles” on page 236](#)
- [“Selecting Profiles” on page 237](#)
- [“Automatic Configuration Changes” on page 238](#)

Available Profiles

The following deployment profiles are currently available on the G8296:

- **Default Profile:** This profile is recommended for general network usage. Switch resources are allocated to support a wide range of features such as IPv6, ACLs, and FCoE/CEE.
- **ACL Profile:** This profile enables you to configure the maximum number of ACLs. The IPv6, FCoE/CEE, and VMready features will not be supported. This profile also does not support the forwarding of IPMC packets with IP options.

The properties of each mode are compared in the following table.

Table 29. *Deployment Mode Comparison*

Switch Feature	Capacity, by Mode		
	Default	Openflow	ACL
ACLs	256	256	896
IPv6	Supported	Not Supported	Not Supported
VMready	Supported	Not Supported	Not Supported
VMap	Supported	Not Supported	Not Supported
FCoE/CEE	Supported	Not Supported	Not Supported

Note: Throughout this guide, where feature capacities are listed, values reflect those of the Default profile only, unless otherwise noted.

Selecting Profiles

To change the deployment profile, you must first select the new profile and then reboot the switch.

Note: Before changing profiles, it is recommended that you save the active switch configuration to a backup file so that it may be restored later if desired.

When you select a profile, you will see a warning message. For example, if you select the ACL profile, you will see the following message:

```
Warning: Setting boot profile to "ACL" will cause FIPS, IPv6 and VM ACL
configuration to be lost in next boot and error messages will be
displayed when above configurations are restored.

Next boot will use "ACL" profile.
```

To view the current deployment profile, use the following command:

Use the following commands to change the deployment profile:

When using a specialized profile, menus and commands are unavailable for features that are not supported under the profile. Such menus and commands will be available again only when a supporting profile is used.

Note: Only use deployment profiles other than those listed in this section when under the direction of your support personnel.

Automatic Configuration Changes

When a new profile is loaded, configuration settings for any unsupported features will be ignored. However, these settings are retained in memory until you change or save the current configuration under the new profile. Until then, you can return to the old profile with all prior configuration settings intact.

Once you change or save the configuration under a new profile, any configuration settings related to unsupported features will be reset to their default values. At that point, you will have to reconfigure these settings or use a backup configuration if you reapply the old profile.

For example, when using the ACL profile, because IPv6 is not supported in that mode, IPv6 settings will be excluded when the configuration is saved. Then, if returning to the Default profile, it will be necessary to reconfigure the IPv6 settings, or to use the backup configuration.

Chapter 17. Virtualization

Virtualization allows resources to be allocated in a fluid manner based on the logical needs of the data center, rather than on the strict, physical nature of components. The following virtualization features are included in Lenovo Network Operating System 8.2 on the RackSwitch G8296 (G8296):

- Virtual Local Area Networks (VLANs)

VLANs are commonly used to split groups of networks into manageable broadcast domains, create logical segmentation of workgroups, and to enforce security policies among logical network segments.

For details on this feature, see [“VLANs” on page 119](#).

- Port trunking

A port trunk pools multiple physical switch ports into a single, high-bandwidth logical link to other devices. In addition to aggregating capacity, trunks provides link redundancy.

For details on this feature, see [“Ports and Trunking” on page 131](#).

- Virtual Link Aggregation (VLAs)

With VLAs, two switches can act as a single logical device for the purpose of establishing port trunking. Active trunk links from one device can lead to both VLAG peer switches, providing enhanced redundancy, including active-active VRRP configuration.

For details on this feature, see [“Virtual Link Aggregation Groups” on page 167](#)

- VMready

The switch’s VMready software makes it *virtualization aware*. Servers that run hypervisor software with multiple instances of one or more operating systems can present each as an independent *virtual machine* (VM). With VMready, the switch automatically discovers virtual machines (VMs) connected to switch.

For details on this feature, see [“VMready” on page 241](#).

- Edge Virtual Bridging (QBG)

The 802.1Qbg/Edge Virtual Bridging (EVB) is an emerging IEEE standard for allowing networks to become virtual machine (VM)?aware. EVB bridges the gap between physical and virtual network resources.

For details on this feature, see [“Edge Virtual Bridging” on page 297](#).

- Unified Fabric Port (UFP)

An architecture that logically subdivides a high-speed physical link connecting to a server NIC. UFP provides a switch fabric component to control the NIC.

For details on this feature, see [“Unified Fabric Port” on page 307](#).

N/OS virtualization features provide a highly-flexible framework for allocating and managing switch resources.

Chapter 18. VMready

Virtualization is used to allocate server resources based on logical needs, rather than on strict physical structure. With appropriate hardware and software support, servers can be virtualized to host multiple instances of operating systems, known as virtual machines (VMs). Each VM has its own presence on the network and runs its own service applications.

Software known as a *hypervisor* manages the various virtual entities (VEs) that reside on the host server: VMs, virtual switches, and so on. Depending on the virtualization solution, a virtualization management server may be used to configure and manage multiple hypervisors across the network. With some solutions, VMs can even migrate between host hypervisors, moving to different physical hosts while maintaining their virtual identity and services.

The Lenovo Network Operating System 8.2 VMready feature supports up to 4096 VEs in a virtualized data center environment. The switch automatically discovers the VEs attached to switch ports, and distinguishes between regular VMs, Service Console Interfaces, and Kernel/Management Interfaces in a VMware[®] environment.

VEs may be placed into VM groups on the switch to define communication boundaries: VEs in the same VM group may communicate with each other, while VEs in different groups may not. VM groups also allow for configuring group-level settings such as virtualization policies and ACLs.

The administrator can also pre-provision VEs by adding their MAC addresses (or their IPv4 address or VM name in a VMware environment) to a VM group. When a VE with a pre-provisioned MAC address becomes connected to the switch, the switch will automatically apply the appropriate group membership configuration.

The G8296 with VMready also detects the migration of VEs across different hypervisors. As VEs move, the G8296 NMotion[™] feature automatically moves the appropriate network configuration as well. NMotion gives the switch the ability to maintain assigned group membership and associated policies, even when a VE moves to a different port on the switch.

VMready also works with VMware Virtual Center (vCenter) management software. Connecting with a vCenter allows the G8296 to collect information about more distant VEs, synchronize switch and VE configuration, and extend migration properties.

VE Capacity

When VMready is enabled, the switch will automatically discover VEs that reside in hypervisors directly connected on the switch ports. Networking OS 8.2 supports up to 4096 VEs. Once this limit is reached, the switch will reject additional VEs.

Note: In rare situations, the switch may reject new VEs prior to reaching the supported limit. This can occur when the internal hash corresponding to the new VE is already in use. If this occurs, change the MAC address of the VE and retry the operation. The MAC address can usually be changed from the virtualization management server console (such as the VMware Virtual Center).

Defining Server Ports

Before you configure VMready features, you must first define whether ports are connected to servers or are used as uplink ports. Use the following ISCLI configuration command to define a port as a server port:

```
RS8296(config)# system server-ports port <port alias or number>
```

Ports that are not defined as server ports are automatically considered uplink ports.

VM Group Types

VEs, as well as switch server ports, switch uplink ports, static trunks, and LACP trunks, can be placed into VM groups on the switch to define virtual communication boundaries. Elements in a given VM group are permitted to communicate with each other, while those in different groups are not. The elements within a VM group automatically share certain group-level settings.

N/OS 8.2 supports up to 4093 VM groups. There are two different types:

- Local VM groups are maintained locally on the switch. Their configuration is not synchronized with hypervisors.
- Distributed VM groups are automatically synchronized with a virtualization management server (see [“Assigning a vCenter” on page 252](#)).

Each VM group type is covered in detail in the following sections.

Local VM Groups

The configuration for local VM groups is maintained on the switch (locally) and is not directly synchronized with hypervisors. Local VM groups may include only local elements: local switch ports and trunks, and only those VEs connected to one of the switch ports or pre-provisioned on the switch.

Local VM groups support limited VE migration: as VMs and other VEs move to different hypervisors connected to different ports on the switch, the configuration of their group identity and features moves with them. However, VE migration to and from more distant hypervisors (those not connected to the G8296, may require manual configuration when using local VM groups.

Configuring a Local VM Group

Use the following ISCLI configuration commands to assign group properties and membership:

```
RS8296(config)# virt vmgroup <VM group number> ?
cpu                               (Enable sending unregistered IPMC to CPU)
flood                             (Enable flooding unregistered IPMC)
key <LACP trunk key>              (Add LACP trunk to group)
optflood                          (Enable optimized flooding)
port <port alias or number>       (Add port member to group)
portchannel <trunk group number>  (Add static trunk to group)
profile <profile name>            (Not used for local groups)
stg <Spanning Tree group>         (Add STG to group)
tag                               (Set VLAN tagging on ports)
validate <advanced|basic>         (Validate mode for the group)
vlan <VLAN number>                (Specify the group VLAN)
vm <MAC>|<index>|<UUID>|<IPv4 address>|<name>(Add VM member to group)
vmap <VMAP number> [intports|extports](Specify VMAP number)
vport                             (Add a virtual port to the group)
```

The following rules apply to the local VM group configuration commands:

- `cpu`: Enable sending unregistered IPMC to CPU.
- `flood`: Enable flooding unregistered IPMC.
- `key`: Add LACP trunks to the group.
- `optflood`: Enable optimized flooding to allow sending unregistered IPMC to the Mrouter ports without having any packet loss during the learning period; This option is disabled by default; When `optflood` is enabled, the `flood` and `cpu` settings are ignored.
- `port`: Add switch server ports or switch uplink ports to the group.
- `portchannel`: Add static port trunks to the group.
- `profile`: The profile options are not applicable to local VM groups. Only distributed VM groups may use VM profiles (see [“VM Profiles” on page 245](#)).
- `stg`: The group may be assigned to a Spanning-Tree group for broadcast loop control (see [“Spanning Tree Protocols” on page 145](#)).
- `tag`: Enable VLAN tagging for the VM group. If the VM group contains ports which also exist in other VM groups, enable tagging in both VM groups.
- `validate`: Set validation mode for the group.
- `vlan`: Each VM group must have a unique VLAN number. This is required for local VM groups. If one is not explicitly configured, the switch will automatically assign the next unconfigured VLAN when a VE or port is added to the VM group.
- `vmap`: Each VM group may optionally be assigned a VLAN-based ACL (see [“VLAN Maps” on page 256](#)).
- `vm`: Add VMs.

VMs and other VEs are primarily specified by MAC address. They can also be specified by UUID, IP address, or by the index number as shown in various VMready information output (see [“VMready Information Displays” on page 259](#)).

- `vpport`: Add a virtual port to the group.

If VMware Tools software is installed in the guest operating system (see VMware documentation for information on installing recommended tools), VEs may also be specified by IPv4 address or VE name. However, if there is more than one possible VE for the input, the switch will display a list of candidates and prompt for a specific MAC address.

Only VEs currently connected to the switch port (local) or pending connection (pre-provisioned) are permitted in local VM groups.

Use the `no` variant of the commands to remove or disable VM group configuration settings:

```
RS8296(config)# no virt vmggroup <VM group number> [?]
```

Distributed VM Groups

Distributed VM groups allow configuration profiles to be synchronized between the G8296 and associated hypervisors and VEs. This allows VE configuration to be centralized, and provides for more reliable VE migration across hypervisors.

Using distributed VM groups requires a virtualization management server. The management server acts as a central point of access to configure and maintain multiple hypervisors and their VEs (VMs, virtual switches, and so on).

The G8296 must connect to a virtualization management server before distributed VM groups can be used. The switch uses this connection to collect configuration information about associated VEs, and can also automatically push configuration profiles to the virtualization management server, which in turn configures the hypervisors and VEs. See [“Virtualization Management Servers” on page 252](#) for more information.

VM Profiles

VM profiles are required for configuring distributed VM groups. They are not used with local VM groups. A VM profile defines the VLAN and virtual switch bandwidth shaping characteristics for the distributed VM group. The switch distributes these settings to the virtualization management server, which in turn distributes them to the appropriate hypervisors for VE members associated with the group.

Creating VM profiles is a two part process. First, the VM profile is created as shown in the following command on the switch:

```
RS8296(config)# virt vmprofile <profile name>
```

Next, the profile must be edited and configured using the following configuration commands:

```
RS8296(config)# virt vmprofile edit <profile name> ?
eshaping <average bandwidth> <burst size> <peak>
shaping <average bandwidth> <burst size> <peak>
vlan <VLAN number>
```

For virtual switch bandwidth shaping parameters, average and peak bandwidth are specified in kilobits per second (a value of 1000 represents 1 Mbps). Burst size is specified in kilobytes (a value of 1000 represents 1 MB). Eshaping (egress shaping) is used for distributed virtual switch.

Note: The bandwidth shaping parameters in the VM profile are used by the hypervisor virtual switch software. To set bandwidth policies for individual VEs, see [“VM Policy Bandwidth Control” on page 257](#).

Once configured, the VM profile may be assigned to a distributed VM group as shown in the following section.

Initializing a Distributed VM Group

Note: A VM profile is required before a distributed VM group may be configured. See [“VM Profiles” on page 245](#) for details.

Once a VM profile is available, a distributed VM group may be initialized using the following configuration command:

```
RS8296(config)# virt vmgrouop <VM group number> profile <VM profile name>
```

Only one VM profile can be assigned to a given distributed VM group. To change the VM profile, the old one must first be removed using the following ISCLI configuration command:

```
RS8296(config)# no virt vmgrouop <VM group number> profile
```

Note: The VM profile can be added only to an empty VM group (one that has no VLAN, VMs, or port members). Any VM group number currently configured for a local VM group (see [“Local VM Groups” on page 243](#)) cannot be converted and must be deleted before it can be used for a distributed VM group.

Assigning Members

VMs, ports, and trunks may be added to the distributed VM group only after the VM profile is assigned. Group members are added, pre-provisioned, or removed from distributed VM groups in the same manner as with local VM groups ([“Local VM Groups” on page 243](#)), with the following exceptions:

- VMs: VMs and other VEs are not required to be local. Any VE known by the virtualization management server can be part of a distributed VM group.
- The VM group `vlan` option (see [page 244](#)) cannot be used with distributed VM groups. For distributed VM groups, the VLAN is assigned in the VM profile.

Synchronizing the Configuration

When the configuration for a distributed VM group is modified, the switch updates the assigned virtualization management server. The management server then distributes changes to the appropriate hypervisors.

For VM membership changes, hypervisors modify their internal virtual switch port groups, adding or removing server port memberships to enforce the boundaries defined by the distributed VM groups. Virtual switch port groups created in this fashion can be identified in the virtual management server by the name of the VM profile, formatted as follows:

```
Lenovo_<VM profile name>
```

(or)

```
Lenovo_<VM profile name> <index number> (for vDS)
```

Adding a server host interface to a distributed VM group does not create a new port group on the virtual switch or move the host. Instead, because the host interface already has its own virtual switch port group on the hypervisor, the VM profile settings are applied to its existing port group.

Note: When applying the distributed VM group configuration, the virtualization management server and associated hypervisors must take appropriate actions. If a hypervisor is unable to make requested changes, an error message will be displayed on the switch. Be sure to evaluate all error message and take the appropriate actions for the expected changes to apply.

Removing Member VEs

Removing a VE from a distributed VM group on the switch will have the following effects on the hypervisor:

- The VE will be moved to the `Lenovo_Default` port group in VLAN 0 (zero).
- Traffic shaping will be disabled for the VE.
- All other properties will be reset to default values inherited from the virtual switch.

VMcheck

The G8296 primarily identifies virtual machines by their MAC addresses. An untrusted server or a VM could identify itself by a trusted MAC address leading to MAC spoofing attacks. Sometimes, MAC addresses get transferred to another VM, or they get duplicated.

The VMcheck solution addresses these security concerns by validating the MAC addresses assigned to VMs. The switch periodically sends hello messages on server ports. These messages include the switch identifier and port number. The hypervisor listens to these messages on physical NICs and stores the information, which can be retrieved using the VMware Infrastructure Application Programming Interface (VI API). This information is used to validate VM MAC addresses. Two modes of validation are available: Basic and Advanced.

Use the following command to select the validation mode or to disable validation:

```
RS8296(config)# [no] virt vmgroup <VM group number> validate  
{basic|advanced}
```

Basic Validation

This mode provides port-based validation by identifying the port used by a hypervisor. It is suitable for environments in which MAC reassignment or duplication cannot occur.

The switch, using the hello message information, identifies a hypervisor port. If the hypervisor port is found in the hello message information, it is deemed to be a trusted port. Basic validation should be enabled when:

- A VM is added to a VM group, and the MAC address of the VM interface is in the Layer 2 table of the switch.
- A VM interface that belongs to a VM group experiences a “source miss” i.e. is not able to learn new MAC address.
- A trusted port goes down. Port validation must be performed to ensure that the port does not get connected to an untrusted source when it comes back up.

Use the following command to set the action to be performed if the switch is unable to validate the VM MAC address:

```
RS8296(config)# virt vmcheck action basic {log|link}  
  
log - generates a log  
link - disables the port
```

Advanced Validation

This mode provides VM-based validation by mapping a switch port to a VM MAC address. It is suitable for environments in which spoofing, MAC reassignment, or MAC duplication is possible.

When the switch receives frames from a VM, it first validates the VM interface based on the VM MAC address, VM Universally Unique Identifier (UUID), Switch port, and Switch ID available in the hello message information. Only if all the four parameters are matched, the VM MAC address is considered valid.

In advanced validation mode, if the VM MAC address validation fails, an ACL can be automatically created to drop the traffic received from the VM MAC address on the switch port. Use the following command to specify the number of ACLs to be automatically created for dropping traffic:

```
RS8296(config)# virt vmcheck acls max <1-256>
```

Use the following command to set the action to be performed if the switch is unable to validate the VM MAC address:

```
RS8296(config)# virt vmcheck action advanced {log|link|acl}
```

Following are the other VMcheck commands:

Table 30. VMcheck Commands

Command	Description
RS8296(config)# virt vmware hello {enable hport <port number> haddr htimer}	Hello messages setting: enable/add port/advertise this IP address in the hello messages instead of the default management IP address/set the timer to send the hello messages
RS8296(config)# no virt vmware hello {enable hport <port number>}	Disable hello messages/remove port
RS8296(config)# [no] virt vmcheck trust <port number or range>	Mark a port as trusted; Use the no form of the command to mark port as untrusted
RS8296# no virt vmcheck acls	ACLs cannot be used for VMcheck

Virtual Distributed Switch

A virtual Distributed Switch (vDS) allows the hypervisor's NIC to be attached to the vDS instead of its own virtual switch. The vDS connects to the vCenter and spans across multiple hypervisors in a datacenter. The administrator can manage virtual machine networking for the entire data center from a single interface. The vDS enables centralized provisioning and administration of virtual machine networking in the data center using the VMware vCenter server.

When a member is added to a distributed VM group, a distributed port group is created on the vDS. The member is then added to the distributed port group.

Distributed port groups on a vDS are available to all hypervisors that are connected to the vDS. Members of a single distributed port group can communicate with each other.

Note: vDS works with ESX 4.0 or higher versions.

To add a vDS, use the command:

```
RS8296# virt vmware dvswitch add <datacenter name> <dvSwitch name>
[<dvSwitch-version>]
```

Prerequisites

Before adding a vDS on the G8296, ensure the following:

- VMware vCenter is fully installed and configured and includes a “bladevm” administration account and a valid SSL certificate.
- A virtual distributed switch instance has been created on the vCenter. The vDS version must be higher or the same as the hypervisor version on the hosts.
- At least two hypervisors are configured.

Guidelines

Before migrating VMs to a vDS, consider the following:

- At any one time, a VM NIC can be associated with only one virtual switch: to the hypervisor's virtual switch, or to the vDS.
- Management connection to the server must be ensured during the migration. The connection is via the Service Console or the Kernel/Management Interface.
- The vDS configuration and migration can be viewed in vCenter at the following locations:
 - o vDS: Home > Inventory > Networking
 - o vDS Hosts: Home > Inventory > Networking > vDS > Hosts

Note: These changes will not be displayed in the running configuration on the G8296.

Migrating to vDS

You can migrate VMs to the vDS using vCenter. The migration may also be accomplished using the operational commands on the G8296 available in the following CLI menus:

For VMware vDS operations:

```
RS8296# virt vmware dvswitch ?
add                Add a dvSwitch to a DataCenter
addhost            Add a host to a dvSwitch
adduplnk           Add a physical NIC to dvSwitch uplink ports
del                Remove a dvSwitch from a DataCenter
remhost            Remove a host from a dvSwitch
remuplnk           Remove a physical NIC from dvSwitch uplink ports
```

For VMware distributed port group operations:

```
RS8296# virt vmware dpg ?
add                Add a port group to a dvSwitch
del                Delete a port group from a dvSwitch
update             Update a port group on a dvSwitch
vmac               Change a VM NIC's port group
```

Virtualization Management Servers

The G8296 can connect with a virtualization management server to collect configuration information about associated VEs. The switch can also automatically push VM group configuration profiles to the virtualization management server, which in turn configures the hypervisors and VEs, providing enhanced VE mobility.

One virtual management server must be assigned on the switch before distributed VM groups may be used. N/OS 8.2 currently supports only the VMware Virtual Center (vCenter).

Assigning a vCenter

Assigning a vCenter to the switch requires the following:

- The vCenter must have a valid IPv4 address which is accessible to the switch (IPv6 addressing is not supported for the vCenter).
- A user account must be configured on the vCenter to provide access for the switch. The account must have (at a minimum) the following vCenter user privileges:
 - Network
 - Host Network > Configuration
 - Virtual Machine > Modify Device Settings

Once vCenter requirements are met, the following configuration command can be used on the G8296 to associate the vCenter with the switch:

```
RS8296(config)# virt vmware vcspec <vCenter IPv4 address> <username> [noauth]
```

This command specifies the IPv4 address and account username that the switch will use for vCenter access. Once entered, the administrator will be prompted to enter the password for the specified vCenter account.

The `noauth` option causes the switch to ignore SSL certificate authentication. This is required when no authoritative SSL certificate is installed on the vCenter.

Note: By default, the vCenter includes only a self-signed SSL certificate. If using the default certificate, the `noauth` option is required.

Once the vCenter configuration has been applied on the switch, the G8296 will connect to the vCenter to collect VE information.

vCenter Scans

Once the vCenter is assigned, the switch will periodically scan the vCenter to collect basic information about all the VEs in the datacenter, and more detailed information about the local VEs that the switch has discovered attached to its own ports.

The switch completes a vCenter scan approximately every two minutes. Any major changes made through the vCenter may take up to two minutes to be reflected on the switch. However, you can force an immediate scan of the vCenter by using one of the following ISCLI privileged EXEC commands:

RS8296# virt vmware scan	<i>(Scan the vCenter)</i>
<i>-or-</i>	
RS8296# show virt vm -v -r	<i>(Scan vCenter and display result)</i>

Deleting the vCenter

To detach the vCenter from the switch, use the following configuration command:

RS8296(config)# no virt vmware vcspec
--

Note: Without a valid vCenter assigned on the switch, any VE configuration changes must be manually synchronized.

Deleting the assigned vCenter prevents synchronizing the configuration between the G8296 and VEs. VEs already operating in distributed VM groups will continue to function as configured, but any changes made to any VM profile or distributed VM group on the switch will affect only switch operation; changes on the switch will not be reflected in the vCenter or on the VEs. Likewise, any changes made to VE configuration on the vCenter will no longer be reflected on the switch.

Exporting Profiles

VM profiles for discovered VEs in distributed VM groups are automatically synchronized with the virtual management server and the appropriate hypervisors. However, VM profiles can also be manually exported to specific hosts before individual VEs are defined on them.

By exporting VM profiles to a specific host, virtual machine port groups will be available to the host's internal virtual switches so that new VMs may be configured to use them.

VM migration requires that the target hypervisor includes all the virtual switch port groups to which the VM connects on the source hypervisor. The VM profile export feature can be used to distribute the associated port groups to all the potential hosts for a given VM.

A VM profile can be exported to a host using the following ISCLI privileged EXEC command:

```
RS8296# virt vmware export <VM profile name> <host list> <virtual switch name>
```

The host list can include one or more target hosts, specified by host name, IPv4 address, or UUID, with each list item separated by a space.

Once executed, the requisite port group will be created on the specified virtual switch. If the specified virtual switch does not exist on the target host, the port group will not be created.

VMware Operational Commands

The G8296 may be used as a central point of configuration for VMware virtual switches and port groups using the following ISCLI privileged EXEC commands:

```
RS8296# virt vmware ?
dpg          Distributed port group operations
dvswitch     VMWare dvSwitch operations
export       Create or update a vm profile on one host
pg           Add a port group to a host
scan         Perform a VM Agent scan operation now
updpg        Update a port group on a host
vmacpg       Change a vnic's port group
vsw          Add a vswitch to a host
```

Pre-Provisioning VEs

VEs may be manually added to VM groups in advance of being detected on the switch ports. By pre-provisioning the MAC address of VEs that are not yet active, the switch will be able to later recognize the VE when it becomes active on a switch port, and immediately assign the proper VM group properties without further configuration.

Undiscovered VEs are added to or removed from VM groups using the following configuration commands:

```
RS8296(config)# [no] virt vgroup <VM group number> vm <VE MAC address>
```

For the pre-provisioning of undiscovered VEs, a MAC address is required. Other identifying properties, such as IPv4 address or VM name permitted for known VEs, cannot be used for pre-provisioning.

VLAN Maps

A VLAN map (VMAP) is a type of Access Control List (ACL) that is applied to a VLAN or VM group rather than to a switch port as with regular ACLs (see [“Access Control Lists” on page 103](#)). In a virtualized environment, VMAPs allow you to create traffic filtering and metering policies that are associated with a VM group VLAN, allowing filters to follow VMs as they migrate between hypervisors.

N/OS 8.2 supports up to 128 VMAPs.

Individual VMAP filters are configured in the same fashion as regular ACLs, except that VLANs cannot be specified as a filtering criteria (unnecessary, since VMAPs are assigned to a specific VLAN or associated with a VM group VLAN).

VMAPs are configured using the following ISCLI configuration command path:

```
RS8296(config)# access-control vmap <VMAP ID> ?
  action          Set filter action
  egress-port     Set to filter for packets egressing this port
  ethernet        Ethernet header options
  ipv4            IP version 4 header options
  meter           ACL metering configuration
  mirror          Mirror options
  packet-format   Set to filter specific packet format types
  re-mark         ACL re-mark configuration
  statistics      Enable access control list statistics
  tcp-udp         TCP and UDP filtering options
```

Once a VMAP filter is created, it can be assigned or removed using the following commands:

- For regular VLANs, use config-vlan mode:

```
RS8296(config)# vlan <VLAN ID>
RS8296(config-vlan)# [no] vmap <VMAP ID> [serverports |
non-serverports]
```

- For a VM group, use the global configuration mode:

```
RS8296(config)# [no] virt vmgroup <ID> vmap <VMAP ID>
[serverports | non-serverports]
```

Note: Each VMAP can be assigned to only one VLAN or VM group. However, each VLAN or VM group may have multiple VMAPs assigned to it.

The optional `serverports` or `non-serverports` parameter can be specified to apply the action (to add or remove the VMAP) for either the switch server ports (`serverports`) or switch uplink ports (`non-serverports`). If omitted, the operation will be applied to all ports in the associated VLAN or VM group.

Note: VMAPs have a lower priority than port-based ACLs. If both an ACL and a VMAP match a particular packet, both filter actions will be applied as long as there is no conflict. In the event of a conflict, the port ACL will take priority, though switch statistics will count matches for both the ACL and VMAP.

VM Policy Bandwidth Control

In a virtualized environment where VEs can migrate between hypervisors and thus move among different ports on the switch, traffic bandwidth policies must be attached to VEs, rather than to a specific switch port.

VM Policy Bandwidth Control allows the administrator to specify the amount of data the switch will permit to flow from a particular VE, without defining a complicated matrix of ACLs or VMAPs for all port combinations where a VE may appear.

VM Policy Bandwidth Control Commands

VM Policy Bandwidth Control can be configured using the following configuration commands:

```
RS8296(config)# virt vmpolicy vmbwidth <VM MAC>|<index>|<UUID>| <IPv4  
address>|<name>?  
txrate <committed rate> <burst> [<ACL number>] (Set the VM transmit bandwidth –  
ingress for switch)  
rxrate <committed rate> <burst> (Set the VM receive bandwidth – egress for switch)  
bwctrl (Enable bandwidth control)
```

Bandwidth allocation can be defined for transmit (TX) traffic or receive (RX) traffic. Because bandwidth allocation is specified from the perspective of the VE, the switch command for TX Rate Control (`txrate`) sets the data rate to be sent from the VM to the switch, and the RX rate sets the data rate to be received by the VM from the switch.

The *committed rate* is specified in multiples of 64 kbps, from 64 to 10,000,000. The maximum *burst* rate is specified as 32, 64, 128, 256, 1024, 2048, or 4096 kb. If both the committed rate and burst are set to 0, bandwidth control will be disabled.

When `txrate` is specified, the switch automatically selects an available ACL for internal use with bandwidth control. Optionally, if automatic ACL selection is not desired, a specific ACL may be selected. If there are no unassigned ACLs available, `txrate` cannot be configured.

Bandwidth Policies vs. Bandwidth Shaping

VM Profile Bandwidth Shaping differs from VM Policy Bandwidth Control.

VM Profile Bandwidth Shaping (see [“VM Profiles” on page 245](#)) is configured per VM group and is enforced on the server by a virtual switch in the hypervisor. Shaping is unidirectional and limits traffic transmitted from the virtual switch to the G8296. Shaping is performed prior to transmit VM Policy Bandwidth Control. If the egress traffic for a virtual switch port group exceeds shaping parameters, the traffic is dropped by the virtual switch in the hypervisor. Shaping uses server CPU resources, but prevents extra traffic from consuming bandwidth between the server and the G8296.

VM Policy Bandwidth Control is configured per VE, and can be set independently for transmit traffic. Bandwidth policies are enforced by the G8296. VE traffic that exceeds configured levels is dropped by the switch upon ingress. Setting `txrate` uses ACL resources on the switch.

Bandwidth shaping and bandwidth policies can be used separately or in concert.

VMready Information Displays

The G8296 can be used to display a variety of VMready information.

Note: Some displays depict information collected from scans of a VMware vCenter and may not be available without a valid vCenter. If a vCenter is assigned (see [“Assigning a vCenter” on page 252](#)), scan information might not be available for up to two minutes after the switch boots or when VMready is first enabled. Also, any major changes made through the vCenter may take up to two minutes to be reflected on the switch unless you force an immediate vCenter scan (see [“vCenter Scans” on page 252](#)).

Local VE Information

A concise list of local VEs and pre-provisioned VEs is available with the following ISCLI privileged EXEC command:

```
RS8296# show virt vm
```

IP Address	VMAC Address	Index	Port	VM Group (Profile)
*172.16.46.50	00:50:56:4e:62:00	4	3	
*172.16.46.10	00:50:56:4f:f2:00	2	4	
+172.16.46.51	00:50:56:72:ec:00	1	3	
+172.16.46.11	00:50:56:7c:1c:00	3	4	
172.16.46.25	00:50:56:9c:00:00	5	4	
172.16.46.15	00:50:56:9c:21:00	0	4	
172.16.46.35	00:50:56:9c:29:00	6	3	
172.16.46.45	00:50:56:9c:47:00	7	3	

Number of entries: 8
* indicates VMware ESX Service Console Interface
+ indicates VMware ESX/ESXi VMKernel or Management Interface

Note: The Index numbers shown in the VE information displays can be used to specify a particular VE in configuration commands.

If a vCenter is available, more verbose information can be obtained using the following ISCLI privileged EXEC command option:

```
RS8296# show virt vm -v
```

Index	MAC Address, IP Address	Name (VM or Host), @Host (VMs only)	Port, VLAN	Group	Vswitch, Port Group
0	00:50:56:9c:21:2f 172.16.46.15	atom @172.16.46.10	4 500		vSwitch0 Eng_A
+1	00:50:56:72:ec:86 172.16.46.51	172.16.46.50	3 0		vSwitch0 VMkernel
*2	00:50:56:4f:f2:85 172.16.46.10	172.16.46.10	4 0		vSwitch0 Mgmt
+3	00:50:56:7c:1c:ca 172.16.46.11	172.16.46.10	4 0		vSwitch0 VMkernel
*4	00:50:56:4e:62:f5 172.16.46.50	172.16.46.50	3 0		vSwitch0 Mgmt
5	00:50:56:9c:00:c8 172.16.46.25	quark @172.16.46.10	4 0		vSwitch0 Corp
6	00:50:56:9c:29:29 172.16.46.35	particle @172.16.46.50	3 0		vSwitch0 VM Network
7	00:50:56:9c:47:fd 172.16.46.45	nucleus @172.16.46.50	3 0		vSwitch0 Finance

```
--
12 of 12 entries printed
* indicates VMware ESX Service Console Interface
+ indicates VMware ESX/ESXi VMkernel or Management Interface
```

To view additional detail regarding any specific VE, see [“vCenter VE Details” on page 262](#)).

vCenter Hypervisor Hosts

If a vCenter is available, the following ISCLI privileged EXEC command displays the name and UUID of all VMware hosts, providing an essential overview of the data center:

```
RS8296# show virt vmware hosts
```

UUID	Name(s), IP Address
00a42681-d0e5-5910-a0bf-bd23bd3f7800	172.16.41.30
002e063c-153c-dd11-8b32-a78dd1909a00	172.16.46.10
00f1fe30-143c-dd11-84f2-a8ba2cd7ae00	172.16.44.50
0018938e-143c-dd11-9f7a-d8defa4b8300	172.16.46.20
...	

Using the following command, the administrator can view more detailed vCenter host information, including a list of virtual switches and their port groups, as well as details for all associated VEs:

```

RS8296# show virt vmware showhost {<UUID>|<IPv4 address>|<host name>}
Vswitches available on the host:
        vSwitch0
Port Groups and their Vswitches on the host:
        Lenovo_Default          vSwitch0
        VM Network              vSwitch0
        Service Console         vSwitch0
        VMkernel                vSwitch0
-----
MAC Address      00:50:56:9c:21:2f
Port            4
Type            Virtual Machine
VM vCenter Name halibut
VM OS hostname  localhost.localdomain
VM IP Address   172.16.46.15
VM UUID        001c41f3-ccd8-94bb-1b94-6b94b03b9200
Current VM Host 172.16.46.10
Vswitch        vSwitch0
Port Group     Lenovo_Default
VLAN ID       0
...

```

vCenter VEs

If a vCenter is available, the following ISCLI privileged EXEC command displays a list of all known VEs:

```

RS8296# show virt vmware vms
UUID                                     Name(s), IP Address
-----
001cdf1d-863a-fa5e-58c0-d197ed3e3300   30vm1
001c1fba-5483-863f-de04-4953b5caa700   VM90
001c0441-c9ed-184c-7030-d6a6bc9b4d00   VM91
001cc06e-393b-a36b-2da9-c71098d9a700   vm_new
001c6384-f764-983c-83e3-e94fc78f2c00   sturgeon
001c7434-6bf9-52bd-c48c-a410da0c2300   VM70
001cad78-8a3c-9cbe-35f6-59ca5f392500   VM60
001cf762-a577-f42a-c6ea-090216c11800   30VM6
001c41f3-ccd8-94bb-1b94-6b94b03b9200   halibut, localhost.localdomain,
                                         172.16.46.15
001cf17b-5581-ea80-c22c-3236b89ee900   30vm5
001c4312-a145-bf44-7edd-49b7a2fc3800   vm3
001caf40-a40a-de6f-7b44-9c496f123b00   30VM7

```

vCenter VE Details

If a vCenter is available, the following ISCLI privileged EXEC command displays detailed information about a specific VE:

```
RS8296# show virt vmware showvm {<VM UUID>|<VM IPv4 address>|<VM name>}
-----
MAC Address      00:50:56:9c:21:2f
Port             4
Type             Virtual Machine
VM vCenter Name  halibut
VM OS hostname   localhost.localdomain
VM IP Address    172.16.46.15
VM UUID          001c41f3-ccd8-94bb-1b94-6b94b03b9200
Current VM Host  172.16.46.10
Vswitch          vSwitch0
Port Group       Lenovo_Default
VLAN ID          0
```

VMready Configuration Example

This example has the following characteristics:

- A VMware vCenter is fully installed and configured prior to VMready configuration and includes a “bladevm” administration account and a valid SSL certificate.
- The distributed VM group model is used.
- The VM profile named “Finance” is configured for VLAN 30, and specifies NIC-to-switch bandwidth shaping for 1Mbps average bandwidth, 2MB bursts, and 3Mbps maximum bandwidth.
- The VM group includes four discovered VMs on switch server ports 1 and 2, and one static trunk (previously configured) that includes switch uplink ports 3 and 4.

1. Define the server ports.

```
RS8296(config)# system server-ports port 1-2
```

2. Enable the VMready feature.

```
RS8296(config)# virt enable
```

3. Specify the VMware vCenter IPv4 address.

```
RS8296(config)# virt vmware vmware vcspec 172.16.100.1 <user> noauth
```

When prompted, enter the user password that the switch must use for access to the vCenter.

4. Create the VM profile.

```
RS8296(config)# virt vmprofile Finance
RS8296(config)# virt vmprofile edit Finance vlan 30
RS8296(config)# virt vmprofile edit Finance shaping 1000 2000 3000
```

5. Define the VM group.

```
RS8296(config)# virt vmgroup 1 profile Finance
RS8296(config)# virt vmgroup 1 vm arctic
RS8296(config)# virt vmgroup 1 vm monster
RS8296(config)# virt vmgroup 1 vm sierra
RS8296(config)# virt vmgroup 1 vm 00:50:56:4f:f2:00
RS8296(config)# virt vmgroup 1 portchannel 1
```

When VMs are added, the server ports on which they appear are automatically added to the VM group. In this example, there is no need to manually add ports 1 and 2.

6. If necessary, enable VLAN tagging for the VM group:

```
RS8296(config)# virt vmgroup 1 tag
```

Note: If the VM group contains ports that also exist in other VM groups, make sure tagging is enabled in both VM groups. In this example configuration, no ports exist in more than one VM group.

7. Save the configuration.

Chapter 19. FCoE and CEE

This chapter provides conceptual background and configuration examples for using Converged Enhanced Ethernet (CEE) features of the RackSwitch G8296, with an emphasis on Fibre Channel over Ethernet (FCoE) solutions. The following topics are addressed in this chapter:

- [“Fibre Channel over Ethernet” on page 267](#)

Fibre Channel over Ethernet (FCoE) allows Fibre Channel traffic to be transported over Ethernet links. This provides an evolutionary approach toward network consolidation, allowing Fibre Channel equipment and tools to be retained, while leveraging cheap, ubiquitous Ethernet networks for growth.

- [“Converged Enhanced Ethernet” on page 269](#)

Converged Enhanced Ethernet (CEE) refers to a set of IEEE standards developed primarily to enable FCoE, requiring enhancing the existing Ethernet standards to make them lossless on a per-priority traffic basis, and providing a mechanism to carry converged (LAN/SAN/IPC) traffic on a single physical link. CEE features can also be utilized in traditional LAN (non-FCoE) networks to provide lossless guarantees on a per-priority basis, and to provide efficient bandwidth allocation.

- [“FCoE Initialization Protocol Snooping” on page 272](#)

Using FCoE Initialization Protocol (FIP) snooping, the G8296 examines the FIP frames exchanged between ENodes and FCoE Forwarders (FCFs). This information is used to dynamically determine the ACLs required to block certain types of undesired or unvalidated traffic on FCoE links.

- [“FCoE Forwarder” on page 277](#)

An FCoE Forwarder (FCF) authenticates connecting neighbors, provides Fibre Channel IDs, enforce port security among zones, and informs neighboring devices of network changes.

- [“Priority-Based Flow Control” on page 283](#)

Priority-Based Flow Control (PFC) extends 802.3x standard flow control to allow the switch to pause traffic based on the 802.1p priority value in each packet’s VLAN tag. PFC is vital for FCoE environments, where SAN traffic must remain lossless and must be paused during congestion, while LAN traffic on the same links is delivered with “best effort” characteristics.

- [“Enhanced Transmission Selection” on page 286](#)

Enhanced Transmission Selection (ETS) provides a method for allocating link bandwidth based on the 802.1p priority value in each packet’s VLAN tag. Using ETS, different types of traffic (such as LAN, SAN, and management) that are sensitive to different handling criteria can be configured either for specific bandwidth characteristics, low-latency, or best-effort transmission, despite sharing converged links as in an FCoE environment.

- [“Data Center Bridging Capability Exchange” on page 292](#)

Data Center Bridging Capability Exchange Protocol (DCBX) allows neighboring network devices to exchange information about their capabilities. This is used between CEE-capable devices for the purpose of discovering their peers, negotiating peer configurations, and detecting misconfigurations.

Fibre Channel over Ethernet

Fibre Channel over Ethernet (FCoE) is an effort to converge two of the different physical networks in today's data centers. It allows Fibre Channel traffic (such as that commonly used in Storage Area Networks, or SANs) to be transported without loss over 10Gb Ethernet links (typically used for high-speed Local Area Networks, or LANs). This provides an evolutionary approach toward network consolidation, allowing Fibre Channel equipment and tools to be retained, while leveraging cheap, ubiquitous Ethernet networks for growth.

With server virtualization, servers capable of hosting both Fibre Channel and Ethernet applications will provide advantages in server efficiency, particularly as FCoE-enabled network adapters provide consolidated SAN and LAN traffic capabilities.

The RackSwitch G8296 with Lenovo Network Operating System 8.2 software is compliant with the INCITS T11.3, FC-BB-5 FCoE specification.

Note: The G8296 supports up to 2048 FCoE login sessions.

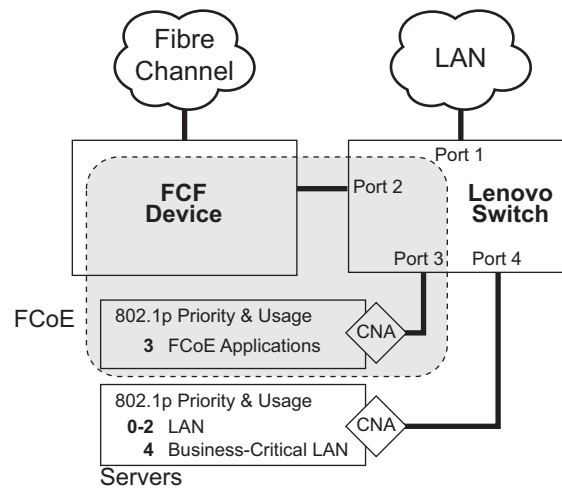
The FCoE Topology

Note: The FCoE topology discussed in this section applies to FCoE Initialization Protocol (FIP) snooping only.

In an end-to-end Fibre Channel network, switches and end devices generally establish trusted, point-to-point links. Fibre Channel switches validate end devices, enforce zoning configurations and device addressing, and prevent certain types of errors and attacks on the network.

In a converged multi-hop FCoE network where Fibre Channel devices are bridged to Ethernet devices, the direct point-to-point QoS capabilities normally provided by the Fibre Channel fabric may be lost in the transition between the different network types. The G8296 FIP snooping bridge provides a solution to overcome this.

Figure 24. A Mixed Fibre Channel and FCoE Network



In [Figure 24 on page 267](#), the Fibre Channel network is connected to the FCoE network through an FCoE Forwarder (FCF). The FCF acts as a Fibre Channel gateway to and from the FCoE network.

For the FCoE portion of the network, the FCF is connected to the FCoE-enabled G8296, which is connected to a server (running Fibre Channel applications) through an FCoE-enabled Converged Network Adapter (CNA) known in Fibre Channel as Ethernet Nodes (ENodes).

Note: The figure also shows a non-FCoE LAN server connected to the G8296 using a CNA. This allows the LAN server to take advantage of some CEE features that are useful even outside of an FCoE environment.

To block undesired or unvalidated traffic on FCoE links that exists outside the regular Fibre Channel topology, Ethernet ports used in FCoE are configured with Access Control Lists (ACLs) that are narrowly tailored to permit expected FCoE traffic to and from confirmed FCFs and ENodes, and deny all other FCoE or FIP traffic. This ensures that all FCoE traffic to and from the ENode passes through the FCF.

Because manual ACL configuration is an administratively complex task, the G8296 can automatically and dynamically configure the ACLs required for use with FCoE. Using FCoE Initialization Protocol (FIP) snooping (see [“FCoE Initialization Protocol Snooping” on page 272](#)), the G8296 examines the FIP frames normally exchanged between the FCF and ENodes to determine information about connected FCoE devices. This information is used to automatically determine the appropriate ACLs required to block certain types of undesired or unvalidated FCoE traffic.

Automatic FCoE-related ACLs are independent from ACLs used for typical Ethernet purposes.

FIP Snooping Requirements

The following are required for implementing the FIP snooping bridge feature using the RackSwitch G8296 (G8296) with N/OS 8.2 software:

- The G8296, when configured as an FIP snooping bridge (see [“FCoE Initialization Protocol Snooping” on page 272](#)), must be connected to the Fibre Channel network through an FCF such as a Lenovo RackSwitch G8296 or a Cisco Nexus 5000 Series Switch.
- For each G8296 port participating in FCoE, the connected server must use a FCoE-licensed CNA.
- CEE must be turned on (see [“Turning CEE On or Off” on page 269](#)). When CEE is on, the DCBX, PFC, and ETS features are enabled and configured with default FCoE settings. These features may be reconfigured, but must remain enabled for FCoE to function.
- FIP snooping must be turned on (see [“FCoE Initialization Protocol Snooping” on page 272](#)). When FIP snooping is turned on, the feature is enabled on all ports by default. The administrator can disable FIP snooping on individual ports that do not require FCoE, but FIP snooping must remain enabled on all FCoE ports for FCoE to function.

Converged Enhanced Ethernet

Converged Enhanced Ethernet (CEE) refers to a set of IEEE standards designed to allow different physical networks with different data handling requirements to be converged together, simplifying management, increasing efficiency and utilization, and leveraging legacy investments without sacrificing evolutionary growth.

CEE standards were developed primarily to enable Fibre Channel traffic to be carried over Ethernet networks. This required enhancing the existing Ethernet standards to make them lossless on a per-priority traffic basis, and to provide a mechanism to carry converged (LAN/SAN/IPC) traffic on a single physical link. Although CEE standards were designed with FCoE in mind, they are not limited to FCoE installations. CEE features can be utilized in traditional LAN (non-FCoE) networks to provide lossless guarantees on a per-priority basis, and to provide efficient bandwidth allocation based on application needs.

Turning CEE On or Off

By default on the G8296, CEE is turned off. To turn CEE on or off, use the following CLI commands:

```
RS8296(config)# [no] cee enable
```



CAUTION:

Turning CEE on will automatically change some 802.1p QoS and 802.3x standard flow control settings on the G8296. Read the following material carefully to determine whether you will need to take action to reconfigure expected settings.

It is recommended that you backup your configuration prior to turning CEE on. Viewing the file will allow you to manually re-create the equivalent configuration once CEE is turned on, and will also allow you to recover your prior configuration if you need to turn CEE off.

Effects on Link Layer Discovery Protocol

When CEE is turned on, Link Layer Discovery Protocol (LLDP) is automatically turned on and enabled for receiving and transmitting DCBX information. LLDP cannot be turned off while CEE is turned on.

Effects on 802.1p Quality of Service

While CEE is off (the default), the G8296 allows 802.1p priority values to be used for Quality of Service (QoS) configuration (see [page 189](#)). 802.1p QoS default settings are shown in [Table 31](#), but can be changed by the administrator.

When CEE is turned on, 802.1p QoS is replaced by ETS (see [“Enhanced Transmission Selection” on page 286](#)). As a result, while CEE is turned on, the 802.1p QoS configuration commands are no longer available on the switch (the menu is restored when CEE is turned off).

In addition, when CEE is turned on, prior 802.1p QoS settings are replaced with new defaults designed for use with ETS priority groups (PGIDs) as shown in [Table 31](#):

Table 31. *CEE Effects on 802.1p Defaults*

802.1p QoS Configuration With CEE Off (default)			ETS Configuration With CEE On		
Priority-	COSq	Weight	Priority	COSq	PGID
0	0	1	0	0	0
1	1	2	1	0	0
2	2	3	2	0	0
3	3	4	3	1	1
4	4	5	4	2	2
5	5	7	5	2	2
6	6	15	6	2	2
7	7	0	7	2	2

When CEE is on, the default ETS configuration also allocates a portion of link bandwidth to each PGID as shown in [Table 32](#):

Table 32. *Default ETS Bandwidth Allocation*

PGID	Typical Use	Bandwidth
0	LAN	10%
1	SAN	50%
2	Latency-sensitive LAN	40%

If the prior, non-CEE configuration used 802.1p priority values for different purposes, or does not expect bandwidth allocation as shown in [Table 32 on page 270](#), when CEE is turned on, have the administrator reconfigure ETS settings as appropriate.

It is recommended that a configuration backup be made prior to turning CEE on or off. Viewing the configuration file will allow the administrator to manually re-create the equivalent configuration under the new CEE mode, and will also allow for the recovery of the prior configuration if necessary.

Effects on Flow Control

When CEE is turned on, standard flow control is disabled on all ports, and in its place, PFC (see [“Priority-Based Flow Control” on page 283](#)) is enabled on all ports for 802.1p priority value 3. This default is chosen because priority value 3 is commonly used to identify FCoE traffic in a CEE environment and must be guaranteed lossless behavior. PFC is disabled for all other priority values.

Each time CEE is turned off, the prior 802.3x standard flow control settings will be restored (including any previous changes from the defaults).

It is recommend that a configuration backup be made prior to turning CEE on or off. Viewing the configuration file will allow the administrator to manually re-create the equivalent configuration under the new CEE mode, and will also allow for the recovery of the prior configuration if necessary.

When CEE is on, PFC can be enabled only on priority value 3 and one other priority. If flow control is required on additional priorities on any given port, consider using standard flow control on that port, so that regardless of which priority traffic becomes congested, a flow control frame is generated.

FCoE Initialization Protocol Snooping

FCoE Initialization Protocol (FIP) snooping is an FCoE feature. To enforce point-to-point links for FCoE traffic outside the regular Fibre Channel topology, Ethernet ports used in FCoE can be automatically and dynamically configured with Access Control Lists (ACLs).

Using FIP snooping, the G8296 examines the FIP frames normally exchanged between the FCF and ENodes to determine information about connected FCoE devices. This information is used to create narrowly tailored ACLs that permit expected FCoE traffic to and from confirmed Fibre Channel nodes, and deny all other undesirable FCoE or FIP traffic.

Global FIP Snooping Settings

By default, the FIP snooping feature is turned off for the G8296. The following commands are used to turn the feature on or off:

```
RS8296(config)# [no] fcoe fips enable
```

Note: FIP snooping requires CEE to be turned on (see [“Turning CEE On or Off” on page 269](#)).

When FIP snooping is on, port participation may be configured on a port-by-port basis (see the next sections).

When FIP snooping is off, all FCoE-related ACLs generated by the feature are removed from all switch ports.

FIP Snooping for Specific Ports

When FIP snooping is globally turned on (see the previous section), ports may be individually configured for participation in FIP snooping and automatic ACL generation. By default, FIP snooping is enabled for each port. To change the setting for any specific port, use the following CLI commands:

```
RS8296(config)# [no] fcoe fips port <port alias, number, list, or range> enable
```

When FIP snooping is enabled on a port, FCoE-related ACLs will be automatically configured.

When FIP snooping is disabled on a port, all FCoE-related ACLs on the port are removed, and the switch will enforce no FCoE-related rules for traffic on the port.

Port FCF and ENode Detection

When FIP snooping is enabled on a port, the port is placed in FCF auto-detect mode by default. In this mode, the port assumes connection to an ENode unless FIP packets show the port is connected to an FCF.

Ports can also be specifically configured as to whether automatic FCF detection will be used, or whether the port is connected to an FCF or ENode:

```
RS8296(config)# fcoe fips port <port alias, number, list, or range> fcf-mode {auto|on|off}
```

When FCF mode is `on`, the port is assumed to be connected to a trusted FCF, and only ACLs appropriate to FCFs will be installed on the port. When `off`, the port is assumed to be connected to an ENode, and only ACLs appropriate to ENodes will be installed. When the mode is changed (either through manual configuration or as a result of automatic detection), the appropriate ACLs are automatically added, removed, or changed to reflect the new FCF or ENode connection.

FCoE Connection Timeout

FCoE-related ACLs are added, changed, and removed as FCoE device connection and disconnection are discovered. In addition, the administrator can enable or disable automatic removal of ACLs for FCFs and other FCoE connections that timeout (fail or are disconnected) without FIP notification.

By default, automatic removal of ACLs upon timeout is enabled. To change this function, use the following CLI command:

```
RS8296(config)# [no] fcoe fips timeout-acl
```

FCoE ACL Rules

When FIP Snooping is enabled on a port, the switch automatically installs the appropriate ACLs to enforce the following rules for FCoE traffic:

- Ensure that FIP frames from ENodes may only be addressed to FCFs.
- Flag important FIP packets for switch processing.
- Ensure no end device uses an FCF MAC address as its source.
- Each FCoE port is assumed to be connected to an ENode and include ENode-specific ACLs installed, until the port is either detected or configured to be connected to an FCF.
- Ports that are configured to have FIP snooping disabled will not have any FIP or FCoE related ACLs installed.
- Prevent transmission of all FCoE frames from an ENode prior to its successful completion of login (FLOGI) to the FCF.
- After successful completion of FLOGI, ensure that the ENode uses only those FCoE source addresses assigned to it by FCF.
- After successful completion of FLOGI, ensure that all ENode FCoE source addresses originate from or are destined to the appropriate ENode port.
- After successful completion of each FLOGI, ensure that FCoE frames may only be addressed to the FCFs that accept them.

Initially, a basic set of FCoE-related ACLs will be installed on all ports where FIP snooping is enabled. As the switch encounters FIP frames and learns about FCFs and ENodes that are attached or disconnect, ACLs are dynamically installed or expanded to provide appropriate security.

When an FCoE connection logs out, or times out (if ACL timeout is enabled), the related ACLs will be automatically removed.

FCoE-related ACLs are independent of manually configured ACLs used for regular Ethernet purposes (see [“Access Control Lists” on page 103](#)). FCoE ACLs generally have a higher priority over standard ACLs, and do not inhibit non-FCoE and non-FIP traffic.

FCoE VLANs

FCoE packets to any FCF will be confined to the VLAN advertised by the FCF (typically VLAN 1002). The appropriate VLAN must be configured on the switch with member FCF ports and must be supported by the participating CNAs.

Viewing FIP Snooping Information

ACLs automatically generated under FIP snooping are independent of regular, manually configure ACLs, and are not listed with regular ACLs in switch information and statistics output. Instead, FCoE ACLs are shown using the following CLI commands:

```
RS8296# show fcoe fips information (Show all FIP-related information)
RS8296# show fcoe fips port <ports> information (Show FIP info for a selected port)
```

For example:

```
RS8296# show fcoe fips port 21 information

FIP Snooping on port 21:
This port has been configured to automatically detect FCF.
It has currently detected to have 0 FCF connecting to it.

FIPS ACLs configured on this port:
SMAC 00:05:73:ce:96:67, action deny.
DMAC 00:05:73:ce:96:67, ethertype 0x8914, action permit.
SMAC 0e:fc:00:44:04:04, DMAC 00:05:73:ce:96:67, ethertype 0x8906, vlan
1002, action permit.
DMAC 01:10:18:01:00:01, Ethertype 0x8914, action permit.
DMAC 01:10:18:01:00:02, Ethertype 0x8914, action permit.
Ethertype 0x8914, action deny.
Ethertype 0x8906, action deny.
SMAC 0e:fc:00:00:00:00, SMAC mask ff:ff:ff:00:00:00, action deny.
```

For each ACL, the required traffic criteria are listed, along with the action taken (permit or deny) for matching traffic. ACLs are listed in order of precedence and evaluated in the order shown.

The administrator can also view other FCoE information:

```
RS8296# show fcoe fips fcf           (Show all detected FCFs)
RS8296# show fcoe fips fcoe         (Show all FCoE connections)
```

Operational Commands

The administrator may use the operational commands to delete FIP-related entries from the switch.

To delete a specific FCF entry and all associated ACLs from the switch, use the following command:

```
RS8296# no fcoe fips fcf <FCF MAC address> [<VLAN number>]
```

FIP Snooping Configuration

In this example, as shown in [Figure 24 on page 267](#), FCoE devices are connected to port 2 for the FCF device, and port 3 for an ENode. FIP snooping can be configured on these ports using the following ISCLI commands:

1. Enable VLAN tagging on the FCoE ports:

```
RS8296(config)# interface port 2,3   (Select FCoE ports)
RS8296(config-if)# switchport mode trunk(Enable VLAN tagging)
RS8296(config-if)# exit              (Exit port configuration mode)
```

2. Place FCoE ports into a VLAN supported by the FCF and CNAs (typically VLAN 1002):

```
RS8296(config)# vlan 1002           (Select a VLAN)
RS8296(config-vlan)# exit           (Exit VLAN configuration mode)
RS8296(config)# interface port 2,3  (Add FCoE ports to the VLAN)
RS8296(config-if)# switchport trunk allowed vlan add 1002
RS8296(config-if)# exit
```

3. Turn CEE on.

```
RS8296(config)# cee enable
```

Note: Turning CEE on will automatically change some 802.1p QoS and 802.3x standard flow control settings and menus (see [“Turning CEE On or Off” on page 269](#)).

4. Turn global FIP snooping on:

```
RS8296(config)# fcoe fips enable
```

5. Enable FIP snooping on FCoE ports, and set the desired FCF mode:

```
RS8296(config)# fcoe fips port 2 enable   (Enable FIPS on port 2)
RS8296(config)# fcoe fips port 2 fcf-mode on   (Set as FCF connection)
RS8296(config)# fcoe fips port 3 enable   (Enable FIPS on port 3)
RS8296(config)# fcoe fips port 3 fcf-mode off (Set as ENode connection)
```

Note: By default, FIP snooping is enabled on all ports and the FCF mode set for automatic detection. The configuration in this step is unnecessary, if default settings have not been changed, and is shown merely as a manual configuration example.

6. Save the configuration.

FCoE Forwarder

The RackSwitch G8296 has an integrated FC-BB-5 compliant FCoE Forwarder (FCF), allowing it to take the role of a full fabric FCoE switch. As a full fabric FCoE switch, the G8296 authenticates connecting neighbors, provides Fibre Channel IDs, enforces port security among zones, and informs neighboring devices of network changes. In FCF mode, the G8296 cannot be connected directly to another FCF switch in the same VLAN.

Configuring FCF

The switch's role as a full fabric switch in the FCoE network can be defined on a per-VLAN basis. The G8296 supports up to 12 FCF VLANs. To enable or disable FCF for a specific VLAN:

```
RS8296(config)# vlan <VLAN number>
RS8296(config-vlan)# [no] fcf enable
```

To connect an FCoE device or enode to a port of a G8296, the following conditions must be met:

- The port must be in trunk mode.
- The port must be configured as a Spanning Tree port.
- The FCoE VLAN must be included in the trunk allowed VLANs list.
- The FCoE VLAN must not be the native VLAN of the trunk port.

For example, to make port 2, which is connected to an FCoE device, part of FCF-enabled vlan 1002, use the following commands:

```
RS8296(config)# vlan 1002
RS8296(config-vlan)# fcf enable
RS8296(config-vlan)# exit
RS8296(config)# interface port 2
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# spanning-tree portfast
RS8296(config-if)# switchport trunk allowed vlan add 1002
RS8296(config-if)# exit
```

Fibre Channel Services

The following Fibre Channel (FC) services are supported:

- Name Server (Directory Server) – provides a means to discover information about ENodes connected to the FCF. This Service is provided through Well Known Address (WKA) FFFFFCh.
- Zone Server – provides partitioning of a SAN fabric into smaller subsets to restrict interference, add security, and simplify management.
- Fabric Login – assigns a fabric address (FCID) to the ENode. This service is provided through WKA FFFFFEh.
- Fabric Controller – provides State Change Notifications (SCNs) to registered ENodes whenever there is a change in the fabric topology.

- Fabric Configuration Server (FCS) – provides a way for a management application to discover fabric attributes. This Service is provided through WKA FFFFAh.
- Fabric-Device Management Interface (FDMI) – enables management for devices such as CNAs through the fabric.

FC services are provided as Class 3 (Unacknowledged connectionless services) only.

FCF VLAN Configuration

For a FCF VLAN default FCoE parameters will be set once it is enabled. The default values are:

- FC-MAP : 0x0efc00
- FCF-Priority : 128
- FKA-Adv : 8000

If you want to change these values, use the following commands to configure VLAN settings for FCF:

1. Select the FCF vlan:

```
RS8296(config)# vlan <VLAN number>
```

2. Configure the FCF global fabric priority:

```
RS8296(config-vlan)# fcoe fcf-priority <0 - 255>
```

3. Define an FC map to uniquely identify the FCF:

```
RS8296(config-vlan)# fcoe fcmmap <0xefc00 - 0xefeff>
```

4. Configure the FIP keep-alive duration in ms:

```
RS8296(config-vlan)# fcoe fka-adv-period <4000 - 90000>
```

5. Check FCF Vlan Configuration:

```
RS8296(config)# show fcf vlan <VLAN number>
```

Full Fabric Zoning

The G8296 supports FCoE zones and zonesets for VLANs operating in full fabric mode. Zoning allows logical grouping of ports and storage devices within a storage area network. Zoning defines access control between groups of servers and storage devices.

An SAN is typically divided into zones and zonesets, as described in the following sections.

Zones

A *zone* is a logical grouping of end nodes that are permitted to interact with each other. Zones can be grouped into zonesets that can be activated or deactivated as a single entity. A zone provides security by restricting access to only those devices that reside within the zone. Zoning also confines change notification floods within each zone.

Each zone contains one or more servers and one or more storage devices. Ports and devices in a zone are called zone members. A zone contains one or more zone members. A device can belong to one or more zones. End nodes that are members of a zone can communicate with each other, but they are isolated from nodes in other zones of which they are not a member.

If no zone is configured for the device, it resides in the default zone. You can configure the default zone to permit or deny its member devices to communicate with each other. By default, the default zone is set to deny.

You can specify zone members based on any of the following criteria:

- **pWWN:** The port World Wide Number is a unique ID representing a particular end node. The pWWN is a 64-bit hexadecimal value (for example, 20:34:00:80:e5:23:f4:55)
- **FC alias:** The Fibre Channel alias specifies the device that connects to a node port on the switch. The FC alias is a 1-64 character text value (for example, "StorageOne").

Notes:

- When you create an FC alias using SNMP, a default pWWN of value 10:00:00:00:00:00:00:00 is automatically assigned to the FC alias. You must change this default pWWN value. Not doing so will result in a conflict of pWWN IDs the next time you try to create an FC alias.
- The G8296 uses hard zoning, which is enforced in the switch hardware, based on the pWWNs of the Fibre Channel initiators and targets.
- The G8296 supports up to 512 zones per zoneset, each with up to 32 member devices.

Zonesets

Zonesets provide a mechanism for conveniently grouping zones. Each zoneset can contain one or more zones. A zone can belong to one or more zonesets. Only one zoneset can be activated at a given time. If you deactivate the active zoneset, no zonesets are active until you activate another zoneset. If you activate one zoneset while another zoneset is active, the currently active zoneset is deactivated.

When you activate a zoneset, the new zoneset access policies are applied. Any changes to the active zoneset will impact only the traffic between the affected ports.

Affected ports are those ports for which there is a change in the peering relationship due to active zoneset change. RSCNs are generated only for the affected ports.

By default the default-zone status is deny all.

Without zoning configuration, none of the Storage targets is visible to hosts. The default-zone configuration can be made `permit all` using the following command:

```
RS8296(config)# zone default-zone permit vlan <VLAN number>
```

Note: We recommend that you activate well-defined zoning configuration for increased security.

Defining and Activating Zoning

Define the following general properties for Fibre Channel zones and zonesets:

1. If desired, create (or remove) aliases for Fibre Channel devices:

```
RS8296(config)# [no] fcalias name <device alias name> vlan <VLAN number>
```

Repeat for each alias as necessary.

2. If desired, add (or remove) members to the FC aliases:

```
RS8296(config-fcalias)# [no] member pwnn <Port World Wide Name>
```

3. For each desired zone:

- a. Name (or remove) the zone.

```
RS8296(config)# [no] zone name <zone name> vlan <VLAN number>
```

- b. Add (or remove) one or more members to the zone using either pWWNs or FC Aliases:

```
RS8296(config-zone)# [no] member pwnn <Port World Wide Name>
```

or:

```
RS8296(config-zone)# [no] fcalias <device alias name>
```

Repeat as necessary for each member device.

- c. Exit from zone configuration:

```
RS8296(config-zone)# exit
```

- d. Check the zone setup:

```
RS8296(config)# show zone name <zone name> vlan <VLAN number>
```


4. For each desired zoneset:

a. Name (or remove) the zoneset.

```
RS8296(config)# [no] zoneset name <zoneset name> vlan <VLAN number>
```

b. Add (or remove) one or more member zones to the zoneset:

```
RS8296(config-zoneset)# [no] member name <zone name>
```

Repeat as necessary for each member zone.

c. Exit from zoneset configuration:

```
RS8296(config-zoneset)# exit
```

d. Activate the zoneset:

```
RS8296(config)# zoneset activate name <zoneset name> vlan <VLAN number>
```

e. Check the zoneset setup:

```
RS8296(config)# show zoneset name <zoneset name> vlan <VLAN number>
```

5. To check overview of zoning status for a vlan:

```
RS8296(config)# show zone status vlan <VLAN number>
```

Changing the Active Zoneset

Fibre Channel is intended to operate with minimal disruption. To prevent the various synchronization events that would result if each stage of a live zoning configuration was applied, the cumulative configuration changes for zones and zoneset are held in reserve until explicitly activated by an administrator.

When activated, the new zoneset is synchronized throughout the Fibre Channel fabric for each modified zone. Fibre Channel traffic is temporarily disrupted in modified zones as changes to the fabric are recognized by the connected devices. Until activation, the previously established zoneset remains in effect.

The basic zoneset commands are as follows:

- Activate or deactivate a zoneset:

```
RS8296(config)# [no] zoneset activate name <zoneset name> vlan <VLAN number>
```

- View the settings for the active zoneset:

```
RS8296(config)# show zoneset active vlan <VLAN number>
```

- View the settings for the pending configuration changes:

```
RS8296(config)# show zoneset vlan <VLAN number>
```

FCF Scalability

The following scaling limits apply for full fabric FCF:

- Maximum number of VLANs on which FCF can be enabled: 12
- Maximum number of logins per VLAN: 2048
- Maximum number of logins across all 12 VLANs: 2048
- Maximum number of zonesets per VLAN: 8
- Maximum number of zones per VLAN: 512
- Maximum number of members per zone: 32
- Maximum number of members in all zones per VLAN: 1024
- Maximum number of aliases in all zones per VLAN: 1024

Priority-Based Flow Control

Priority-based Flow Control (PFC) is defined in IEEE 802.1Qbb. PFC extends the IEEE 802.3x standard flow control mechanism. Under standard flow control, when a port becomes busy, the switch manages congestion by pausing all the traffic on the port, regardless of the traffic type. PFC provides more granular flow control, allowing the switch to pause specified types of traffic on the port, while other traffic on the port continues.

PFC pauses traffic based on 802.1p priority values in the VLAN tag. The administrator can assign different priority values to different types of traffic and then enable PFC for up to two specific priority values: priority value 3, and one other. The configuration can be applied globally for all ports on the switch. Then, when traffic congestion occurs on a port (caused when ingress traffic exceeds internal buffer thresholds), only traffic with priority values where PFC is enabled is paused. Traffic with priority values where PFC is disabled proceeds without interruption but may be subject to loss if port ingress buffers become full.

Although PFC is useful for a variety of applications, it is required for FCoE implementation where storage (SAN) and networking (LAN) traffic are converged on the same Ethernet links. Typical LAN traffic tolerates Ethernet packet loss that can occur from congestion or other factors, but SAN traffic must be lossless and requires flow control.

For FCoE, standard flow control would pause both SAN and LAN traffic during congestion. While this approach would limit SAN traffic loss, it could degrade the performance of some LAN applications that expect to handle congestion by dropping traffic. PFC resolves these FCoE flow control issues. Different types of SAN and LAN traffic can be assigned different IEEE 802.1p priority values. PFC can then be enabled for priority values that represent SAN and LAN traffic that must be paused during congestion, and disabled for priority values that represent LAN traffic that is more loss-tolerant.

PFC requires CEE to be turned on ([“Turning CEE On or Off” on page 269](#)). When CEE is turned on, PFC is enabled on priority value 3 by default. Optionally, the administrator can also enable PFC on one other priority value, providing lossless handling for another traffic type, such as for a business-critical LAN application.

Note: For any given port, only one flow control method can be implemented at any given time: either PFC or standard IEEE 802.3x flow control.

Global Configuration

PFC requires CEE to be turned on ([“Turning CEE On or Off” on page 269](#)). When CEE is turned on, standard flow control is disabled on all ports, and PFC is enabled on all ports for 802.1p priority value 3. While CEE is turned on, PFC cannot be disabled for priority value 3. This default is chosen because priority value 3 is commonly used to identify FCoE traffic in a CEE environment and must be guaranteed lossless behavior. PFC is disabled for all other priority values by default, but can be enabled for one additional priority value.

Note: When using global PFC configuration in conjunction with the ETS feature (see [“Enhanced Transmission Selection” on page 286](#)), ensure that only pause-tolerant traffic (such as lossless FCoE traffic) is assigned priority values where PFC is enabled. Pausing other types of traffic can have adverse effects on LAN applications that expect uninterrupted traffic flow and tolerate dropping packets during congestion.

PFC Configuration Example

Note: DCBX may be configured to permit sharing or learning PFC configuration with or from external devices. This example assumes that PFC configuration is being performed manually. See [“Data Center Bridging Capability Exchange” on page 292](#) for more information on DCBX. Even if the G8296 learns the PFC configuration from a DCBX peer, the PFC configuration must be performed manually.

This example is consistent with the network shown in [Figure 24 on page 267](#). In this example, the following topology is used.

Table 33. *Port-Based PFC Configuration*

Switch Port	802.1p Priority	Usage	PFC Setting
1	0-2	LAN	Disabled
	3	(not used)	Enabled
	4	Business-critical LAN	Enabled
	others	(not used)	Disabled
3	3	FCoE	Enabled
	others	(not used)	Disabled
4	0-2	LAN	Disabled
	3	(not used)	Enabled
	4	Business-critical LAN	Enabled
	others	(not used)	Disabled

In this example, PFC is to facilitate lossless traffic handling for FCoE (priority value 3) and a business-critical LAN application (priority value 4).

Assuming that CEE is off (the G8296 default), the example topology shown in [Table 33](#) can be configured using the following commands:

1. Turn CEE on.

```
RS8296(config)# cee enable
```

Note: Turning CEE on will automatically change some 802.1p QoS and 802.3x standard flow control settings and menus (see [“Turning CEE On or Off” on page 269](#)).

2. Enable PFC for the FCoE traffic.

Note: PFC is enabled on priority 3 by default. If using the defaults, the manual configuration commands shown in this step are not necessary.

3. Enable PFC for the business-critical LAN application:

```
RS8296(config)# cee port 1 pfc priority 4 enable(Enable on LAN priority)
RS8296(config)# cee port 1 pfc priority 4 description "Critical LAN"
(Optional description)
RS8296(config)# cee port 4 pfc priority 4 enable(Enable on LAN priority)
RS8296(config)# cee port 4 pfc priority 4 description "Critical LAN"
(Optional description)
```

4. Save the configuration.

Enhanced Transmission Selection

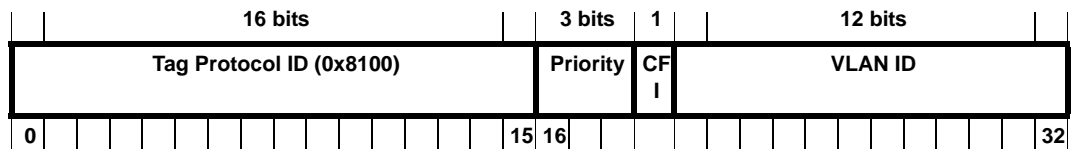
Enhanced Transmission Selection (ETS) is defined in IEEE 802.1Qaz. ETS provides a method for allocating port bandwidth based on 802.1p priority values in the VLAN tag. Using ETS, different amounts of link bandwidth can be specified for different traffic types (such as for LAN, SAN, and management).

ETS is an essential component in a CEE environment that carries different types of traffic, each of which is sensitive to different handling criteria, such as Storage Area Networks (SANs) that are sensitive to packet loss, and LAN applications that may be latency-sensitive. In a single converged link, such as when implementing FCoE, ETS allows SAN and LAN traffic to coexist without imposing contrary handling requirements upon each other.

The ETS feature requires CEE to be turned on (see “Turning CEE On or Off” on page 269).

802.1p Priority Values

Under the 802.1p standard, there are eight available priority values, with values numbered 0 through 7, which can be placed in the priority field of the 802.1Q VLAN tag:



Servers and other network devices may be configured to assign different priority values to packets belonging to different traffic types (such as SAN and LAN).

ETS uses the assigned 802.1p priority values to identify different traffic types. The various priority values are assigned to priority groups (PGID), and each priority group is assigned a portion of available link bandwidth.

Priority values within in any specific ETS priority group are expected to have similar traffic handling requirements with respect to latency and loss.

802.1p priority values may be assigned by the administrator for a variety of purposes. However, when CEE is turned on, the G8296 sets the initial default values for ETS configuration as follows:

Figure 25. Default ETS Priority Groups

Typical Traffic Type	802.1p Priority	PGID	Bandwidth Allocation
LAN	0	0	10%
LAN	1		
LAN	2		
SAN	3	1	50%
Latency-Sensitive LAN	4	2	40%
Latency-Sensitive LAN	5		
Latency-Sensitive LAN	6		
Latency-Sensitive LAN	7		

In the assignment model shown in [Figure 25 on page 286](#), priority values 0 through 2 are assigned for regular Ethernet traffic, which has “best effort” transport characteristics.

Because CEE and ETS features are generally associated with FCoE, Priority 3 is typically used to identify FCoE (SAN) traffic.

Priorities 4-7 are typically used for latency sensitive traffic and other important business applications. For example, priority 4 and 5 are often used for video and voice applications such as IPTV, Video on Demand (VoD), and Voice over IP (VoIP). Priority 6 and 7 are often used for traffic characterized with a “must get there” requirement, with priority 7 used for network control which requires guaranteed delivery to support configuration and maintenance of the network infrastructure.

Note: The default assignment of 802.1p priority values on the G8296 changes depending on whether CEE is on or off. See [“Turning CEE On or Off” on page 269](#) for details.

Priority Groups

For ETS use, each 801.2p priority value is assigned to a priority group which can then be allocated a specific portion of available link bandwidth. To configure a priority group, the following is required:

- CEE must be turned on ([“Turning CEE On or Off” on page 269](#)) for the ETS feature to function.
- A priority group must be assigned a priority group ID (PGID), one or more 802.1p priority values, and allocated link bandwidth greater than 9%.

PGID

Each priority group is identified with number (0 through 7, and 15) known as the PGID.

PGID 0 through 7 may each be assigned a portion of the switch’s available bandwidth.

PGID 8 through 14 are reserved as per the 802.1Qaz ETS standard.

PGID 15 is a strict priority group. It is generally used for critical traffic, such as network management. Any traffic with priority values assigned to PGID 15 is permitted as much bandwidth as required, up to the maximum available on the switch. After serving PGID 15, any remaining link bandwidth is shared among the other groups, divided according to the configured bandwidth allocation settings.

Make sure all 802.1p priority values assigned to a particular PGID have similar traffic handling requirements. For example, PFC-enabled traffic must not be grouped with non-PFC traffic. Also, traffic of the same general type must be assigned to the same PGID. Splitting one type of traffic into multiple 802.1p priorities, and then assigning those priorities to different PGIDs may result in unexpected network behavior.

Each 802.1p priority value may be assigned to only one PGID. However, each PGID may include multiple priority values. Up to eight PGIDs may be configured at any given time.

Assigning Priority Values to a Priority Group

Each priority group may be configured from its corresponding ETS Priority Group, available using the following command:

```
RS8296(config)# cee global ets priority-group pgid <group number (0-7, or 15)>  
priority <priority list>
```

where *priority list* is one or more 802.1p priority values (with each separated by a comma). For example, to assign priority values 0 through 2:

```
RS8296(config)# cee global ets priority-group pgid <group number (0-7, or 15)>  
priority 0,1,2
```

Note: Within any specific PGID, the PFC settings (see [“Priority-Based Flow Control” on page 283](#)) must be the same (enabled or disabled) for all priority values within the group. PFC can be enabled only on priority value 3 and one other priority.

When assigning priority values to a PGID, the specified priority value will be automatically removed from its old group and assigned to the new group when the configuration is applied.

Each priority value must be assigned to a PGID. Priority values may not be deleted or unassigned. To remove a priority value from a PGID, it must be moved to another PGID.

For PGIDs 0 through 7, bandwidth allocation can also be configured through the ETS Priority Group menu. See for [“Allocating Bandwidth” on page 288](#) for details.

Deleting a Priority Group

A priority group is automatically deleted when it contains no associated priority values, and its bandwidth allocation is set to 0%.

Note: The total bandwidth allocated to PGID 0 through 7 must equal exactly 100%. Reducing the bandwidth allocation of any group will require increasing the allocation to one or more of the other groups (see [“Allocating Bandwidth” on page 288](#)).

Allocating Bandwidth

Allocated Bandwidth for PGID 0 Through 7

The administrator may allocate a portion of the switch’s available bandwidth to PGIDs 0 through 7. Available bandwidth is defined as the amount of link bandwidth that remains after priorities within PGID 15 are serviced (see [“Unlimited Bandwidth for PGID 15” on page 289](#)), and assuming that all PGIDs are fully subscribed. If any PGID does not fully consume its allocated bandwidth, the unused portion is made available to the other priority groups.

Priority group bandwidth allocation can be configured using the following command:

```
RS8296(config)# cee global ets priority-group pgid <priority group number>  
bandwidth <bandwidth allocation> pgid <priority group number> bandwidth  
<bandwidth allocation>
```


where *bandwidth allocation* represents the percentage of link bandwidth, specified as a number between 10 and 100, in 1% increments, or 0.

The following bandwidth allocation rules apply:

- Bandwidth allocation must be 0% for any PGID that has no assigned 802.1p priority values.
- Any PGID assigned one or more priority values must have a bandwidth allocation greater than 9%.
- Total bandwidth allocation for groups 0 through 7 must equal exactly 100%. Increasing or reducing the bandwidth allocation of any PGID also requires adjusting the allocation of other PGIDs to compensate.

If these conditions are not met, the switch will report an error when applying the configuration.

Note: Actual bandwidth used by any specific PGID may vary from configured values by up to 10% of the available bandwidth in accordance with 802.1Qaz ETS standard. For example, a setting of 10% may be served anywhere from 0% to 20% of the available bandwidth at any given time.

Unlimited Bandwidth for PGID 15

PGID 15 is permitted unlimited bandwidth and is generally intended for critical traffic (such as switch management). Traffic in this group is given highest priority and is served before the traffic in any other priority group.

If PGID 15 has low traffic levels, most of the switch's bandwidth will be available to serve priority groups 0 through 7. However, if PGID 15 consumes a larger part of the switch's total bandwidth, the amount available to the other groups is reduced.

Note: Consider traffic load when assigning priority values to PGID 15. Heavy traffic in this group may restrict the bandwidth available to other groups.

Configuring ETS

Consider an example consistent with that used for port-based PFC configuration (on [page 284](#)):

Table 34. *ETS Configuration*

Priority	Usage	PGID	Bandwidth
0	LAN (best effort delivery)	0	10%
1	LAN (best effort delivery)		
2	LAN (best effort delivery)		
3	SAN (Fibre Channel over Ethernet, with PFC)	1	20%
4	Business Critical LAN (lossless Ethernet, with PFC)	2	30%
5	Latency-sensitive LAN	3	40%
6	Latency-sensitive LAN		
7	Network Management (strict)	15	unlimited

The example shown in [Table 34](#) is only slightly different than the default configuration shown in [Figure 25 on page 286](#). In this example, latency-sensitive LAN traffic (802.1p priority 5 through 6) are moved from priority group 2 to priority group 3. This leaves Business Critical LAN traffic (802.1p priority 4) in priority group 2 by itself. Also, a new group for network management traffic has been assigned. Finally, the bandwidth allocation for priority groups 1, 2, and 3 are revised.

Note: DCBX may be configured to permit sharing or learning PFC configuration with or from external devices. This example assumes that PFC configuration is being performed manually. See [“Data Center Bridging Capability Exchange” on page 292](#) for more information on DCBX.

This example can be configured using the following commands:

1. Turn CEE on.

```
RS8296(config)# cee enable
```

Note: Turning CEE on will automatically change some 802.1p QoS and 802.3x standard flow control settings and menus (see [“Turning CEE On or Off” on page 269](#)).

2. Configure each allocated priority group with a description (optional), list of 802.1p priority values, and bandwidth allocation:

```
RS8296(config)# cee global ets priority-group pgid 0 priority 0,1,2
                                     (Select a group for regular LAN, and
                                     set for 802.1p priorities 0, 1, and 2)
RS8296(config)# cee global ets priority-group pgid 0 description
"Regular LAN"
                                     (Set a group description—optional)
RS8296(config)# cee global ets priority-group pgid 1 priority 3
                                     (Select a group for SAN traffic, and
                                     set for 802.1p priority 3)
RS8296(config)# cee global ets priority-group pgid 1 description "SAN"
                                     (Set a group description—optional)
RS8296(config)# cee global ets priority-group pgid 2 priority 4
                                     (Select a group for latency traffic,
                                     and set for 802.1p priority 4)
RS8296(config)# cee global ets priority-group pgid 2 description
"Biz-Critical LAN"
                                     (Set a group description—optional)
RS8296(config)# cee global ets priority-group pgid 3 description
"Latency-Sensitive LAN"
                                     (Set a group description—optional)
RS8296(config)# cee global ets priority-group pgid 3 priority 5,6 pgid 0
bandwidth 10 pgid 1 bandwidth 20 pgid 2 bandwidth 30 pgid 3 bandwidth
40
                                     (Configure link bandwidth restriction))
```

3. Configure the strict priority group with a description (optional) and a list of 802.1p priority values:

```
RS8296(config)# cee global ets priority-group pgid 15 priority 7
                                     (Select a group for strict traffic, and
                                     Set 802.1p priority 7)
RS8296(config)# cee global ets priority-group pgid 15 description
"Network Management"
                                     (Set a group description—optional)
```

Note: Priority group 15 is permitted unlimited bandwidth. As such, the commands for priority group 15 do not include bandwidth allocation.

4. Save the configuration.

To view the configuration, use the following command:

```
RS8296(config)# show cee global ets

Current ETS Configuration:
Number of COSq: 8

Current Mapping of 802.1p Priority to Priority Groups:

Priority  PGID  COSq
-----  ----  ----
    0      0      0
    1      0      0
    2      0      0
    3      1      1
    4      2      2
    5      3      3
    6      3      3
    7     15      7

Current Bandwidth Allocation to Priority Groups:

PGID  PG%  Description
-----  ---  -----
    0   10  Regular LAN
    1   20  SAN
    2   30  Biz-Critical LAN
    3   40  Latency-sensitive LAN
   15    -  Network Management
```

Data Center Bridging Capability Exchange

Data Center Bridging Capability Exchange (DCBX) protocol is a vital element of CEE. DCBX allows peer CEE devices to exchange information about their advanced capabilities. Using DCBX, neighboring network devices discover their peers, negotiate peer configurations, and detect misconfigurations.

DCBX provides two main functions on the G8296:

- Peer information exchange

The switch uses DCBX to exchange information with connected CEE devices. For normal operation of any FCoE implementation on the G8296, DCBX must remain enabled on all ports participating in FCoE.

- Peer configuration negotiation

DCBX also allows CEE devices to negotiate with each other for the purpose of automatically configuring advanced CEE features such as PFC, ETS, and (for some CNAs) FIP. The administrator can determine which CEE feature settings on the switch are communicated to and matched by CEE neighbors, and also which CEE feature settings on the switch may be configured by neighbor requirements.

The DCBX feature requires CEE to be turned on (see [“Turning CEE On or Off” on page 269](#)).

DCBX Settings

When CEE is turned on, DCBX is enabled for peer information exchange on all ports. For configuration negotiation, the following default settings are configured:

- Application Protocol: FCoE and FIP snooping is set for traffic with 802.1p priority 3
- PFC: Enabled on 802.1p priority 3
- ETS
 - Priority group includes priority values 0 through 2, with bandwidth allocation of 10%
 - Priority group includes priority value 3, with bandwidth allocation of 50%
 - Priority group includes priority values 4 through 7, with bandwidth allocation of 40%

Enabling and Disabling DCBX

When CEE is turned on, DCBX can be enabled and disabled on a per-port basis, using the following commands:

```
RS8296(config)# [no] cee port <port alias or number> dcbx enable
```

When DCBX is enabled on a port, Link Layer Detection Protocol (LLDP) is used to exchange DCBX parameters between CEE peers. Also, the interval for LLDP transmission time is set to one second for the first five initial LLDP transmissions,

after which it is returned to the administratively configured value. The minimum delay between consecutive LLDP frames is also set to one second as a DCBX default.

Peer Configuration Negotiation

CEE peer configuration negotiation can be set on a per-port basis for a number of CEE features. For each supported feature, the administrator can configure two independent flags:

- The `advertise` flag

When this flag is set for a particular feature, the switch settings will be transmitted to the remote CEE peer. If the peer is capable of the feature, and willing to accept the G8296 settings, it will be automatically reconfigured to match the switch.

- The `willing` flag

Set this flag when required by the remote CEE peer for a particular feature as part of DCBX signaling and support. Although some devices may also expect this flag to indicate that the switch will accept overrides on feature settings, the G8296 retains its configured settings. As a result, the administrator must configure the feature settings on the switch to match those expected by the remote CEE peer.

These flags are available for the following CEE features:

- Application Protocol

DCBX exchanges information regarding FCoE and FIP snooping, including the 802.1p priority value used for FCoE traffic. The `advertise` flag is set or reset using the following command:

```
RS8296(config)# [no] cee port <port alias or number> dcbx app_proto advertise
```

The `willing` flag is set or reset using the following command:

```
RS8296(config)# [no] cee port <port alias or number> dcbx app_proto willing
```

- PFC

DCBX exchanges information regarding whether PFC is enabled or disabled on the port. The `advertise` flag is set or reset using the following command:

```
RS8296(config)# [no] cee port <port alias or number> dcbx pfc advertise
```

The `willing` flag is set or reset using the following command:

```
RS8296(config)# [no] cee port <port alias or number> dcbx pfc willing
```

- ETS

DCBX exchanges information regarding ETS priority groups, including their 802.1p priority members and bandwidth allocation percentages. The `advertise` flag is set or reset using the following command:

```
RS8296(config)# [no] cee port <port alias or number> dcbx ets advertise
```

The `willing` flag is set or reset using the following command:

```
RS8296(config)# [no] cee port <port alias or number> dcbx ets willing
```

Configuring DCBX

Consider an example consistent [Figure 24 on page 267](#) and used with the previous FCoE examples in this chapter:

- FCoE is used on ports 2 and 3.
- CEE features are also used with LANs on ports 1 and 4.
- All other ports are disabled or are connected to regular (non-CEE) LAN devices.

In this example, the G8296 acts as the central point for CEE configuration. FCoE-related ports will be configured for advertising CEE capabilities, but not to accept external configuration. Other LAN ports that use CEE features will also be configured to advertise feature settings to remote peers, but not to accept external configuration. DCBX will be disabled on all non-CEE ports.

This example can be configured using the following commands:

1. Turn CEE on.

```
RS8296(config)# cee enable
```

Note: Turning CEE on will automatically change some 802.1p QoS and 802.3x standard flow control settings and menus (see [“Turning CEE On or Off” on page 269](#)).

2. Enable desired DCBX configuration negotiation on FCoE ports:

```
RS8296(config)# cee port 2 dcbx enable
RS8296(config)# cee port 2 dcbx app_proto advertise
RS8296(config)# cee port 2 dcbx ets advertise
RS8296(config)# cee port 2 dcbx pfc advertise

RS8296(config)# cee port 3 dcbx enable
RS8296(config)# cee port 3 dcbx app_proto advertise
RS8296(config)# cee port 3 dcbx ets advertise
RS8296(config)# cee port 3 dcbx pfc advertise
```

3. Enable desired DCBX advertisements on other CEE ports:

```
RS8296(config)# cee port 1 dcbx enable
RS8296(config)# cee port 1 dcbx app_proto advertise
RS8296(config)# cee port 1 dcbx ets advertise
RS8296(config)# cee port 1 dcbx pfc advertise

RS8296(config)# cee port 4 dcbx enable
RS8296(config)# cee port 4 dcbx app_proto advertise
RS8296(config)# cee port 4 dcbx ets advertise
RS8296(config)# cee port 4 dcbx pfc advertise
```

4. Disable DCBX for each non-CEE port as appropriate:

```
RS8296(config)# no cee port 5-64 dcbx enable
```

5. Save the configuration.

Port Trunking

Networking OS 8.2 supports port trunking for FCoE connections. You can use Link Aggregation (LAG) for separate FCoE traffic or for Ethernet and FCoE traffic. FCoE and Ethernet traffic can co-exist within the same trunk (ports). By default, FCoE servers (CNA/HBA) do not support trunking, while Ethernet (NIC/CNA) are trunk-capable.

Uplink ports, connected to the FCF, can be grouped as static or dynamic trunks. Internal ports cannot be grouped as trunks.

You can perform normal trunk operations such as creating or enabling the trunk and adding or removing member ports. When a port is added to a trunk group, FCFs previously detected on the port will be deleted. The deleted FCF may be relearned later. However, this may cause flickering in the network traffic.

Priority-based Flow Control (PFC), and Data Center Bridging Exchange (DCBX) are configured on a per-port basis. Each port in a trunk must have the same PFC and DCBX configuration. When a port ceases to be a trunk group member, the port's configuration does not change.

Chapter 20. Edge Virtual Bridging

The 802.1Qbg/Edge Virtual Bridging (EVB) is an emerging IEEE standard for allowing networks to become virtual machine (VM)-aware. EVB bridges the gap between physical and virtual network resources. The IEEE 802.1Qbg simplifies network management by providing a standards-based protocol that defines how virtual Ethernet bridges exchange configuration information. In EVB environments, physical end stations, containing multiple virtual end stations, use a bridge to form a LAN. The virtual NIC (vNIC) configuration information of a virtual machine is available to these EVB devices. This information is generally not available to an 802.1Q bridge.

Lenovo Network Operating System EVB features are compliant with the IEEE 802.1Qbg Authors Group Draft 0.2. For a list of documents on this feature, see: <http://www.ieee802.org/1/pages/802.1bg.html>.

The RackSwitch G8296 performs the role of a 802.1Qbg bridge in an EVB environment.

Networking OS implementation of EVB supports the following protocols:

- Virtual Ethernet Bridging (VEB) and Virtual Ethernet Port Aggregator (VEPA): VEB and VEPA are mechanisms for switching between VMs on the same hypervisor. VEB enables switching with the server, either in the software (vSwitch), or in the hardware (using single root I/O virtualization capable NICs). VEPA requires the edge switch to support “Reflective Relay” — an operation where the switch forwards a frame back to the port on which it arrived if the destination MAC address is on the same port.
- Edge Control Protocol (ECP): ECP is a transport protocol that operates between two peers over an IEEE 802 LAN. ECP provides reliable, in-order delivery of ULP (Upper Layer Protocol) PDUs (Protocol Data Units).
- Virtual Station Interface (VSI) Discovery and Configuration Protocol (VDP): VDP allows hypervisors to advertise VSIs to the physical network. This protocol also allows centralized configuration of network policies that will persist with the VM, independent of its location.
- EVB Type-Length-Value (TLV): EVB TLV is a Link Layer Discovery protocol (LLDP)-based TLV used to discover and configure VEPA, ECP, and VDP.

EVB Operations Overview

The N/OS includes a pre-standards VSI Type Database (VSIDB) implemented through the System Networking Switch Center (SNSC), the Flex System Manager (FSM), or the IBM System Networking Distributed Switch 5000V. The VSIDB is the central repository for defining sets of network policies that apply to VM network ports. You can configure only one VSIDB.

Note: This document does not include the VSIDB configuration details. Please see the SNSC, FSM, or IBM System Networking Distributed Switch 5000V guide for details on how to configure VSIDB.

The VSIDB operates in the following sequence:

1. Define VSI types in the VSIDB. The VSIDB exports the database when the G8296 sends a request.
2. Create a VM. Specify VSI type for each VM interface. See the SNSC, FSM, or IBM System Networking Distributed Switch 5000V guide for details on how to specify the VSI type.

The hypervisor sends a VSI ASSOCIATE, which contains the VSI type ID, to the switch port after the VM is started. The switch updates its configuration based on the requested VSI type. The switch configures the per-VM bandwidth using the VMpolicy.

The N/OS supports the following policies for VMs:

- ACLs
- Bandwidth metering

VSIDB Synchronization

The switch periodically checks for VSIDB changes based on the configured interval. You can configure this interval using the following command:

To disable periodic updates, configure the interval value as 0.

If the switch finds that the VSIDB has changed, it updates the local VSIDB cache. When the cache is successfully updated, it sends a syslog message.

After updating the local VSIDB cache, the switch disassociates any VM whose type ID or VLAN no longer exists in the updated cache.

The switch updates the local VSIDB cache when any of the following takes place:

- When, at the configured refresh interval, the switch finds that the VSIDB configuration has changed since the last poll.
- When a VM sends an ASSOCIATE message, but the VSI type does not exist in the local VSIDB cache.
- When a VM sends an ASSOCIATE message, and the VSI type exists but the VSI type's VLAN ID does not exist in the local VSIDB cache.
- When you update the VSIDB using the following command:
- When the management port link status changes from down to up

VLAN Behavior

When a VM gets associated, the corresponding VLAN is dynamically created on the switch port if the VLAN does not already exist.

VLANs that are dynamically created will be automatically removed from the switch port when there are no VMs using that VLAN on the port.

Dynamic VLAN information will not be displayed in the running configuration. However, the VLAN, port, and STP commands display the dynamic VLAN information with a "*".

If you configure any Layer 2/Layer 3 features on dynamically created VLANs, the VLAN information is displayed in the running configuration.

Deleting a VLAN

If you delete a VLAN that has a VM associated with it, you will see a warning message similar to the following:

```
Warning: Vlan 10 is used by VM and can't be removed.
```

The VMs will not get disassociated.

If a VM is associated with a port, and you remove this port from a VLAN, you will see a warning message similar to the following:

```
Warning: Port 23 in Vlan 10 is used by VM and can't be removed.
```

The VMs will not get disassociated.

EVB Configuration

This section includes the steps to configure EVB based on the following values:

- Profile number: 1
- Port number:
- Retry interval: 8000 milliseconds
- VSI Database:
 - Manager IP: 172.31.37.187
 - Port: 80

1. Create an EVB profile.

```
RS8296(config)# virt evb profile 1 (Enter number from 1-16)
```

2. Enable Reflective Relay.

```
RS8296(conf-evbprof)# reflective-relay
```

3. Enable VSI discovery.

```
RS8296(conf-evbprof)# vsi-discovery  
RS8296(conf-evbprof)# exit
```

4. Add EVB profile to port.

```
RS8296(config)# interface port 1  
RS8296(config-if)# evb profile 1 (Enter EVB profile ID)  
RS8296(config-if)# exit
```

5. Configure ECP retransmission interval.

```
RS8296(config)# ecp retransmit-interval 8000  
(Enter retransmission interval in milliseconds (100-9000))
```

6. Set VSI database information.

```
RS8296(config)# virt evb vsidb 1  
RS8296(conf-vsldb)# host 172.31.37.187 (Set VSI database Manager IP)  
RS8296(conf-vsldb)# port 80 (Set VSI database Manager port)  
RS8296(conf-vsldb)# filepath "vsldb" (Set VSI database document path)  
RS8296(conf-vsldb)# filename "all.xml" (Set VSI database file name)  
RS8296(conf-vsldb)# update-interval 30 (Set update interval in seconds)  
RS8296(conf-vsldb)# exit
```

Note: When you connect to a SNSC VSIDB, the port/docpath configuration is as follows:

HTTP:

- Port: 40080

- Docpath: snsc/rest/vsitypes

HTTPS:

- Port: 40443
- Docpath: snsc/rest/vsitypes

When you connect to a 5000v VSIDB, the port/docpath configuration is as follows:

- Port: 80
- Docpath: vsitypes

7. Enable LLDP.

```
RS8296(config)# lldp enable
```

(Turn on LLDP)

Limitations

- If both ACL and egress bandwidth metering are enabled, traffic will first be matched with the ACL and will not be limited by bandwidth metering.
- ACLs based on a source MAC or VLAN must match the source MAC and VLAN of the VM. If not, the policy will be ignored and you will see the following warning message:

```
"vm: VSI Type ID 100 Associated mac 00:50:56:b6:c0:ff on port 6,  
ignore 1 mismatched ACL"
```

Unsupported features

The following features are not supported with EVB:

- LAG/VLAG
- vNIC

Chapter 21. Static Multicast ARP

The Microsoft Windows operating system includes the Network Load Balancing (NLB) technology that helps to balance incoming IP traffic among multi-node clusters. In multicast mode, NLB uses a shared multicast MAC address with a unicast IP address. Since the address resolution protocol (ARP) can map an IP address to only one MAC address, port, and VLAN, the packet reaches only one of the servers (the one attached to the port on which the ARP was learnt).

To avoid the ARP resolution, you must create a static ARP entry with multicast MAC address. You must also specify the list of ports through which the multicast packet must be sent out from the gateway or Layer 2/Layer 3 node.

With these configurations, a packet with a unicast IPv4 destination address and multicast MAC address can be sent out as per the multicast MAC address configuration. NLB maps the unicast IP address and multicast MAC address as follows:

Cluster multicast MAC address: 03-BF-W-X-Y-Z; where W.X.Y.Z is the cluster unicast IP address.

You must configure the static multicast ARP entry only at the Layer 2/Layer 3 or Router node, and not at the Layer 2-only node.

Lenovo Network Operating System supports a maximum of 20 static multicast ARP entries.

Note: If you use the ACL profile, an ACL entry is consumed for each Static Multicast ARP entry that you configure. Hence, you can configure a maximum of 896 ACLs and multicast MAC entries together when using the ACL profile. The ACL entries have a higher priority. In the default profile, the number of static multicast ARP entries that you configure does not affect the total number of ACL entries.

Configuring Static Multicast ARP

To configure multicast MAC ARP, you must perform the following steps:

- Configure the static multicast forwarding database (FDB) entry: Since there is no port list specified for static multicast ARP, and the associated MAC address is multicast, you must specify a static multicast FDB entry for the cluster MAC address to limit the multicast domain. If there is no static multicast FDB entry defined for the cluster MAC address, traffic will not be forwarded. Use the following command:

```
RS8296(config)# mac-address-table multicast <cluster MAC address> <port(s)>
```

- Configure the static multicast ARP entry: Multicast ARP static entries should be configured without specifying the list of ports to be used. Use the following command:

```
RS8296(config)# ip arp <destination unicast IP address> <destination multicast MAC address> vlan <cluster VLAN number>
```

Configuration Example

Consider the following example:

- Cluster unicast IP address: 10.10.10.42
- Cluster multicast MAC address: 03:bf:0A:0A:0A:2A
- Cluster VLAN: 42
- List of individual or port trunks to which traffic should be forwarded: 54 and 56

Following are the steps to configure the static multicast ARP based on the given example:

1. Configure the static multicast FDB entry.

```
RS8296(config)# mac-address-table multicast 03:bf:0A:0A:0A:2A 42 54,56
```

2. Configure the static multicast ARP entry:

```
RS8296(config)# ip arp 10.10.10.42 03:bf:0A:0A:0A:2A vlan 42
```

You can verify the configuration using the following commands:

- Verify static multicast FDB entry:

```
RS8296(config)# show mac-address-table multicast address
03:bf:0A:0A:0A:2A

Multicast Address  VLAN  Port(s)
-----
03:bf:0A:0A:0A:2A  42    54 56
```


- Verify static multicast ARP entry:

```

RS8296(config)# show ip arp

Current ARP configuration:
  rearp 5
Current static ARP:
  ip          mac          port  vlan
  -----
  10.10.10.42  03:bf:0A:0A:0A:2A      42
  -----
Total number of arp entries : 2
  IP address  Flags  MAC address  VLAN  Age Port
  -----
  10.10.10.1   P    fc:cf:62:9d:74:00  42
  10.10.10.42  P    03:bf:0A:0A:0A:2A  42    0

```

Limitations

- You must configure the ARP only in the Layer 2/Layer 3 node or the router node but not in the Layer 2-only node. Networking OS cannot validate if the node is Layer 2-only.
- The packet is always forwarded to all the ports as specified in the Multicast MAC address configuration. If VLAN membership changes for the ports, you must update this static multicast MAC entry. If not, the ports, whose membership has changed, will report discards.
- ACLs take precedence over static multicast ARP. If an ACL is configured to match and permit ingress of unicast traffic, the traffic will be forwarded based on the ACL rule, and the static multicast ARP will be ignored.

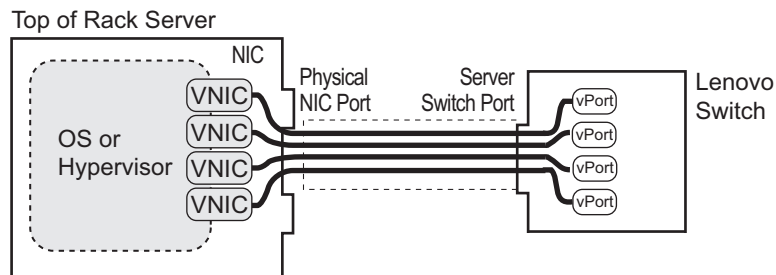
Chapter 22. Unified Fabric Port

Unified Fabric Port (UFP) is a cost-effective way to allocate, share and dynamically control network bandwidth between a server and a switch. UFP lets you create multiple virtual connections. The UFP protocol is a link-level protocol that runs a separate instance for each physical communication link established between a server NIC and a switch port. Virtualizing the ports allows you to separate or aggregate port traffic by applying the network policies defined on the switch. Virtualization lessens bottlenecks and provides higher bandwidth while consolidating equipment use.

UFP provides a switch fabric component to control NICs. The server operating system (OS) or hypervisor recognizes each subdivided link (channel) as an independent physical NIC. Each channel has a unique identity and profile that defines its properties and functionality. The server communicates with the switch over the channel as defined in the channel profile. The channels share the high-speed physical link bandwidth.

For each channel, the vNIC on the server side communicates with virtual port on the switch side.

Figure 26. UFP vPorts



The UFP protocol has the following operation categories:

- **Channel Initialization:** The server NIC and the switch port negotiate the number of channels and establish channel identifiers. Each UFP channel has a data component and a control component. The two components have the same UFP channel ID.
- **Channel Control:** For an established channel, the switch can modify configurable channel properties by sending a control message on the UFP channel. While the channel ID is the same for the control and data components, the destination MAC address of the control message frame is the well-known address "01-80-C2-00-00-03".
- **Discovery Capability:** UFP can discover other ports that are UFP enabled. Once you enable UFP, you can check the information statistics for established channels.

UFP Limitations

The following limitations apply when configuring UFP:

- FCoE must be configured only on vPort 2 of the physical NIC.
- UFP port in FCoE mode cannot operate with FIP auto-VLAN feature.
- VLANs that have member vPorts configured in trunk-, access-, or auto-modes cannot have member vPorts configured in tunnel mode or FCoE.
- vPorts on a physical port must be members of separate VLANs.
- VLANs 4002-4005 are reserved for outer tagging.
- A UFP-enabled port with no vPorts configured cannot belong to the same VLAN as a UFP-enabled port that has vPorts configured in trunk, access, or auto modes.
- UFP bandwidth is guaranteed lossless only for unicast traffic.
- VMready is supported only on a vPort which is configured in auto-VLAN mode. When a vPort is in auto-VLAN mode, it can support up to 32 VMGroups.
- EVB is supported only on a vPort which is configured in auto-VLAN mode.
- VMready and EVB cannot be configured on the same physical port.
- UFP vPorts can support up to 1024 VLANs.
- When CEE is turned on, FCoE vPort must be used for lossless priority traffic. For loss-tolerant priority traffic, a non-FCoE UFP vPort must be used. The lossless property of FCoE vPort is not guaranteed, if lossless and loss-tolerant traffic are combined.
- When the vPort is enabled and the channel link state is up, the system does not support changing vPort VLAN type from private/non-private to non-private/private.
- A maximum of four vPorts can be configured for each physical switch port.
- VMReady Local Group configuration is not supported by UFP.

Virtual Ports Modes

A single physical switch port is configured with virtual ports (vPorts). Each UFP channel connects the server vNIC with a switch vPort. Properties that are defined for a vPort, such as native VLAN and bandwidth, are applied to the traffic that belongs to the vPort.

Note: A maximum of four vPorts can be configured for each physical switch port.

vPort-S-Tag Mapping

A vPort can also be identified with an S-tag (service tag or outer tag). When a vPort is initialized, the switch communicates the UFP channel ID of the vPort to the server vNIC. When the server NIC or switch transmit frames, they add this S-tag to indicate the vPort or vNIC to which the packet is being transmitted. No VLAN mapping is required. Such packets can be single tagged or double tagged (with S-tag).

vPort-VLAN Mapping

In local domain data path type, the switch and server identify the vPort and vNIC by the port and VLAN tag in the incoming and outgoing packets. Because no two vPorts carry traffic for the same VLAN, the port-and-VLAN combination must be uniquely mapped to a vPort.

UFP vPort Mode

The UFP mode is configured based on the type of switching domain (single VLAN or multiple VLANs) where the vPort is connected.

- Use local domain data path types for trunk or access mode.
- Use pass-through domain data path types for tunnel mode. In tunnel mode, a vPort can belong to only one VLAN.

Use the following command to configure UFP vPort mode:

```
RS8296(config)# ufp port <num> vport <num>
RS8296(config_ufp_vport)# network mode {access|trunk|auto|tunnel|fcoe}

Default mode is 'tunnel'
```

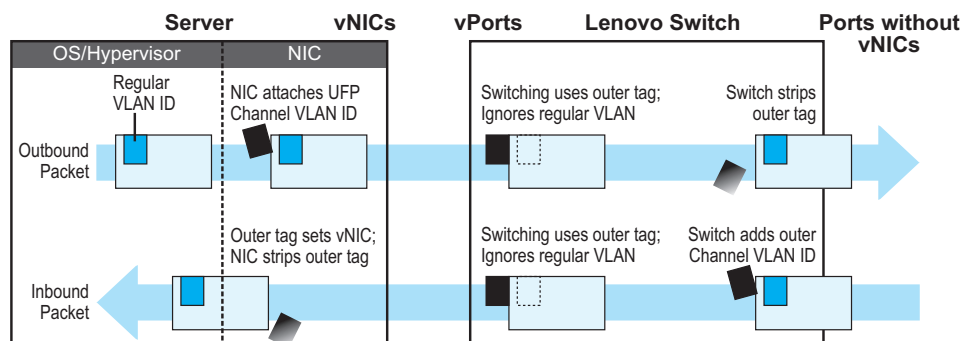
Tunnel Mode

In tunnel mode, a vPort can belong to only one VLAN. An outer tag with the vPort's VLAN ID is inserted in packets that egress the vPort. The inner VLAN tag remains unchanged. The switch processes packets based on the outer tag. When all the ports or vPorts that belong to a particular VLAN are placed in tunnel mode, they belong to one pass-through domain.

Use tunnel mode to send all VM data traffic to an upstream switch, for Layer 2 or Layer 3 processing, in one domain. In such cases, the UFP port or vPort must be in tunnel mode and the upstream switch port must be in 802.1Q trunk mode.

Note: Two vPorts on a physical port cannot be members of the same VLAN.

Figure 27. Packet pass-through in Tunnel Mode

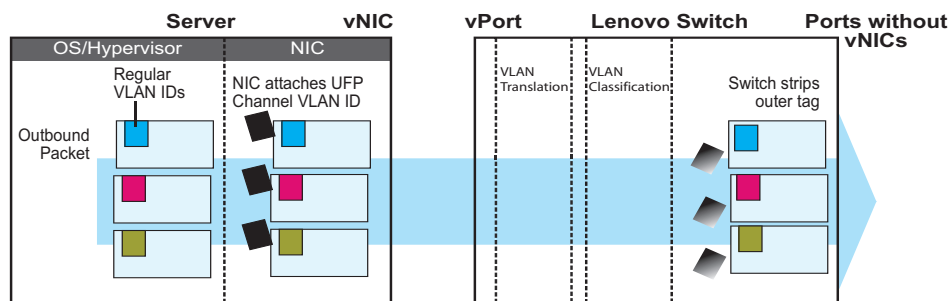


802.1Q Trunk Mode

In trunk mode, a vPort can carry packets that have inner tags that belong to up to 1024 VLANs. A maximum of 4089 VLANs for all UFP vPorts can be configured on the switch in stand-alone mode. Each VLAN in the inner tag requires a VLAN translation entry.

Note: Two vPorts operating in trunk mode on the same physical port cannot carry the same set of VLANs in the inner tag.

Figure 28. Packet passing through in Trunk Mode



Access Mode

In access mode, a vPort carries packets with inner tags that belong to one VLAN. The vPort is associated with the VLAN defined by the command:

```
RS8296(config_ufp_vport)# network default-vlan <2-4094>
```

Note: VLANs 4002-4005 are reserved for outer tagging.

FCoE Mode

FCoE traffic is carried by a vPort. The server-side endpoint of this virtual port will be represented through a FC vHBA. Setting a virtual port in FCoE mode will enable Priority-based Flow Control (PFC) on the physical port.

A vPort configured in FCoE mode can only be attached to a Fibre Channel (FC) VLAN. Only vPort 2 on a physical port can be configured in FCoE mode. A vPort in FCoE mode operates as a local domain data path type with packets being single tagged.

Auto-VLAN Mode

When a vPort is configured in auto-VLAN mode, the vPort participates in VM discovery using VMready or 802.1Qbg. VLANs are dynamically provisioned based on VMready discovery or 802.1Qbg VM association.

When a vPort operates in auto-VLAN mode, it supports 32 VM groups. In the case of 802.1Qbg, when a vPort operates in auto-VLAN mode, the maximum number of VLANs in the inner tag are 1024. The vPort cannot be configured in Virtual Ethernet Port Aggregator (VEPA) mode.

UFP Bandwidth Provisioning

UFP provides one mode of bandwidth provisioning for vPort: Strict Bandwidth Provisioning Mode.

UFP Strict Bandwidth Provisioning Mode

Strict bandwidth provisioning mode configures the switch and NIC apply bidirectional bandwidth control on the vPort as per the defined configuration. By default, a bandwidth of 2.5 Gbps per vPort is guaranteed. If other vPorts are idle, the bandwidth of a vPort can be up to 10 Gbps. A minimum bandwidth of 1 Gbps is provisioned, which can be raised by 100 Mbps increments. The sum of the minimum bandwidth guaranteed for all vPorts together cannot exceed the capacity of the physical link. A vPort can also be configured with a maximum bandwidth.

This mode works with the port scheduler to avoid unintended packet drops due to policing through EFP metering block. If flow control is enabled, the switch provides a no-drop packet forwarding behavior, which improves end-to-end TCP-throughput performance.

Note: If a vPort is configured with low upper limit, it might lead to head-of-line congestion on the egress port.

By default, uplink ports have a separate traffic class for storage traffic with guaranteed bandwidth. The rest of the bandwidth is shared equally among other traffic.

Use the following command to configure strict bandwidth provisioning:

```
RS8296(config_ufp_vport)# qos bandwidth {max|min} <10-100>
```

```
min - Set minimum guaranteed bandwidth
```

```
max - Set maximum allowed bandwidth
```

Using UFP with Other RackSwitch G8296 Features

UFP works with other G8296 features, as described with limitations and details.

Layer 2 Failover

UFP failover can be configured with manual monitoring.

Layer 2 failover is not supported on UFP ports in auto mode.

For more information on failover, see [“Layer 2 Failover” on page 493](#).

For an example configuration, see [“Example 6: Layer 2 Failover Configuration” on page 320](#).

Increased VLAN Limits

Configured with UFP and VLANs, a vPort can support a maximum of 1024 VLANs. A UFP port supports 1024 VLANs.

For more information on VLAN configuration, see [“VLANs” on page 119](#).

VMReady

Configuring with UFP and VMReady, the G8296 can support up to 32 VMGroups with UFP vPorts in auto-mode.

VMReady is supported only on a vPort which is configured in auto-VLAN mode.

For more information on VMReady, see [“VMready” on page 241](#).

802.1Qbg

Configured with Edge Virtual Bridging (EVB), UFP supports up to 1024 VLANs on a vPort.

EVB is supported only on a vPort which is configured in auto-VLAN mode.

For more information on EVB, see [“Edge Virtual Bridging” on page 297](#).

UFP Configuration Examples

Following is an example configuration of UFP vPorts in access mode.

Example 1: Access Mode

1. Configure port 1 as a UFP server port.

```
G8296(config)# system server-ports port 1
```

2. Turn on UFP.

```
G8296(config)# ufp enable
```

3. Configure port 1 as UFP.

```
G8296(config)# ufp port 1 enable
```

```
Warning: "Tagging/Trunk-mode" is enabled on UFP port 1
```

4. Configure virtual port.

```
G8296(config)# ufp port 1 vport 1
```

5. Configure vPort access mode.

```
G8296(config_ufp_vport)# network mode access
```

6. Configure vPort default VLAN.

```
G8296(config_ufp_vport)# network default-vlan 100
```

7. Ensure tagging is disabled on vPort.

```
G8296(config_ufp_vport)# no network default-tag
```

8. Specify QoS parameters for the vPort.

```
G8296(config_ufp_vport)# qos bandwidth min 25 (in percentage)  
G8296(config_ufp_vport)# qos bandwidth max 100 (in percentage)
```

9. Enable the vPort.

```
G8296(config_ufp_vport)# enable  
G8296(config_ufp_vport)# exit
```

10. Configure PVID/Native VLAN for port 2.

Note: Port 2 must be an upstream port.

```
G8296(config)# interface port 2
G8296(config-if)# switchport mode access
G8296(config-if)# switchport access vlan 100
```

Example 2: Trunk Mode

Following is an example configuration of UFP vPorts in trunk mode.

1. Configure ports 1 and 2 as UFP server ports.

```
G8296(config)# system server-ports port 1,2
```

2. Turn on UFP.

```
G8296(config)# ufp enable
```

3. Configure port 1 as UFP.

```
G8296(config)# ufp port 1 enable
Warning: "Tagging/Trunk-mode" is enabled on UFP port 1
```

4. Configure virtual port.

```
G8296(config)# ufp port 1 vport 1
```

5. Configure vPort trunk mode.

```
G8296(config_ufp_vport)# network mode trunk
```

6. Configure vPort default VLAN.

```
G8296(config_ufp_vport)# network default-vlan 100
```

7. Specify QoS parameters for the vPort.

```
G8296(config_ufp_vport)# qos bandwidth min 25 (in percentage)
G8296(config_ufp_vport)# qos bandwidth max 100 (in percentage)
```

8. Enable the vPort.

```
G8296(config_ufp_vport)# enable
G8296(config_ufp_vport)# exit
```

9. Configure port 2 as UFP.

```
G8296(config)# ufp port 2 enable  
Warning: "Tagging/Trunk-mode" is enabled on UFP port 2
```

10. Configure virtual port.

```
G8296(config)# ufp port 2 vport 3
```

11. Configure vPort trunk mode.

```
G8296(config_ufp_vport)# network mode trunk
```

12. Configure vPort default VLAN.

```
G8296(config_ufp_vport)# network default-vlan 100
```

13. Ensure tagging is disabled on vPort.

```
G8296(config_ufp_vport)# no network default-tag
```

14. Specify QoS parameters for the vPort.

```
G8296(config_ufp_vport)# qos bandwidth min 25 (in percentage)  
G8296(config_ufp_vport)# qos bandwidth max 100 (in percentage)
```

15. Enable the vPort.

```
G8296(config_ufp_vport)# enable  
G8296(config_ufp_vport)# exit
```

16. Enable tagging/trunk mode on port 3.

```
G8296(config)# interface port 3  
G8296(config-if)# switchport mode trunk  
G8296(config-if)# switchport trunk native vlan 100  
G8296(config-if)# switchport trunk allowed vlan add 200,300  
G8296(config-if)# exit
```

17. Configure VLAN 200 parameters.

```
G8296(config)# vlan 200  
G8296(config-vlan)# vmember 1.1  
G8296(config-vlan)# vmember 2.3  
G8296(config-vlan)# exit
```

18. Configure VLAN 300 parameters.

```
G8296(config)# vlan 300
G8296(config-vlan)# vmember 1.1
G8296(config-vlan)# vmember 2.3
G8296(config-vlan)# exit
```

Example 3: Auto-VLAN Mode

1. Configure port 1 as a UFP server port.

```
G8296(config)# system server-ports port 1
```

2. Turn on UFP.

```
G8296(config)# ufp enable
```

3. Configure port 1 as UFP.

```
G8296(config)# ufp port 1 enable
Warning: "Tagging/Trunk-mode" is enabled on UFP port 1
```

4. Configure virtual port.

```
G8296(config)# ufp port 1 vport 1
```

5. Configure vPort default VLAN.

```
G8296(config_ufp_vport)# network default-vlan 100
```

6. Configure vPort auto mode.

```
G8296(config_ufp_vport)# network mode auto
```

Note: VLAN is dynamically added by 802.1Qbg.

7. Specify QoS parameters for the vPort.

```
G8296(config_ufp_vport)# qos bandwidth min 25           (in percentage)
G8296(config_ufp_vport)# qos bandwidth max 100        (in percentage)
```

8. Enable the vPort.

```
G8296(config_ufp_vport)# enable
G8296(config_ufp_vport)# exit
```

Example 4: Tunnel Mode

Following is an example configuration of UFP vPorts in tunnel mode.

1. Configure port 1 as a UFP server port.

```
G8296(config)# system server-ports port 1
```

2. Turn on UFP.

```
G8296(config)# ufp enable
```

3. Configure port as UFP.

```
G8296(config)# ufp port 1 enable
```

```
Warning: "Tagging/Trunk-mode" is enabled on UFP port 1
```

4. Configure virtual port.

```
G8296(config)# ufp port 1 vport 1
```

5. Configure vPort tunnel mode.

```
G8296(config_ufp_vport)# network mode tunnel
```

6. Configure vPort default VLAN.

```
G8296(config_ufp_vport)# network default-vlan 4000
```

7. Ensure tagging is disabled on vPort.

```
G8296(config_ufp_vport)# no network default-tag
```

8. Specify QoS parameters for the vPort.

```
G8296(config_ufp_vport)# qos bandwidth min 25 (in percentage)  
G8296(config_ufp_vport)# qos bandwidth max 100 (in percentage)
```

9. Enable the vPort.

```
G8296(config_ufp_vport)# enable  
G8296(config_ufp_vport)# exit
```

10. Configure tagging on port 2.

Note: Port 2 must be an upstream port.

```
G8296(config)# interface port 2
G8296(config-if)# tagpvid-ingress
G8296(config-if)# no vlan dot1q tag native
G8296(config-if)# switchport access vlan 4000
G8296(config-if)# exit
```

Example 5: FCoE Mode

Following is an example configuration of UFP vPorts in FCoE mode.

1. Configure port 1 as a UFP server port.

```
G8296(config)# system server-ports port 1
```

2. Enable CEE.

```
G8296(config)# cee enable
```

3. Enable FIPs.

```
G8296(config)# fcoe fips enable
```

4. Turn on UFP.

```
RS8296(config)# ufp enable
```

5. Configure port as UFP.

```
G8296(config)# ufp port 1 enable
Warning: "Tagging/Trunk-mode" is enabled on UFP port 1
```

6. Configure virtual port.

```
G8296(config)# ufp port 1 vport 2
```

7. Configure vPort FCoE mode.

```
G8296(config_ufp_vport)# network mode fcoe
```

8. Configure vPort default VLAN.

```
G8296(config_ufp_vport)# network default-vlan 1102
```

9. Ensure tagging is disabled on vPort.

```
G8296(config_ufp_vport)# no network default-tag
```

- Specify QoS parameters for the vPort.

```
G8296(config_ufp_vport)# qos bandwidth min 25 (in percentage)
G8296(config_ufp_vport)# qos bandwidth max 100 (in percentage)
```

- Enable the vPort.

```
G8296(config_ufp_vport)# enable
G8296(config_ufp_vport)# exit
```

- Enable tagging/trunk mode on the uplink port.

Note: Port 2 is connected to the upstream FCF.

```
G8296(config)# interface port 2
G8296(config-if)# switchport mode trunk
G8296(config-if)# switchport trunk native vlan 1
G8296(config-if)# switchport trunk allowed vlan add 1102
G8296(config-if)# exit
```

Example 6: Layer 2 Failover Configuration

While configuring a failover trigger, you cannot use the `member` command for a physical port that has vPorts configured. Instead, you must use the `vmember` command to add the vPorts as members of a failover trigger. The following example includes the commands to configure a failover trigger using a physical port 8 (a non-UFP port) and port 9 (a UFP-enabled port with vPorts 9.1, 9.2, 9.3, and 9.4).

See “[Example 1: Access Mode](#)” on page 314 for steps to configure a vPort in access mode. Follow the steps below for configuring the failover trigger:

- Enable failover globally:

```
G8296(config)# failover enable
```

- Configure trigger 1 and add monitor and control ports:

```
G8296(config)# failover trigger 1 mmon monitor member 5
G8296(config)# failover trigger 1 mmon control member 8
G8296(config)# failover trigger 1 mmon control vmember 9.1,9.2,9.3,9.4
```

Note: If you try to add a physical port (that has vPorts configured) with the “member” attribute as a member of a trigger (as opposed to the attribute “vmember”), you may see the following error message when you enable the trigger:

```
G8296(config)#failover trigger 1 ena
```

```
Failover Error: trigger 1 physical port 9 has virtual ports.
```

- Enable failover trigger:

```
G8296(config)# failover trigger 1 enable
```


Note: Layer 2 Failover with Auto Monitoring cannot be added to an Auto-VLAN Mode vPort.

Part 5: IP Routing

This section discusses Layer 3 switching functions. In addition to switching traffic at near line rates, the application switch can perform multi-protocol routing. This section discusses basic routing and advanced routing protocols:

- Basic Routing
- Policy-Based Routing
- Routed Ports
- IPv6 Host Management
- Routing Information Protocol (RIP)
- Internet Group Management Protocol (IGMP)
- Border Gateway Protocol (BGP)
- Open Shortest Path First (OSPF)
- Service Location Protocol (SLP)
- Protocol Independent Multicast (PIM)

Chapter 23. Basic IP Routing

This chapter provides configuration background and examples for using the G8296 to perform IP routing functions. The following topics are addressed in this chapter:

- [“IP Routing Benefits” on page 326](#)
- [“Routing Between IP Subnets” on page 326](#)
- [“Example of Subnet Routing” on page 327](#)
- [“ARP - Local Proxy” on page 331](#)
- [“Network Address Translation” on page 332](#)
- [“ECMP Static Routes” on page 337](#)
- [“Dynamic Host Configuration Protocol” on page 339](#)

IP Routing Benefits

The switch uses a combination of configurable IP switch interfaces and IP routing options. The switch IP routing capabilities provide the following benefits:

- Connects the server IP subnets to the rest of the backbone network.
- Provides the ability to route IP traffic between multiple Virtual Local Area Networks (VLANs) configured on the switch.

Routing Between IP Subnets

The physical layout of most corporate networks has evolved over time. Classic hub/router topologies have given way to faster switched topologies, particularly now that switches are increasingly intelligent. The G8296 is intelligent and fast enough to perform routing functions at wire speed.

The combination of faster routing and switching in a single device allows you to build versatile topologies that account for legacy configurations.

For example, consider a corporate campus that has migrated from a router-centric topology to a faster, more powerful, switch-based topology. As is often the case, the legacy of network growth and redesign has left the system with a mix of illogically distributed subnets.

This is a situation that switching alone cannot cure. Instead, the router is flooded with cross-subnet communication. This compromises efficiency in two ways:

- Routers can be slower than switches. The cross-subnet side trip from the switch to the router and back again adds two hops for the data, slowing throughput considerably.
- Traffic to the router increases, increasing congestion.

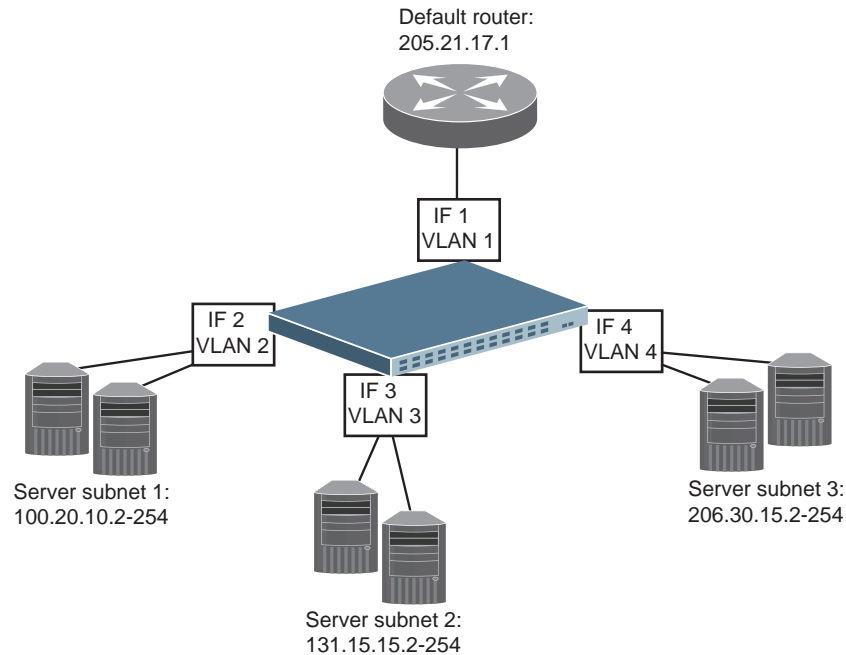
Even if every end-station could be moved to better logical subnets (a daunting task), competition for access to common server pools on different subnets still burdens the routers.

This problem is solved by using switches with built-in IP routing capabilities. Cross-subnet LAN traffic can now be routed within the switches with wire speed switching performance. This eases the load on the router and saves the network administrators from reconfiguring every end-station with new IP addresses.

Example of Subnet Routing

Consider the role of the G8296 in the following configuration example:

Figure 29. Switch-Based Routing Topology



The switch connects the Gigabit Ethernet and Fast Ethernet trunks from various switched subnets throughout one building. Common servers are placed on another subnet attached to the switch. A primary and backup router are attached to the switch on yet another subnet.

Without Layer 3 IP routing on the switch, cross-subnet communication is relayed to the default gateway (in this case, the router) for the next level of routing intelligence. The router fills in the necessary address information and sends the data back to the switch, which then relays the packet to the proper destination subnet using Layer 2 switching.

With Layer 3 IP routing in place on the switch, routing between different IP subnets can be accomplished entirely within the switch. This leaves the routers free to handle inbound and outbound traffic for this group of subnets.

Using VLANs to Segregate Broadcast Domains

If you want to control the broadcasts on your network, use VLANs to create distinct broadcast domains. Create one VLAN for each server subnet, and one for the router.

Configuration Example

This section describes the steps used to configure the example topology shown in [Figure 29 on page 327](#).

1. Assign an IP address (or document the existing one) for each router and each server.

The following IP addresses are used:

Table 35. *Subnet Routing Example: IP Address Assignments*

Subnet	Devices	IP Addresses
1	Default router	205.21.17.1
2	Web servers	100.20.10.2-254
3	Database servers	131.15.15.2-254
4	Terminal Servers	206.30.15.2-254

2. Assign an IP interface for each subnet attached to the switch.

Since there are four IP subnets connected to the switch, four IP interfaces are needed:

Table 36. *Subnet Routing Example: IP Interface Assignments*

Interface	Devices	IP Interface Address
IF 1	Default router	205.21.17.3
IF 2	Web servers	100.20.10.1
IF 3	Database servers	131.15.15.1
IF 4	Terminal Servers	206.30.15.1

3. Determine which switch ports and IP interfaces belong to which VLANs.

The following table adds port and VLAN information:

Table 37. *Subnet Routing Example: Optional VLAN Ports*

Devices	IP Interface	Switch Ports	VLAN #
Default router	1	22	1
Web servers	2	1 and 2	2
Database servers	3	3 and 4	3
Terminal Servers	4	5 and 6	4

Note: To perform this configuration, you must be connected to the switch Command Line Interface (CLI) as the administrator.

4. Add the switch ports to their respective VLANs.

The VLANs shown in [Table 37](#) are configured as follows:

```
RS8296(config)# vlan 1
RS8296(config-vlan)# exit
RS8296(config)# interface port 22      (Add ports to VLAN 1)
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 1
RS8296(config-if)# exit

RS8296(config)# vlan 2
RS8296(config-vlan)# exit
RS8296(config)# interface port 1,2    (Add ports to VLAN 2)
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 2
RS8296(config-if)# exit

RS8296(config)# vlan 3
RS8296(config-vlan)# exit
RS8296(config)# interface port 3,4    (Add ports to VLAN 3)
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 3
RS8296(config-if)# exit

RS8296(config)# vlan 4
RS8296(config-vlan)# exit
RS8296(config)# interface port 5,6    (Add ports to VLAN 4)
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 4
RS8296(config-if)# exit
```

Each time you add a port to a VLAN, you may get the following prompt:

```
Port 4 is an untagged port and its PVID is changed from 1 to 3.
```

5. Assign a VLAN to each IP interface.

Now that the ports are separated into VLANs, the VLANs are assigned to the appropriate IP interface for each subnet. From [Table 37 on page 328](#), the settings are made as follows:

```
RS8296(config)# interface ip 1 (Select IP interface 1)
RS8296(config-ip-if)# ip address 205.21.17.3
RS8296(config-ip-if)# ip netmask 255.255.255.0
RS8296(config-ip-if)# vlan 1 (Add VLAN 1)
RS8296(config-ip-if)# enable
RS8296(config-vlan)# exit
RS8296(config)# interface ip 2 (Select IP interface 2)
RS8296(config-ip-if)# ip address 100.20.10.1
RS8296(config-ip-if)# ip netmask 255.255.255.0
RS8296(config-ip-if)# vlan 2 (Add VLAN 2)
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 3 (Select IP interface 3)
RS8296(config-ip-if)# ip address 131.15.15.1
RS8296(config-ip-if)# ip netmask 255.255.255.0
RS8296(config-ip-if)# vlan 3 (Add VLAN 3)
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 4 (Select IP interface 4)
RS8296(config-ip-if)# ip address 206.30.15.1
RS8296(config-ip-if)# ip netmask 255.255.255.0
RS8296(config-ip-if)# vlan 4 (Add VLAN 4)
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
```

6. Configure the default gateway to the routers' addresses.

The default gateway allows the switch to send outbound traffic to the router:

```
RS8296(config)# ip gateway 1 address 205.21.17.1
RS8296(config)# ip gateway 1 enable
```

7. Enable IP routing.

```
RS8296(config)# ip routing
```

8. Verify the configuration.

```
RS8296(config)# show vlan
RS8296(config)# show interface information
RS8296(config)# show interface ip
```

Examine the resulting information. If any settings are incorrect, make the appropriate changes.

ARP - Local Proxy

An IP interface or a routed port that has local proxy ARP enabled allows the RackSwitch G8296 to respond to all ARP requests for IP addresses within the subnetwork, and to forward all traffic between hosts in the subnetwork. This feature is useful when hosts in a subnetwork are separated at Layer 2 with features such as Private VLAN. After responding to an ARP request, the G8296 sends an ARP request to the destination host for creating an ARP entry, if such an entry does not already exist in the ARP cache. If VRRP is enabled, the G8296 uses the virtual router MAC address of the master in the ARP response. If VRRP is not enabled, the G8296 uses the switch base MAC address.

When local proxy ARP is enabled on an interface, ICMP redirects must be disabled globally.

This feature can be enabled using the following commands:

On an IP interface:

```
RS8296(config-ip-if)# ip local-proxy-arp
```

On a routed port:

```
RS8296(config-if)# ip local-proxy-arp
```

For more information on routed ports, see [Chapter 25, "Routed Ports"](#).

Network Address Translation

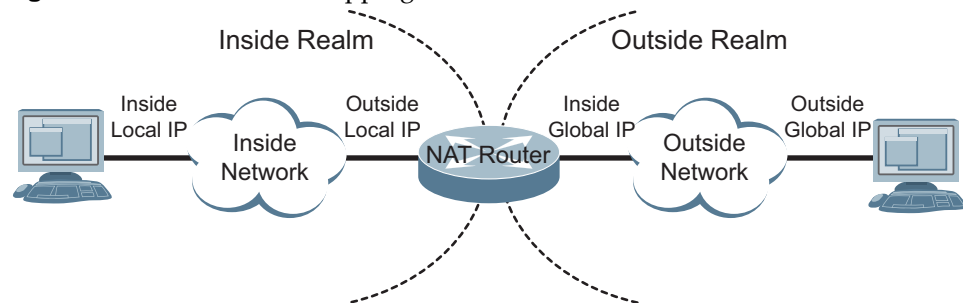
Another way to circumvent using an upstream router, this time for communication between local subnets and the rest of the backbone network, is to use the G8296's integrated Network Address Translation (NAT) functionality. NAT changes the IP addresses inside packets as they pass between different IP address spaces or realms, based on predefined mappings.

Network addresses can be segregated into two realms: the inside realm and the outside realm. This is a naming convention originating from the prevailing use case, where NAT is used to connect an internal address realm with private unregistered addresses to an external realm with globally unique registered addresses. Technically, however, the two realms are symmetrical. It is possible, for example, to define a LAN in the outside realm and the backbone network in the inside realm.

Network addresses are classified as follows:

- Inside Realm (local IPs)
 - inside local IP – the configured IP address of the inside host. All hosts from the inside realm can access the inside local address if they have a route to it.
 - outside local IP – the IP address of an outside host as it appears in the inside realm. Hosts in the inside realm use the outside local address when sending packets to a host in the outside realm.
- Outside Realm (global IPs)
 - inside global IP – the IP address of the inside host as it appears in the outside realm. Hosts in the outside realm use the inside global address as destination when sending packets to a host in the inside realm.
 - outside global IP – the configured IP address of the outside host.

Figure 30. NAT Address Mapping



When an IP packet passes from the inside realm to the outside realm, its original source IP is the inside local IP and the destination IP is the outside local IP. NAT can replace both the source (SNAT) and destination (DNAT) IP addresses as follows:

- SNAT – The inside local IP is replaced with the inside global IP
- DNAT – The outside local IP is replaced with the outside global IP

When an IP packet arrives from the outside realm to the inside realm, the following network address translations are applied:

- SNAT – The outside global IP is replaced with the outside local IP

- DNAT – The inside global IP is replaced with the inside local IP

By default, IP interfaces do not participate in NAT. To configure an interface as part of the inside realm or outside realm, use the following commands:

```
RS8296(config)# interface ip <Interface number>
RS8296(config-ip-if)# ip nat {inside|outside}
```

To exclude the interface from NAT, revert it to the default state:

```
RS8296(config)# interface ip <Interface number>
RS8296(config-ip-if)# ip nat default
```

Notes:

- All IP interfaces defined on the same VLAN or on VLANs participating in the same realm (either inside, outside or default) must be configured as members of the same realm. Setting an interface as part of a realm will automatically add all the interfaces in the VLAN to the respective realm. Reverting an interface to the default state, will automatically revert all interfaces in the VLAN to default.
- Setting up NAT on the switch's management IP or on IP addresses used for routing protocols may compromise your setup.

Static NAT

Static NAT allows one-to-one translations of the local IP addresses to the global IP addresses or of the global IP address to the local IP address.

To set up NAT for traffic passing from the inside realm to the outside realm, use the following commands:

```
RS8296(config)# ip nat inside source static <inside local IP> <inside global IP>
(SNAT)
RS8296(config)# ip nat inside destination static <outside local IP> <outside global
IP>
(DNAT)
```

For traffic passing from the outside realm to the inside realm, use the following commands:

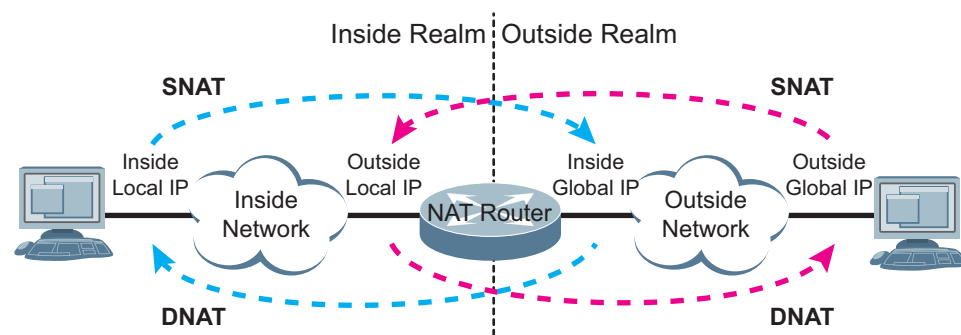
```
RS8296(config)# ip nat outside source static <outside global IP> <outside local IP>
(SNAT)
RS8296(config)# ip nat outside destination static <inside global IP> <inside local
IP>
(DNAT)
```

Unidirectional NAT Translation

By default, static NAT rules create mappings that permit translations for bidirectional flows. A static inside SNAT rule for instance, in addition to the inside local IP to inside global IP mapping for outgoing traffic, will also create a corresponding inside global IP to inside local IP mapping for incoming traffic. In other words:

- An inside SNAT also creates an outside DNAT
- An outside DNAT also creates an inside SNAT
- An outside SNAT also creates an inside DNAT
- An inside DNAT also creates an outside SNAT

Figure 31. SNAT – DNAT Equivalence



To cancel these secondary mappings, use the `one-way` parameter at the end of the static NAT commands:

```
RS8296(config)# ip nat inside source static <inside local IP> <inside global IP>  
one-way
```

Network Address Port Translation

Network Address Port Translation (NAPT) allows NAT to isolate specific traffic streams by translating socket addresses (both the IP address and port number) instead of only the IP addresses. NAPT also enables port forwarding.

To set up NAPT for a static inside SNAT for instance, use the following command:

```
RS8296(config)# ip nat inside source static {tcp|udp} <inside local IP> <local  
TCP/UDP port> <inside global IP> <global TCP/UDP port>
```

Dynamic NAT

Dynamic NAT allows mapping pools of multiple IPs in one rule, instead of individual IP network addresses. Dynamic NAT acts similarly to Static NAT, except that the entries are only temporarily added to hardware when a packet matches the translation rule and age out after a predefined time-out interval.

Up to 1000 IP address pools can be defined. To create IP address pools, use the following command:

```
RS8296(config)# ip nat pool <pool-name> <start_IP_address> <end_IP_address> netmask <netmask> [port-range <start_port> <end_port>]
```

Once the IP address pools are defined, you can configure a dynamic NAT:

```
RS8296(config)# ip nat {inside|outside} source pool <match-pool-name> pool <translation-pool> [overload] [one-way]
```

Notes:

- The `overload` parameter is redundant, and used only for Cisco compatibility.
- The `translation-pool` maps only the first IP address in the range. The parameter is defined as a pool for compatibility reasons.

Because of how the dynamic rules are defined, inside dynamic rules only allow inside nodes to initiate communication with outside nodes. Likewise, outside dynamic rules allow only outside nodes to initiate communication with inside nodes. Port forwarding can be used to circumvent this behavior and allow outside nodes to initiate communication with inside nodes based on inside dynamic rules, and inside nodes to initiate communication with outside nodes based on outside dynamic rules.

Match pools cannot overlap between NAT rules. Translation pools can. It is possible, for instance, to set up two different IP ranges to translate to the same IP address.

If a static and a dynamic rule overlap, the NAT defined in the static rule will take precedence.

NAT Proxy ARP

ARP requests from inside nodes to outside nodes will be addressed to the outside nodes' locally translated IPs (outside local IPs). Likewise, ARP requests from outside nodes to inside nodes will be addressed to the inside global IPs. To enable/disable the switch from replying to these ARP requests for outside local IPs and inside global IPs, use the following command:

```
RS8296(config)# [no] ip nat proxy-arp enable
```

Global NAT Settings

To globally turn NAT on and off without erasing the current NAT configuration, use the following commands:

```
RS8296(config)# [no] ip nat enable
```

To set up the time-out in seconds for dynamic NAT rules:

```
RS8296(config)# ip nat translation timeout <0-4294967>
```

To delete all current dynamically created NAT translations before their time-out expires:

```
RS8296(config)# clear ip nat translations
```

To delete various NAT rules and IP pools:

```
RS8296(config)# no ip nat {all|static|dynamic|pools}
```

To display current NAT settings and statistics:

```
RS8296(config)# show ip nat [statistics|translations]
```

Scaling and Limitations

The following limitations apply for NAT:

- Supported protocols for static NAT and dynamic translations: TCP, UDP
- Only ICMP Echo Request/Reply packets are supported for dynamic rules. All other ICMP packets are supported only for static NAT rules
- IPv6 is not supported
- Multicast traffic is not supported
- Maximum bidirectional (two-way) active translations (SNAT and DNAT): 1024
- Maximum unidirectional (one-way) active translations: 2048 (1024 SNAT + 1024 DNAT)
- Maximum DNAT translations (no SNAT): 1024
- Maximum bidirectional static/dynamic entries: 1024

ECMP Static Routes

Equal-Cost Multi-Path (ECMP) is a forwarding mechanism that routes packets along multiple paths of equal cost. ECMP provides equally-distributed link load sharing across the paths. The hashing algorithm used is based on the destination IP and source IP (DIPSIP) addresses or only on the source IP address (SIP). ECMP routes allow the switch to choose between several next hops toward a given destination. The switch performs periodic health checks (ping) on each ECMP gateway. If a gateway fails, it is removed from the routing table, and an SNMP trap is sent.

ECMP Route Hashing

You can configure the parameters used to perform ECMP route hashing, as follows:

- `sip`: Source IP address
- `dipsip`: Source IP address and destination IP address (default)

The ECMP hash setting applies to all ECMP routes.

Configuring ECMP Static Routes

To configure ECMP static routes, add the same route multiple times, each with the same destination IP address, but with a different gateway IP address. These routes become ECMP routes.

1. Add a static route (IP address, subnet mask, gateway, and interface number).

```
RS8296(config)# ip route 10.10.1.1 255.255.255.255 100.10.1.1 1
```

2. Add another static route with the same IP address and mask, but a different gateway address.

```
RS8296(config)# ip route 10.10.1.1 255.255.255.255 200.20.2.2 1
```

3. Select an ECMP hashing method (optional).

```
RS8296(config)# ip route ecmphash [sip|dipsip]
```

You may add up to 32 gateways for each static route.

Use the following commands to check the status of ECMP static routes:

```
RS8296(config)# show ip route static
Current static routes:
  Destination      Mask                Gateway             If    ECMP
  -----
  10.20.2.2        255.255.255.255    10.4.4.1           *
                   10.5.5.1           *
                   10.6.6.1           *
                   ...
                   10.35.35.1        *
ECMP health-check ping interval: 1
ECMP health-check retries number: 3
ECMP Hash Mechanism: dipsip
Gateway healthcheck: enabled
```

```
RS8296(config)# show ip ecmp
Current ecmp static routes:
  Destination      Mask                Gateway             If    GW Status
  -----
  10.20.2.2        255.255.255.255    10.4.4.1           up
                   10.5.5.1           up
                   10.6.6.1           up
                   ...
                   10.34.34.1        up
                   10.35.35.1        up
```

Dynamic Host Configuration Protocol

Dynamic Host Configuration Protocol (DHCP) is a transport protocol that provides a framework for automatically assigning IP addresses and configuration information to other IP hosts or clients in a large TCP/IP network. Without DHCP, the IP address must be entered manually for each network device. DHCP allows a network administrator to distribute IP addresses from a central point and automatically send a new IP address when a device is connected to a different place in the network.

The switch accepts gateway configuration parameters if they have not been configured manually. The switch ignores DHCP gateway parameters if the gateway is configured.

DHCP is an extension of another network IP management protocol, Bootstrap Protocol (BOOTP), with an additional capability of being able to allocate reusable network addresses and configuration parameters for client operation.

Built on the client/server model, DHCP allows hosts or clients on an IP network to obtain their configurations from a DHCP server, thereby reducing network administration. The most significant configuration the client receives from the server is its required IP address; (other optional parameters include the "generic" file name to be booted, the address of the default gateway, and so forth).

To enable DHCP on a switch interface, use the following command:

```
RS8296(config)# system dhcp
```

DHCP Relay Agent

DHCP is described in RFC 2131, and the DHCP relay agent supported on the G8296 is described in RFC 1542. DHCP uses UDP as its transport protocol. The client sends messages to the server on port 67 and the server sends messages to the client on port 68.

DHCP defines the methods through which clients can be assigned an IP address for a finite lease period and allowing reassignment of the IP address to another client later. Additionally, DHCP provides the mechanism for a client to gather other IP configuration parameters it needs to operate in the TCP/IP network.

In the DHCP environment, the G8296 can act as a relay agent. The DHCP relay feature enables the switch to forward a client request for an IP address to five DHCP/BOOTP servers with IP addresses that have been configured on the switch.

When a switch receives a UDP broadcast on port 67 from a DHCP client requesting an IP address, the switch acts as a proxy for the client, replacing the client source IP (SIP) and destination IP (DIP) addresses. The request is then forwarded as a UDP Unicast MAC layer message to five DHCP/BOOTP servers whose IP addresses are configured on the switch. The servers respond as a UDP Unicast message back to the switch, with the default gateway and IP address for the client. The destination IP address in the server response represents the interface address on the switch that received the client request. This interface address tells the switch on which VLAN to send the server response to the client.

To enable the DHCP relay feature on the G8296, you need to configure the DHCP/BOOTP server IP addresses on the switch. Generally, it is best to configure the switch IP interface on the client side to match the client's subnet, and configure VLANs to separate client and server subnets. The DHCP server knows from which IP subnet the newly allocated IP address will come.

In G8296 implementation, there is no need for primary or secondary servers. The client request is forwarded to the BOOTP servers configured on the switch. The use of five servers provide failover redundancy. However, no health checking is supported.

Use the following commands to configure the switch as a DHCP relay agent:

```
RS8296(config)# ip bootp-relay server 1 <IP address>
RS8296(config)# ip bootp-relay server 2 <IP address>
RS8296(config)# ip bootp-relay server 3 <IP address>
RS8296(config)# ip bootp-relay server 4 <IP address>
RS8296(config)# ip bootp-relay server 5 <IP address>
RS8296(config)# ip bootp-relay enable
RS8296(config)# show ip bootp-relay
```

Additionally, DHCP Relay functionality can be assigned on a per interface basis. Use the following commands to enable the Relay functionality:

```
RS8296(config)# interface ip <Interface number>
RS8296(config-ip-if)# relay
```

Chapter 24. Policy-Based Routing

Policy-based routing (PBR) allows the RackSwitch G8296 to forward traffic based on defined policies rather than entries in the routing table. Such policies are defined based on the protocol, source IP, or other information present in a packet header. PBR provides a mechanism for applying the defined policies based on access control lists (ACLs), and marking packets with a type of service (ToS) to provide preferential treatment.

PBR can be applied only to the ingress traffic. You can configure a PBR policy using route maps and apply the route map to an ingress interface. You need to specify the match (using ACLs) and set (using route maps) criteria in the policy. Based on the defined rules, an action is triggered. If no match is found, or the policy rule specifies that the packet be denied, the packet is routed based on an entry in the routing table.

PBR Policies and ACLs

Up to 256 ACLs can be configured for networks that use IPv4 addressing. Regular ACLs and PBR ACLs together cannot exceed the maximum ACLs supported.

ACLs are prioritized based on the ACL number. Lower numbers have higher priority. You must configure regular ACLs with lower numbers and PBR ACLs with higher numbers.

Note: You cannot apply an ACL directly to an interface and using a PBR policy at the same time.

Applying PBR ACLs

PBR ACLs must be applied to an IP interface that has a VLAN configured. In addition to the defined ACL rules, the Lenovo Network Operating System uses the VLAN ID as a matching criterion. Traffic is filtered on a per-VLAN basis rather than a per-interface basis. If multiple IP interfaces have the same VLAN ID, route maps applied to each interface are used to filter traffic on the VLAN. For example: if interface IP 10 and interface IP 11 are members of VLAN 100; interface IP 10 uses PBR ACL 410 and interface IP 11 uses PBR ACL 411. Traffic on VLAN 100 will be filtered using PBR ACLs 410 and 411.

Note: You cannot apply the PBR ACL to a Layer 2-only port.

Configuring Route Maps

A route map is used to control and modify routing information. When PBR is enabled on an interface, all incoming packets are filtered based on the criteria defined in the route maps. For packets that match the filtering criteria, the precedence or Differentiated Services Code Point (DSCP) value can be changed, or the packets can be routed/forwarded to the appropriate next hop.

PBR and dynamic routing protocols, such as Border Gateway Protocol (BGP) and Open Shortest Path First (OSPF), use route maps. You can define a maximum of 255 route maps. Route maps used by a PBR policy cannot be used by a dynamic routing protocol. You can configure a maximum of 32 access list statements in a route map. You can assign only one route map to a non-management IP interface.

You must define route map criteria using `match` and `set` commands. All sequential match clauses must be met by the packets for the set clauses to be applied.

Match Clauses

IPv4 ACLs can be used to specify the match criteria. The following match criteria can be used in a PBR ACL:

- Source IP
- Destination IP
- Protocol
- ToS
- TCP/UDP source port
- TCP/UDP destination port

If criteria other than the above are used in a PBR ACL, the switch will display an error message.

If ingress packets do not meet any of the match criteria, or if a deny statement is configured in the ACL, then the packets are routed based on the entries in the routing table.

Set Clauses

When the match clause(s) is satisfied, one of the following set clauses can be used to specify the packet forwarding criteria:

- Next hop IP address: This must be the IP address of an interface on the adjacent router. A remote router interface cannot be used to specify the next hop. Packets are forwarded to the next hop IP address. The PBR policy uses the next hops in the order you specify. If the first next hop is down, then the second next hop is used, and so on. If you specify the next hop addresses using separate statements, then the next hops are used in the order you specify, starting from top to down. A maximum of 64 unique next hops can be configured across all route maps.
- IP Differentiated Services Code Point (DSCP) value: A value used to set the DSCP value of the matching packets.

- IP precedence value: A value or keyword used to set the precedence value of the matching packets.

You can use a combination of set commands. However, you cannot use the set commands for DSCP and precedence together in the same route map.

Following are the basic steps and commands for configuring route maps.

1. Configure a route map.

```
RS8296(config)# route-map <route map number>
```

2. Define an access list statement and assign an ACL to the route map.

```
RS8296(config-route-map)# access-list <1-32> match-access-control  
<IP ACL number>
```

3. Enable the access list.

```
RS8296(config-route-map)# access-list <1-32> enable
```

4. Set next hop IP address (the IP address of an adjacent router).

```
RS8296(config-route-map)# set ip next-hop <IP address> [<nh2 IP address>]  
 [<nh3 IP address>] [<nh4 IP address>]  
 [access-list {<access list ID>|<access list range>}]
```

OR

Set IP precedence value.

```
RS8296(config-route-map)# set ip precedence <value or keyword>  
 [access-list {<access list ID>|<access list range>}]
```

OR

Set IP DSCP value.

```
RS8296(config-route-map)# set ip dscp <value>  
 [access-list {<access list ID>|<access list range>}]
```

Configuring Health Check

You can configure tracking/health check parameters for each of the next hop IP address you specify in the route map. By default, Address Resolution Protocol (ARP) resolves the next hop IP address. The ARP re-try interval is two minutes.

Use the command below to configure health check:

```
RS8296(config-route-map)# set ip next-hop verify-availability <next hop IP
address> <priority> [icmp|arp] [interval] [retry] [access-list {<access list
ID>|<access list range>}]
```

Default values:

Protocol: ARP

Interval: 2 seconds

Retry: 3 times

You must configure a separate statement for verifying health check of each next hop. A maximum of four health check statements can be included in a route map.

Note: When you configure next hops using set ip next-hop command and health check using the set ip next-hop verify-availability command in the same route map, only the health check statements will be considered.

Following is an example of a route map health check statement:

```
RS8296(config-route-map)# set ip next-hop verify-availability 12.1.1.1 10
icmp access-list 4
```

Similarly, if there are inconsistent tracking parameters for a particular next hop IP address among multiple route maps, the route map with the lowest route map number is considered.

Note: We strongly recommend that you configure health check if all/multiple next hops specified in the route map belong to the same Spanning Tree Group (STG). This is required in case of an STP topology change where all forwarding database (FDB) entries on all the ports in an STG are cleared. In such a scenario, the associated ARP entries are purged and the next hop specified in the PBR policy will not get resolved. When health check is configured, the PBR policy will route the traffic based on the second next hop that you have specified.

Example PBR Configuration

Following is an example of configuring PBR to match packets with a destination network address of 3.0.0.0. The PBR is applied to ingress packets on the IP interface 11. The next hop IP address is configured as 5.5.5.5 or 10.10.10.10.

1. Configure an ACL and specify the match criteria.

```
RS8296(config)# access-control list 100 action permit
RS8296(config)# access-control list 100 ipv4 destination-ip-address
3.0.0.0 255.0.0.0
```

2. Configure a route map.

```
RS8296(config)# route-map 126
```

3. Apply the ACL to the route map.

```
RS8296(config-route-map)# access-list 1 match-access-control 100
```

4. Set the next hop IP addresses.

```
RS8296(config-route-map)# set ip next-hop 5.5.5.5 10.10.10.10
RS8296(config-route-map)# exit
```

5. Apply the route map to an IP interface that has a VLAN configured.

```
RS8296(config)# interface ip 11
RS8296(config-ip-if)# ip policy route-map 126
RS8296(config-ip-if)# exit
RS8296(config)# exit
```

6. Verify PBR configuration.

```
RS8296# show ip policy

IP Interface      Route map
11                126
```

```
RS8296# show route-map 126

126: PBR, enabled
Match clauses:
  alist 1: access-control list 100, enabled
Set clauses:
  ip next-hop 5.5.5.5 10.10.10.10, alist all
Policy routing matches: 0 packets
```

Configuring PBR with other Features

Consider the following PBR behavior when configured with the features given below:

- DSCP: PBR ACLs can be used to remark an IP packet with a new precedence/DSCP value. PBR ACL remark statements have higher priority than the DSCP remark commands configured on ports.
- Virtual Router Redundancy Protocol (VRRP): If PBR is enabled on a VRRP IP interface, the PBR becomes effective on the interface when the switch becomes the VRRP master on that IP interface.
- Virtual Link Aggregation Group (VLAG): When configuring PBR on VLAG ports, you must configure the same PBR policy on both the VLAG peers. You cannot configure the next hop to be on a remote switch or on the VLAG ports.

Unsupported Features

PBR cannot be configured for:

- Routed ports
- Multicast traffic
- IPv6 packets
- Simple Network Management Protocol (SNMP)
- Virtual Network Interface Card (vNIC)
- Loopback Interface

Dynamic PBR

Dynamic Policy-Based Routing enables simplified configuration of multi-tenant networks. An IP interface can be set up as either a:

- downlink to a specific tenant's endpoint
- uplink for one or more tenants
- link to a service network accessible to one or more tenants

Traffic between different tenants is routed through their uplinks (firewall). Only intra-tenant and service network traffic is routed locally. When configuring interface tenant settings, ACLs are dynamically generated to route data traffic correspondingly.

Features and Limitations

The following features and limitations apply to Dynamic Policy-Based Routing:

- Tenants must not use overlapping IP subnets.
- If a tenant owns multiple subnets, data traffic between these subnets is not forwarded upstream. Instead, it is routed locally.
- Dynamic PBR is not supported in IPv6
- Tenant configuration is not supported on management interfaces
- Support for up to:
 - 64 tenants
 - 64 uplinks
 - 2 uplinks per tenant
 - 5 service networks
 - 10 subnets per tenant

Example Configuration

Following is a configuration example for a three tenants setup, two of which are supported by a local service network:

1. Set up Tenant 1 with endpoints on interfaces 5 and 6

```
RS8296(config)# interface ip 5
RS8296(config-ip-if)# tenant tenant-id 1
RS8296(config-ip-if)# interface ip 6
RS8296(config-ip-if)# tenant tenant-id 1
```

2. Set up Tenant 2 with an endpoint on interface 7

```
RS8296(config-ip-if)# interface ip 7
RS8296(config-ip-if)# tenant tenant-id 2
```

3. Set interface 8 as a common uplink for Tenants 1 and 2

```
RS8296(config-ip-if)# interface ip 8
RS8296(config-ip-if)# tenant uplink tenant-id 1,2
```

4. Set up Tenant 3 with endpoints on interfaces 9 and 10 and uplinks on interfaces 11 and 12

```
RS8296(config-ip-if)# interface ip 9
RS8296(config-ip-if)# tenant tenant-id 3
RS8296(config-ip-if)# interface ip 10
RS8296(config-ip-if)# tenant tenant-id 3
RS8296(config-ip-if)# interface ip 11
RS8296(config-ip-if)# tenant uplink tenant-id 3
RS8296(config-ip-if)# interface ip 12
RS8296(config-ip-if)# tenant uplink tenant-id 3
```

5. Set interface 13 as a link to a service network available to Tenants 1 and 3

```
RS8296(config-ip-if)# interface ip 13
RS8296(config-ip-if)# tenant services tenant-id 1,3
```

Chapter 25. Routed Ports

By default, all ports on the RackSwitch G8296 behave as switch ports, which are capable of performing Layer 2 switch functions, such as VLANs, STP, or bridging. Switch ports also provide a physical point of access for the switch IP interfaces, which can perform global Layer 3 functions, such as routing for BGP or OSPF.

However, G8296 ports can also be configured as routed ports. Routed ports are configured with their own IP address belonging to a unique Layer 3 network, and behave similar to a port on a conventional router. Routed ports are typically used for connecting to a server or to a router.

When a switch port is configured as a routed port, it forwards Layer 3 traffic and no longer performs Layer 2 switching functions.

Overview

A routed port has the following characteristics:

- Does not participate in bridging.
- Does not belong to any user-configurable VLAN.
- Does not implement any Layer 2 functionality, such as Spanning Tree Protocol (STP).
- Is always in a forwarding state.
- Can participate in IPv4 routing.
- Can be configured with basic IP protocols, such as Internet Control Message Protocol (ICMP), and with Layer 3 protocols, such as Protocol-Independent Multicast (PIM), Routing Information Protocol (RIP), Open Shortest Path First (OSPF), and Border Gateway Protocol (BGP).
- Can be configured with Internet Group Management Protocol (IGMP) querier and snooping functionality.
- Layer 3 configuration is saved even when the interface is shutdown.
- MAC address learning is always enabled.
- Tagging and port VLAN ID (PVID) tagging is disabled.
- Flooding is disabled.
- Bridge Protocol Data Unit (BPDU)-guard is disabled.
- Link Aggregation Control Protocol (LACP) is disabled.
- Multicast threshold is disabled.
- Static Multicast MAC and static unicast MAC can be configured.

Note: Ports on which LACP or portchannel is enabled cannot be changed to routed ports.

Note: Ports that have Static MAC addresses configured cannot be changed to routed ports.

When a switch port is configured as a routed port, the following configuration changes are automatically implemented:

- The port is removed from all the VLANs it belonged to.
- The port is added to an internal VLAN on which flooding is disabled. The ID of this internal VLAN could be 4094 or lower. The internal VLAN is assigned to Spanning Tree Group (STG) 1, if RSTP/PVRST is configured; or to Common Internal Spanning Tree (CIST), if MSTP is configured. You cannot change the VLAN number assigned to the routed port.
- STP is disabled and the port is set to a forwarding state.

Note: The maximum number of VLANs you can configure on the RackSwitch G8296 is 4095. This maximum number will be reduced by the number of routed ports you configure.

- All the Layer 2 configuration is lost.

When a routed port is changed back to a switch port, the following changes take place:

- All the IP configuration is lost.
- The ARP entry corresponding to the IP address is lost.
- The switch port is added to the default VLAN and STG. In case of MSTP, it is added to the CIST.
- STP is turned on.
- The switch port can participate in STG and VLAN flooding.
- Can participate in bridging.
- LACP port attributes are set to default.
- Multicast threshold remains disabled.
- BPDU guard remains disabled.
- IGMP configuration is lost.

Note: When you configure a routed port to back to a switch port, it does not restore the Layer 2 configuration it had before it was changed to a routed port.

Configuring a Routed Port

Note: You cannot configure a management port to be a routed port.

Following are the basic steps for configuring a routed port:

1. Enter the interface configuration mode for the port.

```
RS8296(config)# interface port <port number>
```

Note: You must enter only one port number. If you need to change multiple ports to routed ports, repeat the configuration steps for each port.

2. Enable routing.

```
RS8296(config-if)# no switchport
```

3. Assign an IP address.

```
RS8296(config-if)# ip address <IP address> <Subnet Mask> enable
```

4. (Optional) Enable a Layer 3 routing protocol.

```
RS8296(config-if)# ip {<ospf>|<pim>|<rip>}
```

Note: Configure the Layer 3 routing protocol-related parameters in the interface configuration mode.

Configuring OSPF on Routed Ports

The following OSPF configuration commands are supported on routed ports:

```
RS8296(config-if)# ip ospf ?  
  
area                Set area index  
cost                Set interface cost  
dead-interval       Set dead interval in seconds or milliseconds  
enable              Enable OSPF for this interface  
hello-interval      Set hello interval in seconds or milliseconds  
key                 Set authentication key  
message-digest-key  Set MD5 key ID  
passive-interface   Enable passive interface  
point-to-point      Enable point-to-point interface  
priority            Set interface router priority  
retransmit-interval Set retransmit interval in seconds  
transit-delay       Set transit delay in seconds
```

See [Chapter 32, “OSPF](#) for details on the OSPF protocol and its configuration.

OSPFv3 cannot be configured on routed ports.

OSPF Configuration Example

The following example includes the basic steps for configuring OSPF on a routed port:

```
RS8296(config)# router ospf
RS8296(config-router-ospf)# area 0 enable
RS8296(config-router-ospf)# enable
RS8296(config-router-ospf)# exit
RS8296(config)# interface port 1
RS8296(config-if)# no switchport
wait...
RS8296(config-if)# ip address 11.1.12.1 255.255.255.0 enable
wait...
RS8296(config-if)# ip ospf area 0
RS8296(config-if)# ip ospf enable
RS8296(config-if)# exit
```

Configuring RIP on Routed Ports

The following RIP configuration commands are supported on routed ports:

```
RS8296(config-if)# ip rip ?

authentication      Set IP authentication
default-action      Set default route action
enable              Enable RIP interface
listen              Enable listening to route updates
metric              Set metric
multicast-updates  Enable multicast updates
poison              Enable poisoned reverse
split-horizon       Enable split horizon
supply              Enable supplying route updates
triggered           Enable triggered updates
version             RIP version
```

See [Chapter 28, “Routing Information Protocol](#) for details on the RIP protocol and its configuration.

RIP Configuration Example

The following example includes steps for a basic RIP configuration on a routed port:

```
RS8296(config)# router rip
RS8296(config-router-rip)# enable
RS8296(config-router-rip)# exit
RS8296(config)# interface port 1
RS8296(config-if)# no switchport
wait...
RS8296(config-if)# ip address 11.1.12.1 255.255.255.0 enable
wait...
RS8296(config-if)# ip rip enable
RS8296(config-if)# exit
```


Configuring PIM on Routed Ports

The following PIM configuration commands are supported on routed ports:

```
RS8296(config-if)# ip pim ?

border-bit          Set interface as border interface
cbsr-preference     Set preference for local interface as a candidate
                    bootstrap router
component-id        Add interface to the component
dr-priority         Set designated router priority for the router
                    interface
enable              Enable PIM on this interface
hello-holdtime      Set hello message holdtime for the interface
hello-interval      Set the frequency of PIM hello messages on the
                    interface
join-prune-interval Set frequency of PIM Join or Prune interval
lan-delay           Set lan delay for the router interface
lan-prune-delay     Enable lan delay advertisement on interface
neighbor-addr       Neighbor address
neighbor-filter     Enable neighbor filter
override-interval  Set override interval for router interface
```

See [Chapter 33, "Protocol Independent Multicast"](#) for details on the PIM protocol and its configuration.

PIM Configuration Example

The following example includes the basic steps for configuring PIM on a routed port:

```
RS8296(config)# ip pim enable
RS8296(config)# interface port 26
RS8296(config-if)# no switchport
wait...
RS8296(config-if)# ip address 26.26.26.1 255.255.255.0 enable
wait...
RS8296(config-if)# ip pim enable
RS8296(config-if)# exit

RS8296(config)# ip pim component 1
RS8296(config-ip-pim-component)# rp-candidate rp-address 224.0.0.0
240.0.0.0 33.33.33.1
RS8296(config-ip-pim-component)# rp-candidate holdtime 200
RS8296(config-ip-pim-component)# exit

RS8296(config)# interface port 26
RS8296(config-if)# ip pim cbsr-preference 200
RS8296(config-if)# exit
```

Verify the configuration using the following command:

```
RS8296(config)# show ip pim interface port 26

Address      IfName/IfId Ver/Mode  Nbr   Qry      DR-Address  DR-Prio
-----
26.26.26.1  Rport  26      2/Sparse 0    30      26.26.26.1  1
```

Configuring BGP on Routed Ports

The routed port can be used to establish a TCP connection to form peer relationship with another BGP router. See [Chapter 31, "Border Gateway Protocol"](#) for details on the BGP protocol and its configuration.

The following BGP configurations are not supported on routed ports:

- Update source - configuring a local IP interface

Configuring IGMP on Routed Ports

IGMP querier and snooping can be configured on routed ports. For details, see [Chapter 29, "Internet Group Management Protocol"](#).

To configure IGMP snooping on a routed port, enter the following command in the Global Configuration mode:

```
RS8296(config)# ip igmp snoop port <routed port ID>
```

To configure fastleave on routed ports, enter the following command in the Global Configuration mode:

```
RS8296(config)# ip igmp fastleave port <routed port ID>
```

The following IGMP querier commands are supported on routed ports:

```
RS8296(config)# ip igmp querier port <routed port ID> ?  
  
election-type    Set IGMP querier type  
enable           Turn IGMP Querier on  
max-response     Set Queriers max response time  
query-interval   Set general query interval for IGMP Querier only  
robustness       Set IGMP robustness  
source-ip        Set source IP to be used for IGMP  
startup-count    Set startupcount for IGMP  
startup-interval Set startup query interval for IGMP  
version          Sets the operating version of the IGMP snooping  
switch
```

Limitations

Following features/configurations are not supported on routed ports:

- Subinterfaces
- BPDU Guard
- Broadcast Threshold
- Multicast Threshold
- Link Aggregation Control Protocol (LACP)
- Static Trunking
- Fibre Channel over Ethernet (FCoE)
- Converged Enhanced Ethernet (CEE)
- IPv6
- IP Security (IPsec)
- Internet Key Exchange version 2 (IKEv2)
- Virtual Router Redundancy Protocol (VRRP)
- Policy-based Routing (PBR)
- Hotlinks
- Failover
- 802.1X
- Dynamic Host Configuration Protocol (DHCP)
- BOOTP
- Simple Network Management Protocol (SNMP)
- IGMP Relay
- Static Multicast Routes
- Static Mrouter Port
- Management Port

Chapter 26. Internet Protocol Version 6

Internet Protocol version 6 (IPv6) is a network layer protocol intended to expand the network address space. IPv6 is a robust and expandable protocol that meets the need for increased physical address space. The switch supports the following RFCs for IPv6-related features:

- RFC 1981
- RFC 2404
- RFC 2410
- RFC 2451
- RFC 2460
- RFC 2474
- RFC 2526
- RFC 2711
- RFC 2740
- RFC 3289
- RFC 3306
- RFC 3307
- RFC 3411
- RFC 3412
- RFC 3413
- RFC 3414
- RFC 3484
- RFC 3602
- RFC 3810
- RFC 3879
- RFC 4007
- RFC 4213
- RFC 4291
- RFC 4292
- RFC 4293
- RFC 4301
- RFC 4302
- RFC 4303
- RFC 4306
- RFC 4307
- RFC 4443
- RFC 4552
- RFC 4718
- RFC 4835
- RFC 4861
- RFC 4862
- RFC 5095
- RFC 5114
- RFC 5340

This chapter describes the basic configuration of IPv6 addresses and how to manage the switch via IPv6 host management.

IPv6 Limitations

The following IPv6 features are not supported in this release.

- Dynamic Host Control Protocol for IPv6 (DHCPv6)
- Border Gateway Protocol for IPv6 (BGP)
- Routing Information Protocol for IPv6 (RIPng)

Most other Lenovo Network Operating System 8.2 features permit IP addresses to be configured using either IPv4 or IPv6 address formats. However, the following switch features support IPv4 only:

- Bootstrap Protocol (BOOTP) and DHCP
- RADIUS, TACACS+ and LDAP
- VMware Virtual Center (vCenter) for VMready
- Routing Information Protocol (RIP)
- Border Gateway Protocol (BGP)
- Protocol Independent Multicast (PIM)
- Virtual Router Redundancy Protocol (VRRP)
- sFlow

IPv6 Address Format

The IPv6 address is 128 bits (16 bytes) long and is represented as a sequence of eight 16-bit hex values, separated by colons.

Each IPv6 address has two parts:

- Subnet prefix representing the network to which the interface is connected
- Local identifier, either derived from the MAC address or user-configured

The preferred hexadecimal format is as follows:

```
xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx
```

Example IPv6 address:

```
FEDC:BA98:7654:BA98:FEDC:1234:ABCD:5412
```

Some addresses can contain long sequences of zeros. A single contiguous sequence of zeros can be compressed to :: (two colons). For example, consider the following IPv6 address:

```
FE80:0:0:0:2AA:FF:FA:4CA2
```

The address can be compressed as follows:

```
FE80::2AA:FF:FA:4CA2
```

Unlike IPv4, a subnet mask is not used for IPv6 addresses. IPv6 uses the subnet prefix as the network identifier. The prefix is the part of the address that indicates the bits that have fixed values or are the bits of the subnet prefix. An IPv6 prefix is written in address/prefix-length notation. For example, in the following address, 64 is the network prefix:

```
21DA:D300:0000:2F3C::/64
```

IPv6 addresses can be either user-configured or automatically configured. Automatically configured addresses always have a 64-bit subnet prefix and a 64-bit interface identifier. In most implementations, the interface identifier is derived from the switch's MAC address, using a method called EUI-64.

Most Networking OS 8.2 features permit IP addresses to be configured using either IPv4 or IPv6 address formats. Throughout this manual, *IP address* is used in places where either an IPv4 or IPv6 address is allowed. In places where only one type of address is allowed, the type (*IPv4* or *IPv6*) is specified.

IPv6 Address Types

IPv6 supports three types of addresses: unicast (one-to-one), multicast (one-to-many), and anycast (one-to-nearest). Multicast addresses replace the use of broadcast addresses.

Unicast Address

Unicast is a communication between a single host and a single receiver. Packets sent to a unicast address are delivered to the interface identified by that address. IPv6 defines the following types of unicast address:

- **Global Unicast address:** An address that can be reached and identified globally. Global Unicast addresses use the high-order bit range up to FF00, therefore all non-multicast and non-link-local addresses are considered to be global unicast. A manually configured IPv6 address must be fully specified. Autoconfigured IPv6 addresses are comprised of a prefix combined with the 64-bit EUI. RFC 4291 defines the IPv6 addressing architecture.

The interface ID must be unique within the same subnet.

- **Link-local unicast address:** An address used to communicate with a neighbor on the same link. Link-local addresses use the format FE80 : : EUI

Link-local addresses are designed to be used for addressing on a single link for purposes such as automatic address configuration, neighbor discovery, or when no routers are present.

Routers must not forward any packets with link-local source or destination addresses to other links.

Multicast

Multicast is communication between a single host and multiple receivers. Packets are sent to all interfaces identified by that address. An interface may belong to any number of multicast groups.

A multicast address (FF00 - FFFF) is an identifier for a group interface. The multicast address most often encountered is a solicited-node multicast address using prefix FF02 : : 1 : FF00 : 0000 / 104 with the low-order 24 bits of the unicast or anycast address.

The following well-known multicast addresses are pre-defined. The group IDs defined in this section are defined for explicit scope values, as follows:

FF00 : : : : : 0 through FF0F : : : : : 0

Anycast

Packets sent to an anycast address or list of addresses are delivered to the nearest interface identified by that address. Anycast is a communication between a single sender and a list of addresses.

Anycast addresses are allocated from the unicast address space, using any of the defined unicast address formats. Thus, anycast addresses are syntactically indistinguishable from unicast addresses. When a unicast address is assigned to

more than one interface, thus turning it into an anycast address, the nodes to which the address is assigned must be explicitly configured to know that it is an anycast address.

IPv6 Address Autoconfiguration

IPv6 supports the following types of address autoconfiguration:

- **Stateful address configuration**

Address configuration is based on the use of a stateful address configuration protocol, such as DHCPv6, to obtain addresses and other configuration options.

- **Stateless address configuration**

Address configuration is based on the receipt of Router Advertisement messages that contain one or more Prefix Information options.

N/OS 8.2 supports stateless address configuration.

Stateless address configuration allows hosts on a link to configure themselves with link-local addresses and with addresses derived from prefixes advertised by local routers. Even if no router is present, hosts on the same link can configure themselves with link-local addresses and communicate without manual configuration.

IPv6 Interfaces

Each IPv6 interface supports multiple IPv6 addresses. You can manually configure up to two IPv6 addresses for each interface, or you can allow the switch to use stateless autoconfiguration.

You can manually configure two IPv6 addresses for each interface, as follows:

- Initial IPv6 address is a global unicast or anycast address.

```
RS8296(config)# interface ip <interface number>  
RS8296(config-ip-if)# ipv6 address <IPv6 address>
```

Note that you cannot configure both addresses as anycast. If you configure an anycast address on the interface you must also configure a global unicast address on that interface.

- Second IPv6 address can be a unicast or anycast address.

```
RS8296(config-ip-if)# ipv6 secaddr6 <IPv6 address>  
RS8296(config-ip-if)# exit
```

You cannot configure an IPv4 address on an IPv6 management interface. Each interface can be configured with only one address type: either IPv4 or IPv6, but not both. When changing between IPv4 and IPv6 address formats, the prior address settings for the interface are discarded.

Each IPv6 interface can belong to only one VLAN. Each VLAN can support only one IPv6 interface. Each VLAN can support multiple IPv4 interfaces.

Use the following commands to configure the IPv6 gateway:

```
RS8296(config)# ip gateway6 1 address <IPv6 address>  
RS8296(config)# ip gateway6 1 enable
```

IPv6 gateway 1 is reserved for IPv6 data interfaces. IPv6 gateway 4 is the default IPv6 management gateway.

Neighbor Discovery

Neighbor Discovery Overview

The switch uses Neighbor Discovery protocol (ND) to gather information about other router and host nodes, including the IPv6 addresses. Host nodes use ND to configure their interfaces and perform health detection. ND allows each node to determine the link-layer addresses of neighboring nodes, and to keep track of each neighbor's information. A neighboring node is a host or a router that is linked directly to the switch. The switch supports Neighbor Discovery as described in RFC 4861.

Neighbor Discover messages allow network nodes to exchange information, as follows:

- *Neighbor Solicitations* allow a node to discover information about other nodes.
- *Neighbor Advertisements* are sent in response to Neighbor Solicitations. The Neighbor Advertisement contains information required by nodes to determine the link-layer address of the sender, and the sender's role on the network.
- IPv6 hosts use *Router Solicitations* to discover IPv6 routers. When a router receives a Router Solicitation, it responds immediately to the host.
- Routers uses *Router Advertisements* to announce its presence on the network, and to provide its address prefix to neighbor devices. IPv6 hosts listen for Router Advertisements, and uses the information to build a list of default routers. Each host uses this information to perform autoconfiguration of IPv6 addresses.
- *Redirect messages* are sent by IPv6 routers to inform hosts of a better first-hop address for a specific destination. Redirect messages are only sent by routers for unicast traffic, are only unicast to originating hosts, and are only processed by hosts.

ND configuration for general advertisements, flags, and interval settings, as well as for defining prefix profiles for router advertisements, is performed on a per-interface basis using the following command path:

```
RS8296(config)# interface ip <interface number>  
RS8296(config-ip-if)# [no] ipv6 nd ?  
RS8296(config-ip-if)# exit
```

To add or remove entries in the static neighbor cache, use the following command path:

```
RS8296(config)# [no] ip neighbors ?
```

Host vs. Router

Each IPv6 interface can be configured as a router node or a host node, as follows:

- A router node's IP address is configured manually. Router nodes can send Router Advertisements.
- A host node's IP address is autoconfigured. Host nodes listen for Router Advertisements that convey information about devices on the network.

Note: When IP forwarding is turned on, all IPv6 interfaces configured on the switch can forward packets.

You can configure each IPv6 interface as either a host node or a router node. You can manually assign an IPv6 address to an interface in host mode, or the interface can be assigned an IPv6 address by an upstream router, using information from router advertisements to perform stateless auto-configuration.

To set an interface to host mode, use the following command:

```
RS8296(config)# interface ip <interface number>  
RS8296(config-ip-if)# ip6host  
RS8296(config-ip-if)# exit
```

The G8296 supports up to 1156 IPv6 routes.

Supported Applications

The following applications have been enhanced to provide IPv6 support.

- **Ping**

The **ping** command supports IPv6 addresses. Use the following format to ping an IPv6 address:

```
ping <host name>|<IPv6 address> [<tries (0-4294967295)>] [data-port]  
[mgmt-port]
```

- **Traceroute**

The **traceroute** command supports IPv6 addresses (but not link-local addresses).

Use the following format to perform a traceroute to an IPv6 address:

```
traceroute <host name>| <IPv6 address> [<max-hops (1-32)>  
[<msec delay (1-4294967295)>]]
```

- **Telnet server**

The **telnet** command supports IPv6 addresses (but not link-local addresses).

Use the following format to Telnet into an IPv6 interface on the switch:

```
telnet <host name>| <IPv6 address> [<port>]
```

- **Telnet client**

The **telnet** command supports IPv6 addresses (but not link-local addresses).

Use the following format to Telnet to an IPv6 address:

```
telnet <host name>| <IPv6 address> [<port>]
```

- **HTTP/HTTPS**

The HTTP/HTTPS servers support both IPv4 and IPv6 connections.

- **SSH**

Secure Shell (SSH) connections over IPv6 are supported (but not link-local addresses). The following syntax is required from the client:

```
ssh -u <IPv6 address>
```

Example:

```
ssh -u 2001:2:3:4:0:0:0:142
```

- **TFTP**

The TFTP commands support both IPv4 and IPv6 addresses. Link-local addresses are not supported.

- **FTP**

The FTP commands support both IPv4 and IPv6 addresses. Link-local addresses are not supported.

- **DNS client**

DNS commands support both IPv4 and IPv6 addresses. Link-local addresses are not supported. Use the following command to specify the type of DNS query to be sent first:

```
RS8296(config)# ip dns ipv6 request-version {ipv4|ipv6}
```

If you set the request version to `ipv4`, the DNS application sends an **A** query first, to resolve the hostname with an IPv4 address. If no **A** record is found for that hostname (no IPv4 address for that hostname) an **AAAA** query is sent to resolve the hostname with a IPv6 address.

If you set the request version to `ipv6`, the DNS application sends an **AAAA** query first, to resolve the hostname with an IPv6 address. If no **AAAA** record is found for that hostname (no IPv6 address for that hostname) an **A** query is sent to resolve the hostname with an IPv4 address.

Configuration Guidelines

When you configure an interface for IPv6, consider the following guidelines:

- Support for subnet router anycast addresses is not available.
- A single interface can accept either IPv4 or IPv6 addresses, but not both IPv4 and IPv6 addresses.
- A single interface can accept multiple IPv6 addresses.
- A single interface can accept only one IPv4 address.
- If you change the IPv6 address of a configured interface to an IPv4 address, all IPv6 settings are deleted.
- A single VLAN can support only one IPv6 interface.
- Health checks are not supported for IPv6 gateways.
- IPv6 interfaces support Path MTU Discovery. The CPU's MTU is fixed at 1500 bytes.
- Support for jumbo frames (1,500 to 9,216 byte MTUs) is limited. Any jumbo frames intended for the CPU must be fragmented by the remote node. The switch can re-assemble fragmented packets up to 9k. It can also fragment and transmit jumbo packets received from higher layers.

IPv6 Configuration Examples

This section provides steps to configure IPv6 on the switch.

IPv6 Example 1

The following example uses IPv6 host mode to autoconfigure an IPv6 address for the interface. By default, the interface is assigned to VLAN 1.

1. Enable IPv6 host mode on an interface.

```
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip6host
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
```

2. Configure the IPv6 default gateway.

```
RS8296(config)# ip gateway6 1 address
                2001:BA98:7654:BA98:FEDC:1234:ABCD:5412
RS8296(config)# ip gateway6 1 enable
```

3. Verify the interface address.

```
RS8296(config)# show interface ip 2
```

IPv6 Example 2

Use the following example to manually configure IPv6 on an interface.

1. Assign an IPv6 address and prefix length to the interface.

```
RS8296(config)# interface ip 3
RS8296(config-ip-if)# ipv6 address
                2001:BA98:7654:BA98:FEDC:1234:ABCD:5214
RS8296(config-ip-if)# ipv6 prefixlen 64
RS8296(config-ip-if)# ipv6 seccaddr6 2003::1 32
RS8296(config-ip-if)# vlan 2
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
```

The secondary IPv6 address is compressed, and the prefix length is 32.

2. Configure the IPv6 default gateway.

```
RS8296(config)# ip gateway6 1 address
                2001:BA98:7654:BA98:FEDC:1234:ABCD:5412
RS8296(config)# ip gateway6 1 enable
```

3. Configure router advertisements for the interface (optional)

```
RS8296(config)# interface ip 3
RS8296(config-ip-if)# no ipv6 nd suppress-ra
```


4. Verify the configuration.

```
RS8296(config-ip-if)# show layer3
```

Chapter 27. IPsec with IPv6

Internet Protocol Security (IPsec) is a protocol suite for securing Internet Protocol (IP) communications by authenticating and encrypting each IP packet of a communication session. IPsec also includes protocols for establishing mutual authentication between agents at the beginning of the session and negotiation of cryptographic keys to be used during the session.

Since IPsec was implemented in conjunction with IPv6, all implementations of IPv6 must contain IPsec. To support the National Institute of Standards and Technology (NIST) recommendations for IPv6 implementations, Lenovo Network Operating System IPv6 feature compliance has been extended to include the following IETF RFCs, with an emphasis on IP Security (IPsec) and Internet Key Exchange version 2, and authentication/confidentiality for OSPFv3:

- RFC 4301 for IPv6 security
- RFC 4302 for the IPv6 Authentication Header
- RFCs 2404, 2410, 2451, 3602, and 4303 for IPv6 Encapsulating Security Payload (ESP), including NULL encryption, CBC-mode 3DES and AES ciphers, and HMAC-SHA-1-96.
- RFCs 4306, 4307, 4718, and 4835 for IKEv2 and cryptography
- RFC 4552 for OSPFv3 IPv6 authentication/confidentiality
- RFC 5114 for Diffie-Hellman groups

Note: This implementation of IPsec supports DH groups 1, 2, 5, 14, and 24.

The following topics are discussed in this chapter:

- [“IPsec Protocols” on page 372](#)
- [“Using IPsec with the RackSwitch G8296” on page 373](#)

IPsec Protocols

The Networking OS implementation of IPsec supports the following protocols:

- Authentication Header (AH)
AHs provide connectionless integrity and data origin authentication for IP packets, and provide protection against replay attacks. In IPv6, the AH protects the AH itself, the Destination Options extension header after the AH, and the IP payload. It also protects the fixed IPv6 header and all extension headers before the AH, except for the mutable fields DSCP, ECN, Flow Label, and Hop Limit. AH is defined in RFC 4302.
- Encapsulating Security Payload (ESP)
ESPs provide confidentiality, data origin authentication, integrity, an anti-replay service (a form of partial sequence integrity), and some traffic flow confidentiality. ESPs may be applied alone or in combination with an AH. ESP is defined in RFC 4303.
- Internet Key Exchange Version 2 (IKEv2)
IKEv2 is used for mutual authentication between two network elements. An IKE establishes a security association (SA) that includes shared secret information to efficiently establish SAs for ESPs and AHs, and a set of cryptographic algorithms to be used by the SAs to protect the associated traffic. IKEv2 is defined in RFC 4306.

Using IKEv2 as the foundation, IPsec supports ESP for encryption and/or authentication, and/or AH for authentication of the remote partner.

Both ESP and AH rely on security associations. A security association (SA) is the bundle of algorithms and parameters (such as keys) that encrypt and authenticate a particular flow in one direction.

Using IPsec with the RackSwitch G8296

IPsec supports the fragmentation and reassembly of IP packets that occurs when data goes to and comes from an external device. The RackSwitch G8296 acts as an end node that processes any fragmentation and reassembly of packets but does not forward the IPsec traffic. The IKEv2 key must be authenticated before you can use IPsec.

The security protocol for the session key is either ESP or AH. Outgoing packets are labeled with the SA SPI (Security Parameter Index), which the remote device will use in its verification and decryption process.

Every outgoing IPv6 packet is checked against the IPsec policies in force. For each outbound packet, after the packet is encrypted, the software compares the packet size with the MTU size that it either obtains from the default minimum maximum transmission unit (MTU) size (1500) or from path MTU discovery. If the packet size is larger than the MTU size, the receiver drops the packet and sends a message containing the MTU size to the sender. The sender then fragments the packet into smaller pieces and retransmits them using the correct MTU size.

The maximum traffic load for each IPsec packet is limited to the following:

- IKEv2 SAs: 5
- IPsec SAs: 10 (5 SAs in each direction)
- SPDs: 20 (10 policies in each direction)

IPsec is implemented as a software cryptography engine designed for handling control traffic, such as network management. IPsec is not designed for handling data traffic, such as a VPN.

Setting up Authentication

Before you can use IPsec, you need to have key policy authentication in place. There are two types of key policy authentication:

- Preshared key (default)

The parties agree on a shared, secret key that is used for authentication in an IPsec policy. During security negotiation, information is encrypted before transmission by using a session key created by using a Diffie-Hellman calculation and the shared, secret key. Information is decrypted on the receiving end using the same key. One IPsec peer authenticates the other peer's packet by decryption and verification of the hash inside the packet (the hash inside the packet is a hash of the preshared key). If authentication fails, the packet is discarded.

- Digital certificate (using RSA algorithms)

The peer being validated must hold a digital certificate signed by a trusted Certificate Authority and the private key for that digital certificate. The side performing the authentication only needs a copy of the trusted certificate authorities digital certificate. During IKEv2 authentication, the side being validated sends a copy of the digital certificate and a hash value signed using the private key. The certificate can be either generated or imported.

Note: During the IKEv2 negotiation phase, the digital certificate takes precedence over the preshared key.

Creating an IKEv2 Proposal

With IKEv2, a single policy can have multiple encryption and authentication types, as well as multiple integrity algorithms.

To create an IKEv2 proposal:

1. Enter IKEv2 proposal mode.

```
RS8296(config)# ikev2 proposal
```

2. Set the DES encryption algorithm.

```
RS8296(config-ikev2-prop)# encryption 3des|aes-cbc (default: 3des)
```

3. Set the authentication integrity algorithm type.

```
RS8296(config-ikev2-prop)# integrity sha1 (default: sha1)
```

4. Set the Diffie-Hellman group.

```
RS8296(config-ikev2-prop)# group 24 (default: 24)
```

Importing an IKEv2 Digital Certificate

To import an IKEv2 digital certificate for authentication:

1. Import the CA certificate file.
2. Import the host key file.
3. Import the host certificate file.

Note: When prompted for the port to use for download the file, if you used a management port to connect the switch to the server, enter **mgmt**, otherwise enter **data**.

Generating an IKEv2 Digital Certificate

To create an IKEv2 digital certificate for authentication:

1. Create an HTTPS certificate defining the information you want to be used in the various fields.

```
Country Name (2 letter code) [US]: <country code>
State or Province Name (full name) [CA]: <state>
Locality Name (eg, city) [Santa Clara]: <city>
Organization Name (eg, company) [Lenovo]: <company>
Organizational Unit Name (eg, section) [Engineering]: <org. unit>
Common Name (eg, YOUR name) []: <your current IP address>
Email (eg, email address) []: <email address>
Confirm generating certificate? [y/n]: y
Generating certificate. Please wait (approx 30 seconds)
restarting SSL agent
```

2. Save the HTTPS certificate.

The certificate is valid only until the switch is rebooted. To save the certificate so that it is retained beyond reboot or power cycles, use the following command:

3. Enable IKEv2 RSA-signature authentication:

Enabling IKEv2 Preshared Key Authentication

To set up IKEv2 preshared key authentication:

1. Enter the local preshared key.

```
RS8296(config)# ikev2 preshare-key local <preshared key, a string of 1-256 chars>
```

2. If asymmetric authentication is supported, enter the remote key:

```
RS8296(config)# ikev2 preshare-key remote <preshared key> <IPv6 host>
```

where the following parameters are used:

- *preshared key* A string of 1-256 characters
- *IPv6 host* An IPv6-format host, such as "3000::1"

3. Set up the IKEv2 identification type by entering *one* of the following commands:

To disable IKEv2 RSA-signature authentication method and enable preshared key authentication, enter:

Setting Up a Key Policy

When configuring IPsec, you must define a key policy. This key policy can be either manual or dynamic. Either way, configuring a policy involves the following steps:

- Create a transform set—This defines which encryption and authentication algorithms are used.
- Create a traffic selector—This describes the packets to which the policy applies.

- Establish an IPsec policy.
 - Apply the policy.
1. To define which encryption and authentication algorithms are used, create a transform set:

where the following parameters are used:

- *transform ID* A number from 1-10
- *encryption method* One of the following: **esp-3des** | **esp-aes-cbc** | **esp-null**
- *integrity algorithm* One of the following: **esp-sha1**
- *AH authentication algorithm* One of the following: **ah-sha1**

2. Decide whether to use tunnel or transport mode. The default mode is transport.

```
RS8296(config)# ipsec transform-set tunnel|transport
```

3. To describe the packets to which this policy applies, create a traffic selector using the following command:

```
RS8296(config)# ipsec traffic-selector <traffic selector number> permit|deny
any|icmp <type|any> |tcp > <source IP address|any> <destination IP address|any> [<prefix
length>]
```

where the following parameters are used:

- *traffic selector number* an integer from 1-10
- **permit|deny** whether or not to permit IPsec encryption of traffic that meets the criteria specified in this command
- **any** apply the selector to any type of traffic
- **icmp <type>|any** only apply the selector only to ICMP traffic of the specified *type* (an integer from 1-255) or to any ICMP traffic
- **tcp** only apply the selector to TCP traffic
- *source IP address|any* the source IP address in IPv6 format or “any” source
- *destination IP address|any* the destination IP address in IPv6 format or “any” destination
- *prefix length* (Optional) the length of the destination IPv6 prefix; an integer from 1-128

Permitted traffic that matches the policy in force is encrypted, while denied traffic that matches the policy in force is dropped. Traffic that does not match the policy bypasses IPsec and passes through *clear* (unencrypted).

4. Choose whether to use a manual or a dynamic policy.

Using a Manual Key Policy

A manual policy involves configuring policy and manual SA entries for local and remote peers.

To configure a manual key policy, you need:

- The IP address of the peer in IPv6 format (for example, “3000::1”).
- Inbound/Outbound session keys for the security protocols.

You can then assign the policy to an interface. The peer represents the other end of the security association. The security protocol for the session key can be either ESP or AH.

To create and configure a manual policy:

1. Enter a manual policy to configure.
2. Configure the policy.

where the following parameters are used:

- | | |
|---|---|
| • <i>peer's IPv6 address</i> | The IPv6 address of the peer (for example, 3000::1) |
| • <i>IPsec traffic-selector</i> | A number from 1-10 |
| • <i>IPsec of transform-set</i> | A number from 1-10 |
| • <i>inbound AH IPsec key</i> | The inbound AH key code, in hexadecimal |
| • <i>inbound AH IPsec SPI</i> | A number from 256-4294967295 |
| • <i>inbound ESP cipher key</i> | The inbound ESP key code, in hexadecimal |
| • <i>inbound ESP SPI</i> | A number from 256-4294967295 |
| • <i>inbound ESP authenticator key</i> | The inbound ESP authenticator key code, in hexadecimal |
| • <i>outbound AH IPsec key</i> | The outbound AH key code, in hexadecimal |
| • <i>outbound AH IPsec SPI</i> | A number from 256-4294967295 |
| • <i>outbound ESP cipher key</i> | The outbound ESP key code, in hexadecimal |
| • <i>outbound ESP SPI</i> | A number from 256-4294967295 |
| • <i>outbound ESP authenticator key</i> | The outbound ESP authenticator key code, in hexadecimal |

Notes:

- When configuring a manual policy ESP, the ESP authenticator key is optional.
- If using third-party switches, the IPsec manual policy session key must be of fixed length as follows:
 - For AH key: SHA1 is 20 bytes; MD5 is 16 bytes
 - For ESP cipher key: 3DES is 24 bytes; AES-cbc is 24 bytes; DES is 8 bytes
 - For ESP auth key: SHA1 is 20 bytes; MD5 is 16 bytes

3. After you configure the IPSec policy, you need to apply it to the interface to enforce the security policies on that interface and save it to keep it in place after a reboot. To accomplish this, enter:

Using a Dynamic Key Policy

When you use a dynamic key policy, the first packet triggers IKE and sets the IPsec SA and IKEv2 SA. The initial packet negotiation also determines the lifetime of the algorithm, or how long it stays in effect. When the key expires, a new key is automatically created. This helps prevent break-ins.

To configure a dynamic key policy:

1. Choose a dynamic policy to configure.
2. Configure the policy.

where the following parameters are used:

- *peer's IPv6 address* The IPv6 address of the peer (for example, 3000::1)
- *index of traffic-selector* A number from 1-10
- *index of transform-set* A number from 1-10
- *SA lifetime, in seconds* The length of time the SA is to remain in effect; an integer from 120-86400
- **pfs enable|disable** Whether to enable or disable the perfect forward security feature. The default is **disable**.

Note: In a dynamic policy, the AH and ESP keys are created by IKEv2.

3. After you configure the IPsec policy, you need to apply it to the interface to enforce the security policies on that interface and save it to keep it in place after a reboot. To accomplish this, enter:

Chapter 28. Routing Information Protocol

In a routed environment, routers communicate with one another to keep track of available routes. Routers can learn about available routes dynamically using the Routing Information Protocol (RIP). Lenovo Network Operating System software supports RIP version 1 (RIPv1) and RIP version 2 (RIPv2) for exchanging TCP/IPv4 route information with other routers.

Note: Networking OS 8.2 does not support IPv6 for RIP.

Distance Vector Protocol

RIP is known as a distance vector protocol. The vector is the network number and next hop, and the distance is the metric associated with the network number. RIP identifies network reachability based on metric, and metric is defined as hop count. One hop is considered to be the distance from one switch to the next, which typically is 1.

When a switch receives a routing update that contains a new or changed destination network entry, the switch adds 1 to the metric value indicated in the update and enters the network in the routing table. The IPv4 address of the sender is used as the next hop.

Stability

RIP includes a number of other stability features that are common to many routing protocols. For example, RIP implements the split horizon and hold-down mechanisms to prevent incorrect routing information from being propagated.

RIP prevents routing loops from continuing indefinitely by implementing a limit on the number of hops allowed in a path from the source to a destination. The maximum number of hops in a path is 15. The network destination network is considered unreachable if increasing the metric value by 1 causes the metric to be 16 (that is infinity). This limits the maximum diameter of a RIP network to less than 16 hops.

RIP is often used in stub networks and in small autonomous systems that do not have many redundant paths.

Routing Updates

RIP sends routing-update messages at regular intervals and when the network topology changes. Each router “advertises” routing information by sending a routing information update every 30 seconds. If a router doesn’t receive an update from another router for 180 seconds, those routes provided by that router are declared invalid. The routes are removed from the routing table, but they remain in the RIP routes table. After another 120 seconds without receiving an update for those routes, the routes are removed from respective regular updates.

When a router receives a routing update that includes changes to an entry, it updates its routing table to reflect the new route. The metric value for the path is increased by 1, and the sender is indicated as the next hop. RIP routers maintain only the best route (the route with the lowest metric value) to a destination.

For more information, see the Configuration section, Routing Information Protocol Configuration in the *Lenovo Network Operating System Command Reference*.

RIPv1

RIP version 1 use broadcast User Datagram Protocol (UDP) data packets for the regular routing updates. The main disadvantage is that the routing updates do not carry subnet mask information. Hence, the router cannot determine whether the route is a subnet route or a host route. It is of limited usage after the introduction of RIPv2. For more information about RIPv1 and RIPv2, refer to RFC 1058 and RFC 2453.

RIPv2

RIPv2 is the most popular and preferred configuration for most networks. RIPv2 expands the amount of useful information carried in RIP messages and provides a measure of security. For a detailed explanation of RIPv2, refer to RFC 1723 and RFC 2453.

RIPv2 improves efficiency by using multicast UDP (address 224.0.0.9) data packets for regular routing updates. Subnet mask information is provided in the routing updates. A security option is added for authenticating routing updates, by using a shared password. N/OS supports using clear password for RIPv2.

RIPv2 in RIPv1 Compatibility Mode

N/OS allows you to configure RIPv2 in RIPv1 compatibility mode, for using both RIPv2 and RIPv1 routers within a network. In this mode, the regular routing updates use broadcast UDP data packet to allow RIPv1 routers to receive those packets. With RIPv1 routers as recipients, the routing updates have to carry natural or host mask. Hence, it is not a recommended configuration for most network topologies.

Note: When using both RIPv1 and RIPv2 within a network, use a single subnet mask throughout the network.

RIP Features

N/OS provides the following features to support RIPv1 and RIPv2:

Poison

Simple split horizon in RIP scheme omits routes learned from one neighbor in updates sent to that neighbor. That is the most common configuration used in RIP, that is setting this Poison to DISABLE. Split horizon with poisoned reverse includes such routes in updates, but sets their metrics to 16. The disadvantage of using this feature is the increase of size in the routing updates.

Triggered Updates

Triggered updates are an attempt to speed up convergence. When Triggered Updates is enabled, whenever a router changes the metric for a route, it sends update messages almost immediately, without waiting for the regular update interval. It is recommended to enable Triggered Updates.

Multicast

RIPv2 messages use IPv4 multicast address (224.0.0.9) for periodic broadcasts. Multicast RIPv2 announcements are not processed by RIPv1 routers. IGMP is not needed since these are inter-router messages which are not forwarded.

To configure RIPv2 in RIPv1 compatibility mode, set multicast to `disable`, and set version to `both`.

Default

The RIP router can listen and supply a default route, usually represented as IPv4 0.0.0.0 in the routing table. When a router does not have an explicit route to a destination network in its routing table, it uses the default route to forward those packets.

Metric

The metric field contains a configurable value between 1 and 15 (inclusive) which specifies the current metric for the interface. The metric value typically indicates the total number of hops to the destination. The metric value of 16 represents an unreachable destination.

Authentication

RIPv2 authentication uses plaintext password for authentication. If configured using Authentication password, then it is necessary to enter an authentication key value.

The following method is used to authenticate an RIP message:

- If the router is not configured to authenticate RIPv2 messages, then RIPv1 and unauthenticated RIPv2 messages are accepted; authenticated RIPv2 messages are discarded.
- If the router is configured to authenticate RIPv2 messages, then RIPv1 messages and RIPv2 messages which pass authentication testing are accepted; unauthenticated and failed authentication RIPv2 messages are discarded.

For maximum security, RIPv1 messages are ignored when authentication is enabled; otherwise, the routing information from authenticated messages is propagated by RIPv1 routers in an unauthenticated manner.

RIP Configuration Example

The following is an example of RIP configuration.

Note: An interface RIP disabled uses all the default values of the RIP, no matter how the RIP parameters are configured for that interface. RIP sends out RIP regular updates to include an UP interface, but not a DOWN interface.

1. Add VLANs for routing interfaces.

```
>> (config)# vlan 2
>> (config-vlan)# exit
>> (config)# interface port 2
>> (config-if)# switchport mode trunk
>> (config-if)# switchport trunk allowed vlan add 2
>> (config-if)# exit

Port 2 is an UNTAGGED port and its current PVID is 1.
Confirm changing PVID from 1 to 2 [y/n]: y
>> (config)# vlan 3
>> (config-vlan)# exit
>> (config)# interface port 3
>> (config-if)# switchport mode trunk
>> (config-if)# switchport trunk allowed vlan add 3
>> (config-if)# exit
Port 3 is an UNTAGGED port and its current PVID is 1.
Confirm changing PVID from 1 to 3 [y/n]: y
```

2. Add IP interfaces with IPv4 addresses to VLANs.

```
>> # interface ip 2
>> (config-ip-if)# enable
>> (config-ip-if)# ip address 102.1.1.1
>> (config-ip-if)# vlan 2
>> (config-ip-if)# exit
>> # interface ip 3
>> (config-ip-if)# enable
>> (config-ip-if)# ip address 103.1.1.1
>> (config-ip-if)# vlan 3
```

3. Turn on RIP globally and enable RIP for each interface.

```
>> # router rip
>> (config-router-rip)# enable
>> (config-router-rip)# exit
>> # interface ip 2
>> (config-ip-if)# ip rip enable
>> (config-ip-if)# exit
>> # interface ip 3
>> (config-ip-if)# ip rip enable
>> (config-ip-if)# exit
```

Use the following command to check the current valid routes in the routing table of the switch:

```
>> # show ip route
```

For those RIP routes learned within the garbage collection period, that are routes phasing out of the routing table with metric 16, use the following command:

```
>> # show ip rip routes
```

Locally configured static routes do not appear in the RIP Routes table.

Chapter 29. Internet Group Management Protocol

Internet Group Management Protocol (IGMP) is used by IPv4 Multicast routers (M routers) to learn about the existence of host group members on their directly attached subnet. The IPv4 M routers get this information by broadcasting IGMP Membership Queries and listening for IPv4 hosts reporting their host group memberships. This process is used to set up a client/server relationship between an IPv4 multicast source that provides the data streams and the clients that want to receive the data. The switch supports three versions of IGMP:

- IGMPv1: Defines the method for hosts to join a multicast group. However, this version does not define the method for hosts to leave a multicast group. See RFC 1112 for details.
- IGMPv2: Adds the ability for a host to signal its desire to leave a multicast group. See RFC 2236 for details.
- IGMPv3: Adds support for source filtering by which a host can report interest in receiving packets only from specific source addresses, or from all but specific source addresses, sent to a particular multicast address. See RFC 3376 for details.

The G8296 can perform IGMP Snooping, and connect to static M routers. The G8296 can act as a Querier, and participate in the IGMP Querier election process.

The following topics are discussed in this chapter:

- [“IGMP Terms” on page 388](#)
- [“How IGMP Works” on page 389](#)
- [“IGMP Capacity and Default Values” on page 390](#)
- [“IGMP Snooping” on page 391](#)
- [“IGMP Relay” on page 404](#)
- [“Additional IGMP Features” on page 413](#)

IGMP Terms

The following are commonly used IGMP terms:

- Multicast traffic: Flow of data from one source to multiple destinations.
- Group: A multicast stream to which a host can join. Multicast groups have IP addresses in the range: 224.0.1.0 to 239.255.255.255.
- IGMP Querier: A router or switch in the subnet that generates *Membership Queries*.
- IGMP Snooper: A Layer 3 device that forwards multicast traffic only to hosts that are interested in receiving multicast data. This device can be a router or a Layer 3 switch.
- Multicast Router: A router configured to make routing decisions for multicast traffic. The router identifies the type of packet received (unicast or multicast) and forwards the packet to the intended destination.
- IGMP Proxy: A device that filters Join messages and Leave messages sent upstream to the Mrouter to reduce the load on the Mrouter.
- Membership Report: A report sent by the host that indicates an interest in receiving multicast traffic from a multicast group.
- Leave: A message sent by the host when it wants to leave a multicast group.
- FastLeave: A process by which the switch stops forwarding multicast traffic to a port as soon as it receives a Leave message.
- Membership Query: Message sent by the Querier to verify if hosts are listening to a group.
- General Query: A *Membership Query* sent to all hosts. The Group address field for general queries is 0.0.0.0 and the destination address is 224.0.0.1.
- Group-specific Query: A *Membership Query* sent to all hosts in a multicast group.

How IGMP Works

When IGMP is not configured, switches forward multicast traffic through all ports, increasing network load. When IGMPv2 is configured on a switch, multicast traffic flows as follows:

- A server sends multicast traffic to a multicast group.
- The Mrouter sends *Membership Queries* to the switch, which forwards them to all ports in a given VLAN.
- Hosts respond with *Membership Reports* if they want to join a group. The switch forwards these reports to the Mrouter.
- The switch forwards multicast traffic only to hosts that have joined a group.
- The Mrouter periodically sends *Membership Queries* to ensure that a host wants to continue receiving multicast traffic. If a host does not respond, the IGMP Snooper stops sending traffic to the host.
- The host can initiate the Leave process by sending an IGMP Leave packet to the IGMP Snooper.
- When a host sends an IGMPv2 Leave packet, the IGMP Snooper queries to find out if any other host connected to the port is interested in receiving the multicast traffic. If it does not receive a Join message in response, the IGMP Snooper removes the group entry and passes on the information to the Mrouter.

The G8296 supports the following:

- IGMP version 1, 2, and 3
- 128 static Mrouters and 128 dynamic Mrouters

Note: Unknown multicast traffic is sent to all ports if the flood option is enabled and no Membership Report was learned for that specific IGMP group. If the flood option is disabled, unknown multicast traffic is discarded if no hosts or Mrouters are learned on a switch.

To enable or disable IGMP flood, use the following command:

```
RS8296(config)# vlan <vlan ID>  
RS8296(config-vlan)# [no] flood
```

IGMP Capacity and Default Values

The following table lists the maximum and minimum values of the G8296 variables.

Table 38. *G8296 Capacity Table*

Variable	Maximum
IGMP Entries - Snoop	3072
IGMP Entries - Relay	1000
VLANs - Snoop	1024
VLANs - Relay	8
Static Mrouters	128
Dynamic Mrouters	128
Number of IGMP Filters	16
IPMC Groups (IGMP Relay)	1000

The following table lists the default settings for IGMP features and variables.

Table 39. *IGMP Default Configuration Settings*

Field	Default Value
Global IGMP State	Disabled
IGMP Querier	Disabled
IGMP Snooping	Disabled
IGMP Filtering	Disabled
IP Multicast (IPMC) Flood	Enabled
IGMP FastLeave	Disabled for all VLANs
IGMP Mrouter Timeout	255 Seconds
IGMP Report Timeout Variable	10 Seconds
IGMP Query-Interval Variable	125 Seconds
IGMP Robustness Variable	2
IGMPv3	Disabled
IGMPv3 number of sources	8 (The switch processes only the first eight sources listed in the IGMPv3 group record.) Valid range: 1 - 64
IGMPv3 - allow v1v2 Snooping	Enabled

IGMP Snooping

IGMP Snooping allows a switch to listen to the IGMP conversation between hosts and Mrouters. By default, a switch floods multicast traffic to all ports in a broadcast domain. With IGMP Snooping enabled, the switch learns the ports interested in receiving multicast data and forwards it only to those ports. IGMP Snooping conserves network resources.

The switch can sense IGMP *Membership Reports* from attached hosts and acts as a proxy to set up a dedicated path between the requesting host and a local IPv4 Mrouter. After the path is established, the switch blocks the IPv4 multicast stream from flowing through any port that does not connect to a host member, thus conserving bandwidth.

IGMP Querier

For IGMP Snooping to function, you must have an Mrouter on the network that generates IGMP Query packets. Enabling the IGMP Querier feature on the switch allows it to participate in the Querier election process. If the switch is elected as the Querier, it will send IGMP Query packets for the LAN segment.

Querier Election

If multiple Mrouters exist on the network, only one can be configured as a Querier. The Mrouters elect the one with the lowest source IPv4 address or MAC address as the Querier. The Querier performs all periodic membership queries. All other Mrouters (non-Queriers) do not send IGMP Query packets.

Note: When IGMP Querier is enabled on a VLAN, the switch performs the role of an IGMP Querier only if it meets the IGMP Querier election criteria.

Each time the Querier switch sends an IGMP Query packet, it initializes a *general query timer*. If a Querier receives a General Query packet from an Mrouter with a lower IP address or MAC address, it transitions to a non-Querier state and initializes an *other querier present timer*. When this timer expires, the Mrouter transitions back to the Querier state and sends a General Query packet.

Follow this procedure to configure IGMP Querier.

1. Enable IGMP and configure the source IPv4 address for IGMP Querier on a VLAN.

```
RS8296(config)# ip igmp enable
RS8296(config)# ip igmp querier vlan 2 source-ip 10.10.10.1
```

2. Enable IGMP Querier on the VLAN.

```
RS8296(config)# ip igmp querier vlan 2 enable
```

3. Configure the querier election type and define the address.

```
RS8296(config)# ip igmp querier vlan 2 election-type ipv4
```

4. Verify the configuration.

```
RS8296# show ip igmp querier vlan 2

Current IGMP snooping Querier information:
IGMP Querier information for vlan 2:
Other IGMP querier - none
Switch-querier enabled, current state: Querier
Switch-querier type: Ipv4, address 10.10.10.1,
Switch-querier general query interval: 125 secs,
Switch-querier max-response interval: 100 'tenths of secs',
Switch-querier startup interval: 31 secs, count: 2
Switch-querier robustness: 2
IGMP configured version is v3
IGMP Operating version is v3
```

IGMP Groups

One IGMP entry is allocated for each unique join request, based on the VLAN and IGMP group address. If multiple ports join the same IGMP group using the same VLAN, only a single IGMP entry is used.

IGMPv3 Snooping

IGMPv3 includes new Membership Report messages that extend IGMP functionality. The switch provides snooping capability for all types of IGMPv3 *Membership Reports*.

IGMPv3 supports Source-Specific Multicast (SSM). SSM identifies session traffic by both source and group addresses.

The IGMPv3 implementation keeps records on the multicast hosts present in the network. If a host is already registered, when it receives a new *IS_INC*, *TO_INC*, *IS_EXC*, or *TO_EXC* report from same host, the switch makes the correct transition to new (port-host-group) registration based on the IGMPv3 RFC. The registrations of other hosts for the same group on the same port are not changed.

The G8296 supports the following IGMPv3 filter modes:

- **INCLUDE** mode: The host requests membership to a multicast group and provides a list of IPv4 addresses from which it wants to receive traffic.
- **EXCLUDE** mode: The host requests membership to a multicast group and provides a list of IPv4 addresses from which it does not want to receive traffic. This indicates that the host wants to receive traffic only from sources that are not part of the Exclude list. To disable snooping on EXCLUDE mode reports, use the following command:

```
RS8296(config)# no ip igmp snoop igmpv3 exclude
```

By default, the G8296 snoops the first eight sources listed in the IGMPv3 Group Record. Use the following command to change the number of snooping sources:

```
RS8296(config)# ip igmp snoop igmpv3 sources <1-64>
```


IGMPv3 Snooping is compatible with IGMPv1 and IGMPv2 Snooping. To disable snooping on version 1 and version 2 reports, use the following command:

```
RS8296(config)# no ip igmp snoop igmpv3 v1v2
```

IGMP Snooping Configuration Guidelines

Consider the following guidelines when you configure IGMP Snooping:

- IGMP operation is independent of the routing method. You can use RIP, OSPF, or static routes for Layer 3 routing.
- When multicast traffic flood is disabled, the multicast traffic sent by the multicast server is discarded if no hosts or Mrouters are learned on the switch.
- The Mrouter periodically sends IGMP Queries.
- The switch learns the Mrouter on the port connected to the router when it sees Query messages. The switch then floods the IGMP queries on all other ports including a Trunk Group, if any.
- Multicast hosts send IGMP Reports as a reply to the IGMP Queries sent by the Mrouter.
- The switch can also learn an Mrouter when it receives a PIM hello packet from another device. However, an Mrouter learned from a PIM packet has a lower priority than an Mrouter learned from an IGMP Query. A switch overwrites an Mrouter learned from a PIM packet when it receives an IGMP Query on the same port.
- A host sends an IGMP Leave message to its multicast group. The expiration timer for this group is updated to 10 seconds. The Layer 3 switch sends IGMP Group-Specific Query to the host that had sent the Leave message. If the host does not respond with an IGMP Report during these 10 seconds, all the groups expire and the switch deletes the host from the IGMP groups table. The switch then proxies the IGMP Leave messages to the Mrouter.

IGMP Snooping Configuration Example

This section provides steps to configure IGMP Snooping on the G8296.

1. Configure port and VLAN membership on the switch.
2. Add VLANs to IGMP Snooping.

```
RS8296(config)# ip igmp snoop vlan 1
```

3. Enable IGMP Snooping.

```
RS8296(config)# ip igmp snoop enable
```

4. Enable IGMPv3 Snooping (optional).

```
RS8296(config)# ip igmp snoop igmpv3 enable
```

5. Enable the IGMP feature.

```
RS8296(config)# ip igmp enable
```

6. View dynamic IGMP information.

```
RS8296# show ip igmp groups

Total entries: 5 Total IGMP groups: 3
Note: The <Total IGMP groups> number is computed as
      the number of unique (Group, Vlan) entries!
Note: Local groups (224.0.0.x) are not snooped/relayed and will not
      appear.
Source      Group      VLAN      Port      Version    Mode      Expires   Fwd
-----
10.1.1.1    232.1.1.1  2         4         V3         INC       4:16     Yes
10.1.1.5    232.1.1.1  2         4         V3         INC       4:16     Yes
*           232.1.1.1  2         4         V3         INC       -        No
10.10.10.43 235.0.0.1  9         1         V3         INC       2:26     Yes
*           236.0.0.1  9         1         V3         EXC       -        Yes

RS8296# show ip igmp mrouter

Total entries: 3 Total number of dynamic mroouters: 2
Total number of installed static mroouters: 1

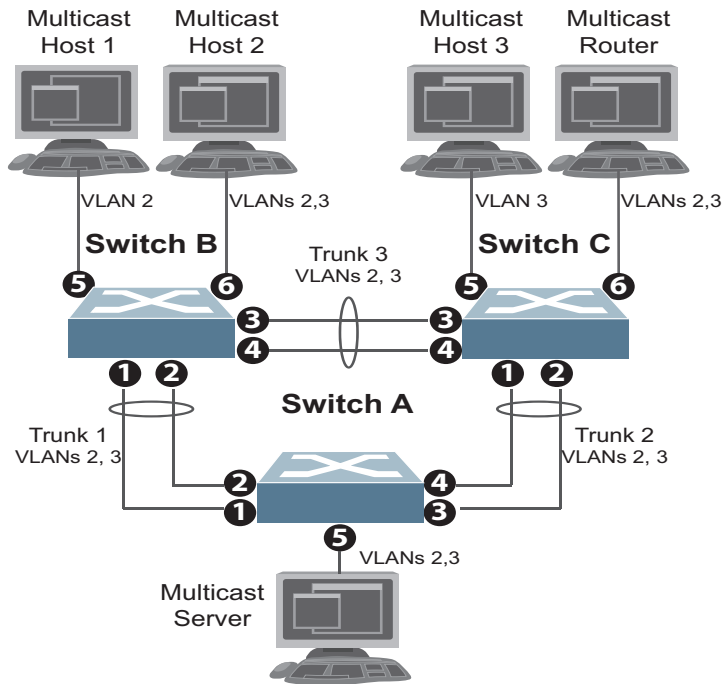
SrcIP      VLAN      Port      Version    Expires    MRT      QRV      QQIC
-----
10.1.1.1   2         21        V3         4:09      128     2        125
10.1.1.5   2         23        V2         4:09      125     -        -
*          9         24        V2         static    -        -        -
```

These commands display information about IGMP Groups and Mroouters learned by the switch.

Advanced Configuration Example: IGMP Snooping

Figure 32 shows an example topology. Switches B and C are configured with IGMP Snooping.

Figure 32. Topology



Devices in this topology are configured as follows:

- STG2 includes VLAN2; STG3 includes VLAN3.
- The multicast server sends IP multicast traffic for the following groups:
 - VLAN 2, 225.10.0.11 – 225.10.0.12, Source: 22.10.0.11
 - VLAN 2, 225.10.0.13 – 225.10.0.15, Source: 22.10.0.13
 - VLAN 3, 230.0.2.1 – 230.0.2.2, Source: 22.10.0.1
 - VLAN 3, 230.0.2.3 – 230.0.2.5, Source: 22.10.0.3
- The Mrouter sends IGMP Query packets in VLAN 2 and VLAN 3. The Mrouter's IP address is 10.10.10.10.
- The multicast hosts send the following IGMP Reports:
 - IGMPv2 Report, VLAN 2, Group: 225.10.0.11, Source: *
 - IGMPv2 Report, VLAN 3, Group: 230.0.2.1, Source: *
 - IGMPv3 IS_INCLUDE Report, VLAN 2, Group: 225.10.0.13, Source: 22.10.0.13
 - IGMPv3 IS_INCLUDE Report, VLAN 3, Group: 230.0.2.3, Source: 22.10.0.3

- The hosts receive multicast traffic as follows:
 - Host 1 receives multicast traffic for groups (*, 225.10.0.11), (22.10.0.13, 225.10.0.13)
 - Host 2 receives multicast traffic for groups (*, 225.10.0.11), (*, 230.0.2.1), (22.10.0.13, 225.10.0.13), (22.10.0.3, 230.0.2.3)
 - Host 3 receives multicast traffic for groups (*, 230.0.2.1), (22.10.0.3, 230.0.2.3)
- The Mrouter receives all the multicast traffic.

Prerequisites

Before you configure IGMP Snooping, ensure you have performed the following actions:

- Configured VLANs.
- Enabled IGMP.
- Added VLANs to IGMP Snooping.
- Configured a switch or Mrouter as the Querier.
- Identified the IGMP version(s) you want to enable.
- Disabled IGMP flooding.

Configuration

This section provides the configuration details of the switches shown in [Figure 32](#).

Switch A Configuration

1. Configure VLANs and tagging.

```
RS8296(config)# interface port 1-5
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 2,3
RS8296(config-if)# switchport trunk allowed vlan remove 1
RS8296(config-if)# exit
```

2. Configure an IP interface with IPv4 address, and assign a VLAN.

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip address 10.10.10.1 enable
RS8296(config-ip-if)# vlan 2
RS8296(config-ip-if)# exit
```

3. Assign a bridge priority lower than the default bridge priority to enable the switch to become the STP root in STG 2 and 3.

```
RS8296(config)# spanning-tree stp 2 bridge priority 4096
RS8296(config)# spanning-tree stp 3 bridge priority 4096
```

4. Configure LACP dynamic trunk groups (portchannels).

```
RS8296(config)# interface port 1
RS8296(config-if)# lacp key 100
RS8296(config-if)# lacp mode active
RS8296(config-if)# exit

RS8296(config)# interface port 2
RS8296(config-if)# lacp key 100
RS8296(config-if)# lacp mode active
RS8296(config-if)# exit

RS8296(config)# interface port 3
RS8296(config-if)# lacp key 200
RS8296(config-if)# lacp mode active
RS8296(config-if)# exit

RS8296(config)# interface port 4
RS8296(config-if)# lacp key 200
RS8296(config-if)# lacp mode active
```

Switch B Configuration

1. Configure VLANs and tagging.

```
RS8296(config)# interface port 1-4,6
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# exit

RS8296(config)# interface port 1-6
RS8296(config-if)# switchport trunk allowed vlan add 2
RS8296(config-if)# exit

RS8296(config)# interface port 1-4,6
RS8296(config-if)# switchport trunk allowed vlan add 3
RS8296(config-if)# exit

RS8296(config)# interface port 1-5
RS8296(config-if)# switchport trunk allowed vlan remove 1
RS8296(config-if)# exit
```

2. Configure an IP interface with IPv4 address, and assign a VLAN.

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip address 10.10.10.2 enable
RS8296(config-ip-if)# vlan 2
RS8296(config-ip-if)# exit
```

3. Configure STP. Reset the ports to make the edge configuration operational.

```
RS8296(config)# interface port 5,6
RS8296(config-if)# spanning-tree portfast
RS8296(config-if)# shutdown
RS8296(config-if)# no shutdown
RS8296(config-if)# exit
```

4. Configure an LACP dynamic trunk group (portchannel).

```
RS8296(config)# interface port 1,2
RS8296(config-if)# lacp key 300
RS8296(config-if)# lacp mode active
RS8296(config-if)# exit
```

5. Configure a static trunk group (portchannel).

```
RS8296(config)# portchannel 1 port 3,4 enable
```

6. Configure IGMP Snooping.

```
RS8296(config)# ip igmp enable
RS8296(config)# ip igmp snoop vlan 2,3
RS8296(config)# ip igmp snoop source-ip 10.10.10.2
RS8296(config)# ip igmp snoop igmpv3 enable
RS8296(config)# ip igmp snoop igmpv3 sources 64
RS8296(config)# ip igmp snoop enable

RS8296(config)# vlan 2
RS8296(config-vlan)# no flood
RS8296(config-vlan)# exit

RS8296(config)# vlan 3
RS8296(config-vlan)# no flood
RS8296(config-vlan)# exit
```

Switch C Configuration

1. Configure VLANs and tagging.

```
RS8296(config)# interface port 1-4,6
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# exit

RS8296(config)# interface port 1-4,6
RS8296(config-if)# switchport trunk allowed vlan add 2
RS8296(config-if)# exit

RS8296(config)# interface port 1-6
RS8296(config-if)# switchport trunk allowed vlan add 3
RS8296(config-if)# exit

RS8296(config)# interface 1-6
RS8296(config-if)# switchport trunk allowed vlan remove 1
RS8296(config-if)# exit
```

2. Configure an IP interface with IPv4 address, and assign a VLAN.

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip address 10.10.10.3 enable
RS8296(config-ip-if)# vlan 2
RS8296(config-ip-if)# exit
```

3. Configure STP. Reset the ports to make the edge configuration operational.

```
RS8296(config)# interface port 5,6  
RS8296(config-if)# spanning-tree portfast  
RS8296(config-if)# shutdown  
RS8296(config-if)# no shutdown  
RS8296(config-if)# exit
```

4. Configure an LACP dynamic trunk group (portchannel).

```
RS8296(config)# interface port 1,2  
RS8296(config-if)# lACP key 400  
RS8296(config-if)# lACP mode active  
RS8296(config-if)# exit
```

5. Configure a static trunk group (portchannel).

```
RS8296(config)# portchannel 1 port 3,4 enable
```

6. Configure IGMP Snooping.

```
RS8296(config)# ip igmp enable  
RS8296(config)# ip igmp snoop vlan 2,3  
RS8296(config)# ip igmp snoop source-ip 10.10.10.3  
RS8296(config)# ip igmp snoop igmpv3 enable  
RS8296(config)# ip igmp snoop igmpv3 sources 64  
RS8296(config)# ip igmp snoop enable  
  
RS8296(config)# vlan 2  
RS8296(config-vlan)# no flood  
RS8296(config-vlan)# exit  
  
RS8296(config)# vlan 3  
RS8296(config-vlan)# no flood  
RS8296(config-vlan)# exit
```


Troubleshooting

This section provides the steps to resolve common IGMP Snooping configuration issues. The topology described in [Figure 32](#) is used as an example.

Multicast traffic from non-member groups reaches the host or Mrouter

- Check if traffic is unregistered. For unregistered traffic, an IGMP entry is not displayed in the IGMP groups table.

```
RS8296# show ip igmp groups
```

- Ensure IPMC flooding is disabled and CPU is enabled.

```
RS8296(config)# vlan <vlan id>  
RS8296(config-vlan)# no flood  
RS8296(config-vlan)# cpu
```

- Check the egress port's VLAN membership. The ports to which the hosts and Mrouter are connected must be used only for VLAN 2 and VLAN 3.

```
RS8296# show vlan
```

Note: To avoid such a scenario, disable IPMC flooding for all VLANs enabled on the switches (if this is an acceptable configuration).

- Check IGMP Reports on switches B and C for information about the IGMP groups.

```
RS8296# show ip igmp groups
```

If the non-member IGMP groups are displayed in the table, close the application that may be sending the IGMP Reports for these groups.

Identify the traffic source by using a sniffer on the hosts and reading the source IP/MAC address. If the source IP/MAC address is unknown, check the port statistics to find the ingress port.

```
RS8296# show interface port <port id> interface-counters
```

- Ensure no static multicast MACs, static multicast groups, or static Mrouters are configured.
- Ensure IGMP Relay and PIM are not configured.

Not all multicast traffic reaches the appropriate receivers.

- Ensure hosts are sending IGMP Reports for all the groups. Check the VLAN on which the groups are learned.

```
RS8296# show ip igmp groups
```

If some of the groups are not displayed, ensure the multicast application is running on the host device and the generated IGMP Reports are correct.

- Ensure multicast traffic reaches the switch to which the host is connected. Close the application sending the IGMP Reports. Clear the IGMP groups by flapping (disabling, then re-enabling) the port.

Note: To clear all IGMP groups, use the following command:

```
RS8296(config)# clear ip igmp groups
```

However, this will clear all the IGMP groups and will influence other hosts.

- Check if the multicast traffic reaches the switch.

```
RS8296# show ip igmp ipmcgrp
```

If the multicast traffic group is not displayed in the table, check the link state, VLAN membership, and STP convergence.

- Ensure multicast server is sending all the multicast traffic.
- Ensure no static multicast MACs, static multicast groups, or static multicast routes are configured.

IGMP queries sent by the Mrouter do not reach the host.

- Ensure the Mrouter is learned on switches B and C.

```
RS8296# show ip igmp mrouter
```

If it is not learned on switch B but is learned on switch C, check the link state of the trunk group, VLAN membership, and STP convergence.

If it is not learned on any switch, ensure the multicast application is running and is sending correct IGMP Query packets.

If it is learned on both switches, check the link state, VLAN membership, and STP port states for the ports connected to the hosts.

IGMP Reports/Leaves sent by the hosts do not reach the Mrouter

- Ensure IGMP Queries sent by the Mrouter reach the hosts.
- Ensure the Mrouter is learned on both switches. Note that the Mrouter may not be learned on switch B immediately after a trunk group failover/failback.

```
RS8296# show ip igmp mrouter
```

- Ensure the host's multicast application is started and is sending correct IGMP Reports/Leaves.

```
RS8296# show ip igmp groups
RS8296# show ip igmp counters
```

A host receives multicast traffic from the incorrect VLAN

- Check port VLAN membership.
- Check IGMP Reports sent by the host.
- Check multicast data sent by the server.

The Mrouter is learned on the incorrect trunk group

- Check link state. Trunk group 1 might be down or in STP discarding state.
- Check STP convergence.
- Check port VLAN membership.

Hosts receive multicast traffic at a lower rate than normal

Note: This behavior is expected if IPMC flood is disabled and CPU is enabled. As soon as the IGMP/IPMC entries are installed on ASIC, the IPMC traffic recovers and is forwarded at line rate. This applies to unregistered IPMC traffic.

- Ensure a storm control is not configured on the trunks.

```
RS8296(config)# interface port <port id>  
RS8296(config-if)# no storm-control multicast
```

- Check link speeds and network congestion.

IGMP Relay

The G8296 can act as an IGMP Relay (or IGMP Proxy) device that relays IGMP multicast messages and traffic between an Mrouter and end stations. IGMP Relay allows the G8296 to participate in network multicasts with no configuration of the various multicast routing protocols, so you can deploy it in the network with minimal effort.

To an IGMP host connected to the G8296, IGMP Relay appears to be an IGMP Mrouter. IGMP Relay sends *Membership Queries* to hosts, which respond by sending an IGMP response message. A host can also send an unsolicited Join message to the IGMP Relay.

To an Mrouter, IGMP Relay appears as a host. The Mrouter sends IGMP host queries to IGMP Relay, and IGMP Relay responds by forwarding IGMP host reports and unsolicited Join messages from its attached hosts.

IGMP Relay also forwards multicast traffic between the Mrouter and end stations, similar to IGMP Snooping.

You can configure up to two Mrouters to use with IGMP Relay. One Mrouter acts as the primary Mrouter, and one is the backup Mrouter. The G8296 uses health checks to select the primary Mrouter.

Configuration Guidelines

Consider the following guidelines when you configure IGMP Relay:

- IGMP Relay and IGMP Snooping are mutually exclusive—if you enable IGMP Relay, you must turn off IGMP Snooping/Querier.
- Add the upstream Mrouter VLAN to the IGMP Relay list, using the following command:

```
RS8296(config)# ip igmp relay vlan <VLAN number>
```

- If IGMP hosts reside on different VLANs, you must:
 - o Disable IGMP flooding.

```
RS8296(config)# vlan <vlan id>  
RS8296(config-vlan)# no flood
```

- o Ensure CPU forwarding is enabled; Ensure that multicast data is forwarded across the VLANs.

```
RS8296(config)# vlan <vlan id>  
RS8296(config-vlan)# cpu
```

Configure IGMP Relay

Use the following procedure to configure IGMP Relay.

1. Configure IP interfaces with IPv4 addresses, and assign VLANs.

```
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip address 10.10.1.1
RS8296(config-ip-if)# ip netmask 255.255.255.0
RS8296(config-ip-if)# vlan 2
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 3
RS8296(config-ip-if)# ip address 10.10.2.1
RS8296(config-ip-if)# ip netmask 255.255.255.0
RS8296(config-ip-if)# vlan 3
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
```

2. Turn IGMP on.

```
RS8296(config)# ip igmp enable
```

3. Configure the upstream Mrouters with IPv4 addresses.

```
RS8296(config)# ip igmp relay mrouter 1 address 100.0.1.2
RS8296(config)# ip igmp relay mrouter 1 enable
RS8296(config)# ip igmp relay mrouter 2 address 100.0.2.4
RS8296(config)# ip igmp relay mrouter 2 enable
```

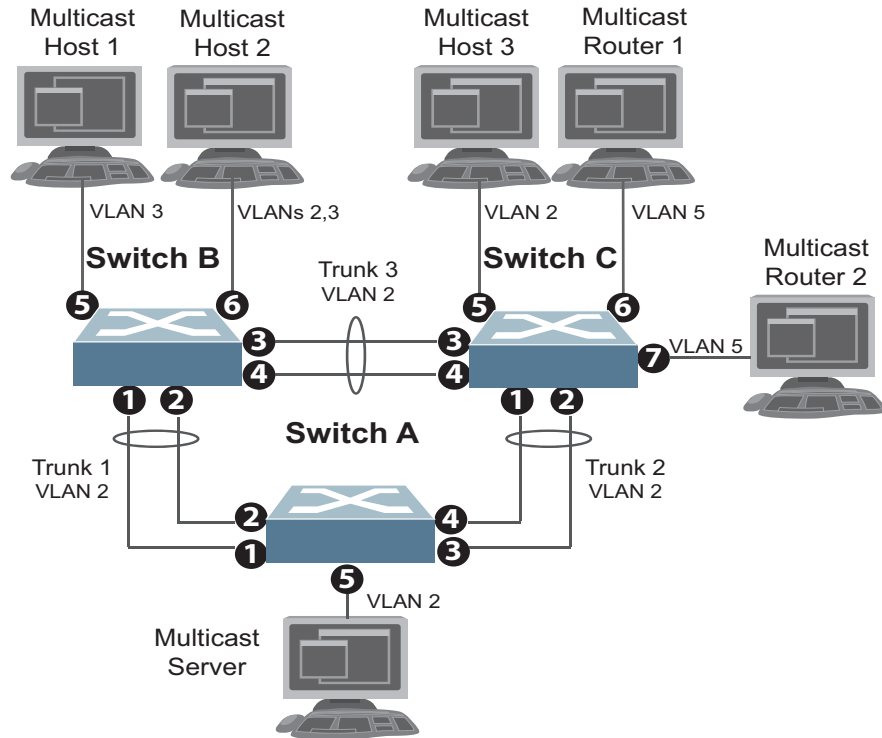
4. Enable IGMP Relay and add VLANs to the downstream network.

```
RS8296(config)# ip igmp relay vlan 2
RS8296(config)# ip igmp relay vlan 3
RS8296(config)# ip igmp relay enable
```

Advanced Configuration Example: IGMP Relay

Figure 33 shows an example topology. Switches B and C are configured with IGMP Relay.

Figure 33. Topology



Devices in this topology are configured as follows:

- The IP address of Multicast Router 1 is 5.5.5.5
- The IP address of Multicast Router 2 is 5.5.5.6
- STG 2 includes VLAN2; STG 3 includes VLAN 3; STG 5 includes VLAN 5.
- The multicast server sends IP multicast traffic for the following groups:
 - VLAN 2, 225.10.0.11 – 225.10.0.15
- The multicast hosts send the following IGMP Reports:
 - Host 1: 225.10.0.11 – 225.10.0.12, VLAN 3
 - Host 2: 225.10.0.12 – 225.10.0.13, VLAN 2; 225.10.0.14 – 225.10.0.15, VLAN 3
 - Host 3: 225.10.0.13 – 225.10.0.14, VLAN 2
- The Mrouter receives all the multicast traffic.

Prerequisites

Before you configure IGMP Relay, ensure you have performed the following actions:

- Configured VLANs.

- Enabled IGMP.
- Disabled IGMP flooding.
- Disabled IGMP Snooping.

Configuration

This section provides the configuration details of the switches in [Figure 33](#).

Switch A Configuration

1. Configure a VLAN.

```
RS8296(config)# interface port 1-5
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 2
RS8296(config-if)# exit
```

2. Configure an IP interface with IPv4 address, and assign a VLAN..

```
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip address 2.2.2.10 enable
RS8296(config-ip-if)# vlan 2
RS8296(config-ip-if)# exit
```

3. Assign a bridge priority lower than the default bridge priority to enable the switch to become the STP root in STG 2 and 3.

```
RS8296(config)# spanning-tree stp 2 bridge priority 4096
```

4. Configure LACP dynamic trunk groups (portchannels).

```
RS8296(config)# interface port 1,2
RS8296(config-if)# lacp key 100
RS8296(config-if)# lacp mode active
RS8296(config-if)# exit

RS8296(config)# interface port 3,4
RS8296(config-if)# lacp key 200
RS8296(config-if)# lacp mode active
```

Switch B Configuration

1. Configure VLANs and tagging.

```
RS8296(config)# interface port 1-4,6
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 2
RS8296(config-if)# exit

RS8296(config)# interface port 5,6
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 3
RS8296(config-if)# exit
```

2. Configure IP interfaces with IPv4 addresses, and assign VLANs.

```
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip address 2.2.2.20 enable
RS8296(config-ip-if)# vlan 2
RS8296(config-ip-if)# exit

RS8296(config)# interface ip 3
RS8296(config-ip-if)# ip address 3.3.3.20 enable
RS8296(config-ip-if)# vlan 3
RS8296(config-ip-if)# exit

RS8296(config)# ip gateway 2 address 2.2.2.30 enable
```

3. Configure STP.

```
RS8296(config)# interface port 5,6
RS8296(config-if)# spanning-tree portfast
RS8296(config-if)# shutdown
RS8296(config-if)# no shutdown
RS8296(config-if)# exit
```

4. Configure an LACP dynamic trunk group (portchannel).

```
RS8296(config)# interface port 1,2
RS8296(config-if)# lacp key 300
RS8296(config-if)# lacp mode active
RS8296(config-if)# exit
```

5. Configure a static trunk group (portchannel).

```
RS8296(config)# portchannel 1 port 3,4 enable
```


6. Configure IGMP Relay.

```
RS8296(config)# ip igmp enable
RS8296(config)# ip igmp relay vlan 2,3
RS8296(config)# ip igmp relay mrouter 1 address 5.5.5.5 enable
RS8296(config)# ip igmp relay mrouter 2 address 5.5.5.6 enable
RS8296(config)# ip igmp relay enable

RS8296(config)# vlan 2
RS8296(config-vlan)# no flood
RS8296(config-vlan)# exit

RS8296(config)# vlan 3
RS8296(config-vlan)# no flood
RS8296(config-vlan)# exit
```

Switch C Configuration

1. Configure VLANs.

```
RS8296(config)# interface port 1-5
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 2
RS8296(config-if)# exit

RS8296(config)# interface port 6,7
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 5
RS8296(config-if)# exit
```

2. Configure IP interfaces with IPv4 addresses and assign VLANs.

```
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip address 2.2.2.30 enable
RS8296(config-ip-if)# vlan 2
RS8296(config-ip-if)# exit

RS8296(config)# interface ip 3
RS8296(config-ip-if)# ip address 5.5.5.30 enable
RS8296(config-ip-if)# vlan 5
RS8296(config-ip-if)# exit

RS8296(config)# ip gateway 2 address 2.2.2.20 enable
```

3. Configure STP.

```
RS8296(config)# interface port 5,6,7
RS8296(config-if)# spanning-tree portfast
RS8296(config-if)# shutdown
RS8296(config-if)# no shutdown
RS8296(config-if)# exit
```

4. Configure LACP dynamic trunk group (portchannel).

```
RS8296(config)# interface port 1,2
RS8296(config-if)# lacp key 400
RS8296(config-if)# lacp mode active
RS8296(config-if)# exit
```

5. Configure a static trunk group (portchannel).

```
RS8296(config)# portchannel 1 port 3,4 enable
```

6. Enable IGMP.

```
RS8296(config)# ip igmp enable
```

7. Configure IGMP Relay.

```
RS8296(config)# ip igmp relay vlan 2,5
RS8296(config)# ip igmp relay mrouter 1 address 5.5.5.5 enable
RS8296(config)# ip igmp relay mrouter 2 address 5.5.5.6 enable
RS8296(config)# ip igmp relay enable

RS8296(config)# vlan 2
RS8296(config-vlan)# no flood
RS8296(config-vlan)# exit

RS8296(config)# vlan 5
RS8296(config-vlan)# no flood
RS8296(config-vlan)# exit
```

Troubleshooting

This section provides the steps to resolve common IGMP Relay configuration issues. The topology described in [Figure 33](#) is used as an example.

Multicast traffic from non-member groups reaches the hosts or the Mrouter

- Ensure IPMC flood is disabled.

```
RS8296(config)# vlan <vlan id>
RS8296(config-vlan)# no flood
```

- Check the egress port's VLAN membership. The ports to which the hosts and Mrouter are connected must be used only for VLAN 2, VLAN 3, or VLAN 5.

```
RS8296(config)# show vlan
```

Note: To avoid such a scenario, disable IPMC flooding for all VLANs enabled on the switches (if this is an acceptable configuration).

- Check IGMP Reports on switches B and C for information about IGMP groups.

```
RS8296(config)# show ip igmp groups
```

If non-member IGMP groups are displayed in the table, close the application that may be sending the IGMP Reports for these groups.

Identify the traffic source by using a sniffer on the hosts and reading the source IP address/MAC address. If the source IP address/MAC address is unknown, check the port statistics to find the ingress port.

```
RS8296(config)# show interface port <port id> interface-counters
```

- Ensure no static multicast MACs and static Mrouters are configured.

Not all multicast traffic reaches the appropriate receivers

- Ensure hosts are sending IGMP Reports for all the groups. Check the VLAN on which the groups are learned.

```
RS8296(config)# show ip igmp groups
```

If some of the groups are not displayed, ensure the multicast application is running on the host device and the generated IGMP Reports are correct.

- Ensure the multicast traffic reaches the switch to which the host is connected. Close the application sending the IGMP Reports. Clear the IGMP groups by flapping (disabling, then re-enabling) the port.

Note: To clear all IGMP groups, use the following command:

```
RS8296(config)# clear ip igmp groups
```

However, this will clear all the IGMP groups and will influence other hosts.

- Check if the multicast traffic reaches the switch.

```
RS8296(config)# show ip igmp ipmcgrp
```

If the multicast traffic group is not displayed in the table, check the link state, VLAN membership, and STP convergence.

- Ensure the multicast server is sending all the multicast traffic.
- Ensure no static multicast MACs or static multicast routes are configured.
- Ensure PIM is not enabled on the switches.

IGMP Reports/Leaves sent by the hosts do not reach the Mrouter

- Ensure that the primary or secondary Mrouter is reachable. If not, the IGMP Reports/Leaves are not forwarded. Note that the Mrouter may not be reachable immediately after a trunk group failover/failback.

```
RS8296(config)# show ip igmp groups
```

- Ensure the host's multicast application is started and is sending correct IGMP Reports/Leaves.

```
RS8296(config)# show ip igmp groups
RS8296(config)# show ip igmp counters
```

The Mrouter is reachable on the incorrect trunk group

- Check link state. Trunk group 1 may be down or in STP discarding state.
- Check STP convergence and port VLAN membership.
- Check IP connectivity between the switch and the configured Mrouter (primary or secondary).

Hosts receive multicast traffic at a lower rate than normal

- Ensure a multicast threshold is not configured on the trunk groups.

```
RS8296(config)# interface port <port id>
RS8296(config-if)# no storm-control multicast
```

- Check link speeds and network congestion.

Additional IGMP Features

The following topics are discussed in this section:

- [“FastLeave” on page 413](#)
- [“IGMP Filtering” on page 413](#)
- [“Static Multicast Router” on page 414](#)

FastLeave

In normal IGMP operation, when the switch receives an IGMPv2 Leave message, it sends a Group-Specific Query to determine if any other devices in the same group (and on the same port) are still interested in the specified multicast group traffic. The switch removes the affiliated port from that particular group, if the switch does not receive an IGMP Membership Report within the query-response-interval.

With FastLeave enabled on the VLAN, a port can be removed immediately from the port list of the group entry when the IGMP Leave message is received.

Note: Only IGMPv2 supports FastLeave. Enable FastLeave on ports that have only one host connected. If more than one host is connected to a port, you may lose some hosts unexpectedly.

Use the following command to enable FastLeave.

```
RS8296(config)# ip igmp fastleave <VLAN number>
```

IGMP Filtering

With IGMP filtering, you can allow or deny certain IGMP groups to be learned on a port.

If access to a multicast group is denied, *IGMP Membership Reports* from the port are dropped, and the port is not allowed to receive IPv4 multicast traffic from that group. If access to the multicast group is allowed, Membership Reports from the port are forwarded for normal processing.

To configure IGMP filtering, you must globally enable IGMP filtering, define an IGMP filter, assign the filter to a port, and enable IGMP filtering on the port. To define an IGMP filter, you must configure a range of IPv4 multicast groups, choose whether the filter will allow or deny multicast traffic for groups within the range, and enable the filter.

Configuring the Range

Each IGMP filter allows you to set a start and end point that defines the range of IPv4 addresses upon which the filter takes action. Each IPv4 address in the range must be between 224.0.0.0 and 239.255.255.255.

Configuring the Action

Each IGMP filter can allow or deny IPv4 multicasts to the range of IPv4 addresses configured. If you configure the filter to deny IPv4 multicasts, then *IGMP Membership Reports* from multicast groups within the range are dropped. You can

configure a secondary filter to allow IPv4 multicasts to a small range of addresses within a larger range that a primary filter is configured to deny. The two filters work together to allow IPv4 multicasts to a small subset of addresses within the larger range of addresses.

Note: Lower-numbered filters take precedence over higher-number filters. For example, the action defined for IGMP filter 1 supersedes the action defined for IGMP filter 2.

Configure IGMP Filtering

1. Enable IGMP filtering on the switch.

```
RS8296(config)# ip igmp filtering
```

2. Define an IGMP filter with IPv4 information.

```
RS8296(config)# ip igmp profile 1 range 224.0.0.0 226.0.0.0
RS8296(config)# ip igmp profile 1 action deny
RS8296(config)# ip igmp profile 1 enable
```

3. Assign the IGMP filter to a port.

```
RS8296(config)# interface port 3
RS8296(config-if)# ip igmp profile 1
RS8296(config-if)# ip igmp filtering
```

Static Multicast Router

A static Mrouter can be configured for a particular port on a particular VLAN. A static Mrouter does not have to be learned through IGMP Snooping. Any data port can accept a static Mrouter.

When you configure a static Mrouter on a VLAN, it replaces any dynamic Mrouters learned through IGMP Snooping.

Configure a Static Multicast Router

1. For each Mrouter, configure a port, VLAN, and IGMP version of the multicast router.

```
RS8296(config)# ip igmp mrouter 5 1 2
```

2. Verify the configuration.

```
RS8296(config)# show ip igmp mrouter
```

Chapter 305. Multicast Listener Discovery

Multicast Listener Discovery (MLD) is an IPv6 protocol that a host uses to request multicast data for a multicast group. An IPv6 router uses MLD to discover the presence of multicast listeners (nodes that want to receive multicast packets) on its directly attached links, and to discover specifically the multicast addresses that are of interest to those neighboring nodes.

MLD version 1 is derived from Internet Group Management Protocol version 2 (IGMPv2) and MLDv2 is derived from IGMPv3. MLD uses ICMPv6 (IP Protocol 58) message types. See RFC 2710 and RFC 3810 for details.

MLDv2 protocol, when compared to MLDv1, adds support for source filtering—the ability for a node to report interest in listening to packets only from specific source addresses, or from all but specific source addresses, sent to a particular multicast address. MLDv2 is interoperable with MLDv1. See RFC 3569 for details on Source-Specific Multicast (SSM).

The following topics are discussed in this chapter:

- [“MLD Terms” on page 416](#)
- [“How MLD Works” on page 417](#)
- [“MLD Capacity and Default Values” on page 420](#)
- [“Configuring MLD” on page 421](#)

MLD Terms

Following are the commonly used MLD terms:

- Multicast traffic: Flow of data from one source to multiple destinations.
- Group: A multicast stream to which a host can join.
- Multicast Router (Mrouter): A router configured to make routing decisions for multicast traffic. The router identifies the type of packet received (unicast or multicast) and forwards the packet to the intended destination.
- Querier: An Mrouter that sends periodic query messages. Only one Mrouter on the subnet can be elected as the Querier.
- Multicast Listener Query: Messages sent by the Querier. There are three types of queries:
 - General Query: Sent periodically to learn multicast address listeners from an attached link. G8296 uses these queries to build and refresh the Multicast Address Listener state. General Queries are sent to the link-scope all-nodes multicast address (FF02::1), with a multicast address field of 0, and a maximum response delay of *query response interval*.
 - Multicast Address Specific Query: Sent to learn if a specific multicast address has any listeners on an attached link. The multicast address field is set to the IPv6 multicast address.
 - Multicast Address and Source Specific Query: Sent to learn if, for a specified multicast address, there are nodes still listening to a specific set of sources. Supported only in MLDv2.

Note: Multicast Address Specific Queries and Multicast Address and Source Specific Queries are sent only in response to State Change Reports, and never in response to Current State Reports.

- Multicast Listener Report: Sent by a host when it joins a multicast group, or in response to a Multicast Listener Query sent by the Querier. Hosts use these reports to indicate their current multicast listening state, or changes in the multicast listening state of their interfaces. These reports are of two types:
 - Current State Report: Contains the current Multicast Address Listening State of the host.
 - State Change Report: If the listening state of a host changes, the host immediately reports these changes through a State Change Report message. These reports contain either Filter Mode Change records and/or Source List Change records. State Change Reports are retransmitted several times to ensure all Mrouters receive it.
- Multicast Listener Done: Sent by a host when it wants to leave a multicast group. This message is sent to the link-scope all-routers IPv6 destination address of FF02::2. When an Mrouter receives a Multicast Listener Done message from the last member of the multicast address on a link, it stops forwarding traffic to this multicast address.

How MLD Works

The software uses the information obtained through MLD to maintain a list of multicast group memberships for each interface and forwards the multicast traffic only to interested listeners.

Without MLD, the switch forwards IPv6 multicast traffic through all ports, increasing network load. Following is an overview of operations when MLD is configured on the G8296:

- The switch acts as an Mrouter when MLDv1/v2 is configured and enabled on each of its directly attached links. If the switch has multiple interfaces connected to the same link, it operates the protocol on any one of the interfaces.
- If there are multiple Mrouters on the subnet, the Mrouter with the numerically lowest IPv6 address is elected as the Querier.
- The Querier sends general queries at short intervals to learn multicast address listener information from an attached link.
- Hosts respond to these queries by reporting their per-interface Multicast Address Listening state, through Current State Report messages sent to a specific multicast address that all MLD routers on the link listen to.
- If the listening state of a host changes, the host immediately reports these changes through a State Change Report message.
- The Querier sends a Multicast Address Specific Query to verify if hosts are listening to a specified multicast address or not. Similarly, if MLDv2 is configured, the Querier sends a Multicast Address and Source Specific Query to verify, for a specified multicast address, if hosts are listening to a specific set of sources, or not. MLDv2 listener report messages consists of Multicast Address Records:
 - INCLUDE: to receive packets from source specified in the MLDv2 message
 - EXCLUDE: to receive packets from all sources except the ones specified in the MLDv2 message
- A host can send a State Change Report to indicate its desire to stop listening to a particular multicast address (or source in MLDv2). The Querier then sends a multicast address specific query to verify if there are other listeners of the multicast address. If there aren't any, the Mrouter deletes the multicast address from its Multicast Address Listener state and stops sending multicast traffic. Similarly in MLDv2, the Mrouter sends a Multicast Address and Source Specific Query to verify if, for a specified multicast address, there are hosts still listening to a specific set of sources.

G8296 supports MLD versions 1 and 2.

Note: MLDv2 operates in version 1 compatibility mode when, in a specific network, not all hosts are configured with MLDv2.

How Flooding Impacts MLD

When `flood` option is disabled, the unknown multicast traffic is discarded if no Mrouters are learned on the switch. You can set the flooding behavior by configuring the `flood` and `cpu` options. You can optimize the flooding to ensure that unknown IP multicast (IPMC) data packets are not dropped during the learning phase.

The flooding options include:

- `flood`: Enable hardware flooding in VLAN for the unregistered IPMC; This option is enabled by default.
- `cpu`: Enable sending unregistered IPMC to the Mrouter ports. However, during the learning period, there will be some packet loss. The `cpu` option is enabled by default. You must ensure that the `flood` and `optflood` options are disabled.
- `optflood`: Enable optimized flooding to allow sending the unregistered IPMC to the Mrouter ports without having any packet loss during the learning period; This option is disabled by default; When `optflood` is enabled, the `flood` and `cpu` settings are ignored.

The flooding parameters must be configured per VLAN. Enter the following command to set the `flood` or `cpu` option:

```
RS8296(config)# vlan <vlan number>
RS8296(config-vlan)# [no] flood
RS8296(config-vlan)# [no] cpu
RS8296(config-vlan)# [no] optflood
```

MLD Querier

An Mrouter acts as a Querier and periodically (at short query intervals) sends query messages in the subnet. If there are multiple Mrouters in the subnet, only one can be the Querier. All Mrouters on the subnet listen to the messages sent by the multicast address listeners, and maintain the same multicast listening information state.

All MLDv2 queries are sent with the FE80::/64 link-local source address prefix.

Querier Election

Only one Mrouter can be the Querier per subnet. All other Mrouters will be non-Queriers. MLD versions 1 and 2 elect the Mrouter with the numerically lowest IPv6 address as the Querier.

If the switch is configured as an Mrouter on a subnet, it also acts as a Querier by default and sends multiple general queries. If the switch receives a general query from another Querier with a numerically lower IPv6 address, it sets the *other querier present timer* to the *other querier present timeout*, and changes its state to non-Querier. When the *other querier present timer* expires, it regains the Querier state and starts sending general queries.

Note: When MLD Querier is enabled on a VLAN, the switch performs the role of an MLD Querier only if it meets the MLD Querier election criteria.

Dynamic Mrouters

The switch learns Mrouters on the ingress VLANs of the MLD-enabled interface. All report or done messages are forwarded to these Mrouters. By default, the option of dynamically learning Mrouters is disabled. To enable it, use the following command:

```
RS8296(config)# interface ip <interface number>  
RS8296(config-ip-if)# ipv6 mld dmrtr enable
```

MLD Capacity and Default Values

Table 40 lists the maximum and minimum values of the G8296 variables.

Table 40. *G8296 Capacity Table*

Variable	Maximum Value
IPv6 Multicast Entries	256
IPv6 Interfaces for MLD	8

Note: IGMP and MLD share the IPMC table. When the IPMC table is full, you cannot allocate additional IGMP/MLD groups.

Table 41 lists the default settings for MLD features and variables.

Table 41. *MLD Timers and Default Values*

Field	Default Value
Robustness Variable (RV)	2
Query Interval (QI)	125 seconds
Query Response Interval (QRI)	10 seconds
Multicast Address Listeners Interval (MALI)	260 seconds [derived: $RV * QI + QRI$]
Other Querier Present Interval [OQPT]	255 seconds [derived: $RV * QI + \frac{1}{2} QRI$]
Start up Query Interval [SQI]	31.25 seconds [derived: $\frac{1}{4} * QI$]
Startup Query Count [SQC]	2 [derived: RV]
Last Listener Query Interval [LLQI]	1 second
Last Listener Query Count [LLQC]	2 [derived: RV]
Last Listener Query Time [LLQT]	2 seconds [derived: $LLQI * LLQT$]
Older Version Querier Present Timeout: [OVQPT]	260 seconds [derived: $RV * QI + QRI$]
Older Version Host Present Interval [OVHPT]	260 seconds [derived: $RV * QI + QRI$]

Configuring MLD

Following are the steps to enable MLD and configure the interface parameters:

1. Turn on MLD globally.

```
RS8296(config)# ipv6 mld  
RS8296(config-router-mld)# enable  
RS8296(config-router-mld)# exit
```

2. Create an IPv6 interface.

```
RS8296(config)# interface ip 2  
RS8296(config-ip-if)# enable  
RS8296(config-ip-if)# ipv6 address 2002:1:0:0:0:0:3  
RS8296(config-ip-if)# ipv6 prefixlen 64
```

3. Enable MLD on the IPv6 interface.

```
RS8296(config-ip-if)# ipv6 mld enable
```

4. Configure the MLD parameters on the interface: version, robustness, query response interval, MLD query interval, and last listener query interval.

```
RS8296(config-ip-if)# ipv6 mld version <1-2>(MLD version)  
RS8296(config-ip-if)# ipv6 mld robust <1-10>(Robustness)  
RS8296(config-ip-if)# ipv6 mld qri <1-256>(In seconds)  
RS8296(config-ip-if)# ipv6 mld qinterval <1-608>(In seconds)  
RS8296(config-ip-if)# ipv6 mld llistnr <1-32>(In seconds)
```

Chapter 31. Border Gateway Protocol

Border Gateway Protocol (BGP) is an Internet protocol that enables routers on an IPv4 network to share and advertise routing information with each other about the segments of the IPv4 address space they can access within their network and with routers on external networks. BGP allows you to decide what is the “best” route for a packet to take from your network to a destination on another network rather than simply setting a default route from your border router(s) to your upstream provider(s). BGP is defined in RFC 1771.

RackSwitch G8296s can advertise their IP interfaces and IPv4 addresses using BGP and take BGP feeds from as many as 96 BGP router peers. This allows more resilience and flexibility in balancing traffic from the Internet.

Note: Lenovo Network Operating System 8.2 does not support IPv6 for BGP.

The following topics are discussed in this section:

- [“Internal Routing Versus External Routing” on page 424](#)
- [“Forming BGP Peer Routers” on page 428](#)
- [“Loopback Interfaces” on page 430](#)
- [“What is a Route Map?” on page 430](#)
- [“Aggregating Routes” on page 434](#)
- [“Redistributing Routes” on page 434](#)
- [“BGP Attributes” on page 436](#)
- [“Selecting Route Paths in BGP” on page 438](#)
- [“BGP Failover Configuration” on page 439](#)
- [“Default Redistribution and Route Aggregation Example” on page 441](#)

Internal Routing Versus External Routing

To ensure effective processing of network traffic, every router on your network needs to know how to send a packet (directly or indirectly) to any other location/destination in your network. This is referred to as *internal routing* and can be done with static routes or using active, internal dynamic routing protocols, such as RIP, RIPv2, and OSPF.

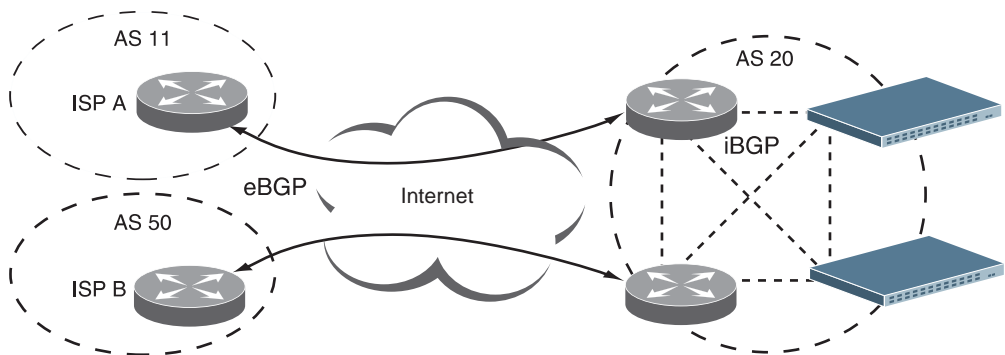
Static routes must have a higher degree of precedence than dynamic routing protocols. If the destination route is not in the route cache, the packets are forwarded to the default gateway which may be incorrect if a dynamic routing protocol is enabled.

It is also useful to tell routers outside your network (upstream providers or *peers*) about the routes you can access in your network. External networks (those outside your own) that are under the same administrative control are referred to as *autonomous systems (AS)*. Sharing of routing information between autonomous systems is known as *external routing*.

External BGP (eBGP) is used to exchange routes between different autonomous systems whereas internal BGP (iBGP) is used to exchange routes within the same autonomous system. An iBGP is a type of internal routing protocol you can use to do active routing inside your network. It also carries AS path information, which is important when you are an ISP or doing BGP transit.

The iBGP peers have to maintain reciprocal sessions to every other iBGP router in the same AS (in a full-mesh manner) to propagate route information throughout the AS. If the iBGP session shown between the two routers in AS 20 was not present (as indicated in [Figure 34](#)), the top router would not learn the route to AS 50, and the bottom router would not learn the route to AS 11, even though the two AS 20 routers are connected via the RackSwitch G8296.

Figure 34. iBGP and eBGP



When there are many iBGP peers, having a full-mesh configuration results in large number of sessions between the iBGP peers. In such situations, configuring a route reflector eliminates the full-mesh configuration requirement, prevents route propagation loops, and provides better scalability to the peers. For details, see [“Route Reflector” on page 425](#).

Typically, an AS has one or more *border routers*—peer routers that exchange routes with other ASs—and an internal routing scheme that enables routers in that AS to reach every other router and destination within that AS. When you *advertise* routes to border routers on other autonomous systems, you are effectively committing to carry data to the IPv4 space represented in the route being advertised. For example, if you advertise 192.204.4.0/24, you are declaring that if another router sends you data destined for any address in 192.204.4.0/24, you know how to carry that data to its destination.

Route Reflector

The Networking OS implementation conforms to the BGP Route Reflection specification defined in RFC 4456.

As per RFC 1771 specification, a route received from an iBGP peer cannot be advertised to another iBGP peer. This makes it mandatory to have full-mesh iBGP sessions between all BGP routers within an AS. A route reflector—a BGP router—breaks this iBGP loop avoidance rule. It does not affect the eBGP behavior. A route reflector is a BGP speaker that advertises a route learnt from an iBGP peer to another iBGP peer. The advertised route is called the reflected route.

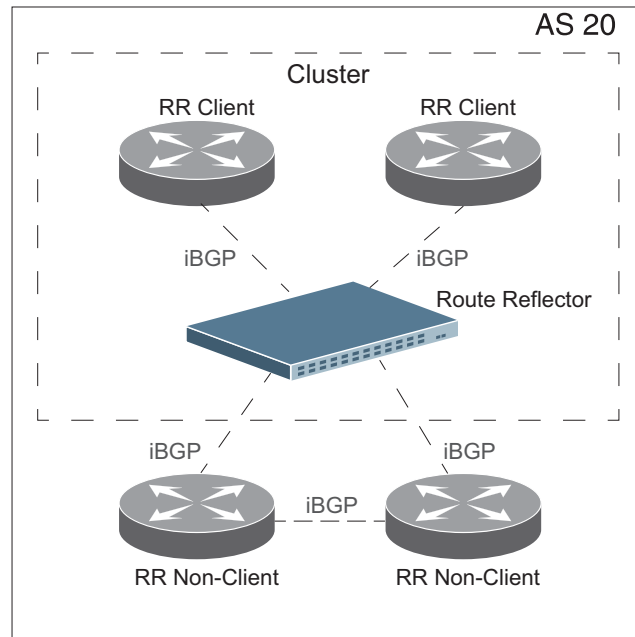
A route reflector has two groups of internal peers: clients and non-clients. A route reflector reflects between these groups and among the clients. The non-client peers must be fully meshed. The route reflector and its clients form a cluster.

When a route reflector receives a route from an iBGP peer, it selects the best path based on its path selection rule. It then does the following based on the type of peer it received the best path from:

- A route received from a non-client iBGP peer is reflected to all clients.
- A route received from an iBGP client peer is reflected to all iBGP clients and iBGP non-clients.

In [Figure 35](#), the G8296 is configured as a route reflector. All clients and non-clients are in the same AS.

Figure 35. iBGP Route Reflector



The following attributes are used by the route reflector functionality:

- **ORIGINATOR ID:** BGP identifier (BGP router ID) of the route originator in the local AS. If the route does not have the ORIGINATOR ID attribute (it has not been reflected before), the router ID of the iBGP peer from which the route has been received is copied into the Originator ID attribute. This attribute is never modified by subsequent route reflectors. A router that identifies its own ID as the ORIGINATOR ID, it ignores the route.
- **CLUSTER LIST:** Sequence of the CLUSTER ID (i.e. router ID) values representing the reflection path that the route has passed. The value configured with the `RS8296(config-router-bgp)# cluster-id <ID>` command (or the router ID of the route reflector if the cluster-id is not configured) is prepended to the Cluster list attribute. If a route reflector detects its own CLUSTER ID in the CLUSTER LIST, it ignores the route. Up to 10 CLUSTER IDs can be added to a CLUSTER LIST.

Route reflection functionality can be configured as follows:

1. Configure an AS.

```
RS8296(config)# router bgp  
RS8296(config-router-bgp)# as 22  
RS8296(config-router-bgp)# enable
```

2. Configure a route reflector client.

```
RS8296(config-router-bgp)# neighbor 2 remote-address 10.1.50.1  
RS8296(config-router-bgp)# neighbor 2 remote-as 22  
RS8296(config-router-bgp)# neighbor 2 route-reflector-client  
RS8296(config-router-bgp)# no neighbor 2 shutdown
```

Note: When a client is configured on the G8296, the switch automatically gets configured as a route reflector.

3. Verify configuration.

```
RS8296(config)# show ip bgp neighbor 2 information

BGP Peer 2 Information:
 2: 10.1.50.1          , version 0, TTL 255, TTL Security hops 0
Remote AS: 0, Local AS: 22, Link type: IBGP
Remote router ID: 0.0.0.0, Local router ID: 9.9.9.9
next-hop-self disabled
RR client enabled
BGP status: connect, Old status: connect
Total received packets: 0, Total sent packets: 0
Received updates: 0, Sent updates: 0
Keepalive: 0, Holdtime: 0, MinAdvTime: 60
LastErrorCode: unknown(0), LastErrorSubcode: unspecified(0)
Established state transitions: 0
```

Once configured as a route reflector, the switch, by default, passes routes between clients. If required, you can disable this by using the following command:

```
RS8296(config-router-bgp)# no client-to-client reflection
```

You can view the route reflector BGP attributes attached to a BGP route using the following command:

```
RS8296(config-router-bgp)# show ip bgp information 5.0.0.0 255.255.255.0
BGP routing table entry for 5.0.0.0/255.255.255.0
Paths: (1 available, best #1)
Multipath: eBGP
Local
 30.1.1.1 (metric 0) from 22.22.1.1(17.17.17.17)
Origin: IGP, localpref 0, valid, internal, best
Originator: 1.16.0.195
Cluster list: 17.17.17.17
```

Restrictions

Consider the following restrictions when configuring route reflection functionality:

- When a CLUSTER ID is changed, all iBGP sessions are restarted.
- When a route reflector client is enabled/disabled, the session is restarted.

Forming BGP Peer Routers

Two BGP routers become peers or neighbors once you establish a TCP connection between them. You can configure BGP peers statically or dynamically. While it may be desirable to configure static peers for security reasons, dynamic peers prove to be useful in cases where the remote address of the peer is unknown. For example in B-RAS applications, where subscriber interfaces are dynamically created and the address is assigned dynamically from a local pool or by using RADIUS.

For each new route, if a peer is interested in that route (for example, if a peer would like to receive your static routes and the new route is static), an update message is sent to that peer containing the new route. For each route removed from the route table, if the route has already been sent to a peer, an update message containing the route to withdraw is sent to that peer.

For each Internet host, you must be able to send a packet to that host, and that host has to have a path back to you. This means that whoever provides Internet connectivity to that host must have a path to you. Ultimately, this means that they must “hear a route” which covers the section of the IPv4 space you are using; otherwise, you will not have connectivity to the host in question.

Static Peers

You can configure BGP static peers by using the following commands:

```
RS8296(config)# router bgp
RS8296(config-router-bgp)# neighbor <1-96> remote-address <IP address>
RS8296(config-router-bgp)# neighbor <1-96> remote-as <1-65535>
RS8296(config-router-bgp)# no neighbor <1-96> shutdown
```

Static peers always take precedence over dynamic peers. Consider the following:

- If the remote address of an incoming BGP connection matches both a static peer address and an IP address from a dynamic group, the peer is configured statically and not dynamically.
- If a new static peer is enabled while a dynamic peer for the same remote address exists, BGP automatically removes the dynamic peer.
- If a new static peer is enabled when the maximum number of BGP peers were already configured, then BGP deletes the dynamic peer that was last created and adds the newly created static peer. A syslog will be generated for the peer that was deleted.

Dynamic Peers

To configure dynamic peers, you must define a range of IP addresses for a group. BGP waits to receive an open message initiated from BGP speakers within that range. Dynamic peers are automatically created when a peer group member accepts the incoming BGP connection. Dynamic peers are passive. When they are not in the established state, they accept inbound connections but do not initiate outbound connections.

You can configure up to 6 AS numbers per group. When the BGP speaker receives an open message from a dynamic peer, the AS number from the packet must match one of the remote AS numbers configured on the corresponding group.

When you delete a remote AS number, all dynamic peers established from that remote AS will be deleted.

You can define attributes for the dynamic peers only at the group level. You cannot configure attributes for any one dynamic peer. All static peer attributes, except the BGP passive mode, can also be configured for groups.

To set the maximum number of dynamic peers for a group that can simultaneously be in an established state, enter the following command:

```
RS8296(config-router-bgp)# neighbor group <1-8> listen limit <1-96>
```

If you reset this limit to a lower number, and if the dynamic peers already established for the group are higher than this new limit, then BGP deletes the last created dynamic peer(s) until the new limit is reached.

Note: The maximum number of static and dynamic peers established simultaneously cannot exceed the maximum peers, i.e. 96, that the switch can support. If the maximum peers are established, no more dynamic peers will be enabled even if the maximum dynamic peers limit you had configured for the groups was not reached.

Given below are the basic commands for configuring dynamic peers:

```
RS8296(config)# router bgp
RS8296(config-router-bgp)# neighbor group <1-8> listen range
<IP address> <subnet mask> (Define
IP address range)
RS8296(config-router-bgp)# neighbor group <1-8> remote-as <1-65535>
alternate-as <1-65535> (Enter up to 5
alternate AS numbers)
RS8296(config-router-bgp)# no neighbor group <1-8> shutdown
```

Removing Dynamic Peers

You cannot remove dynamic peers manually. However, you can stop a dynamic peer using the following command:

```
RS8296(config)# router bgp stop <neighbor number>
```

The stop command interrupts the BGP connection until the peer tries to re-establish the connection.

Also, when a dynamic peer state changes from established to idle, BGP removes the dynamic peer.

Loopback Interfaces

In many networks, multiple connections may exist between network devices. In such environments, it may be useful to employ a loopback interface for a common BGP router address, rather than peering the switch to each individual interface.

Note: To ensure that the loopback interface is reachable from peer devices, it must be advertised using an interior routing protocol (such as OSPF), or a static route must be configured on the peer.

To configure an existing loopback interface for BGP neighbor, use the following commands:

```
RS8296(config)# router bgp
RS8296(config-router-bgp)# neighbor <#> update-source loopback <1-5>
RS8296(config-router-bgp)# exit
```

What is a Route Map?

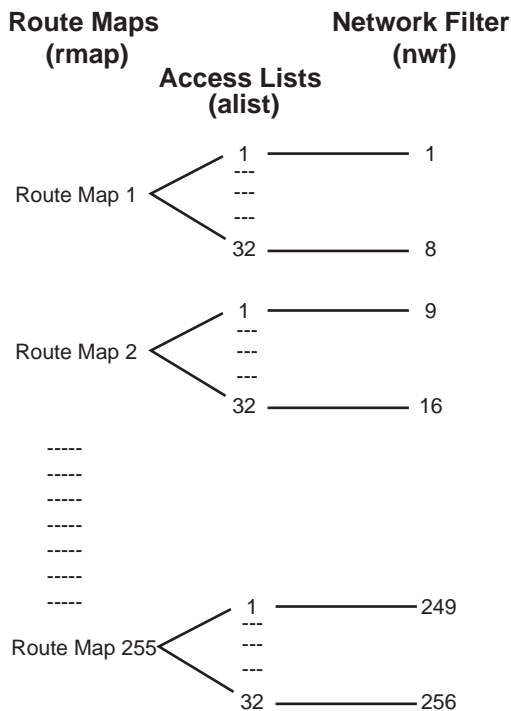
A route map is used to control and modify routing information. Route maps define conditions for redistributing routes from one routing protocol to another or controlling routing information when injecting it in and out of BGP. For example, a route map is used to set a preference value for a specific route from a peer router and another preference value for all other routes learned via the same peer router. For example, the following command is used to enter the Route Map mode for defining a route map:

```
RS8296(config)# route-map <map number> (Select a route map)
RS8296(config-route-map)# ? (List available commands)
```

A route map allows you to match attributes, such as metric, network address, and AS number. It also allows users to overwrite the local preference metric and to append the AS number in the AS route. See [“BGP Failover Configuration” on page 439](#).

Networking OS allows you to configure 255 route maps. Each route map can have up to 32 access lists. Each access list consists of a network filter. A network filter defines an IPv4 address and subnet mask of the network that you want to include in the filter. [Figure 36](#) illustrates the relationship between route maps, access lists, and network filters.

Figure 36. Distributing Network Filters in Access Lists and Route Maps



Next Hop Peer IP Address

Next hop peer IP address can be configured only for route maps used in BGP. When a route map is applied on ingress, the next hop of learnt routes is replaced with peer IP address. When applied on egress, the next hop of the redistributed routes is replaced with the local IP address.

```
RS8296(config)# route-map <map number>
RS8296(config-router-map)# set ip next-hop <peer IP address>
```

Incoming and Outgoing Route Maps

You can have two types of route maps: incoming and outgoing. A BGP peer router can be configured to support up to eight route maps in the incoming route map list and outgoing route map list.

If a route map is not configured in the incoming route map list, the router imports all BGP updates. If a route map is configured in the incoming route map list, the router ignores all unmatched incoming updates. If you set the action to **deny**, you must add another route map to permit all unmatched updates.

Route maps in an outgoing route map list behave similar to route maps in an incoming route map list. If a route map is not configured in the outgoing route map list, all routes are advertised or permitted. If a route map in the outgoing route map list is set to **permit**, matched routes are advertised and unmatched routes are ignored.

Precedence

You can set a priority to a route map by specifying a precedence value with the following command (Route Map mode):

```
RS8296(config)# route-map <map number> (Select a route map)
RS8296(config-route-map)# precedence <1-255>(Specify a precedence)
RS8296(config-route-map)# exit
```

The smaller the value the higher the precedence. If two route maps have the same precedence value, the smaller number has higher precedence.

Configuration Overview

To configure route maps, you need to do the following:

1. Define a network filter.

```
RS8296(config)# ip match-address 1 <IPv4 address> <IPv4 subnet mask>
RS8296(config)# ip match-address 1 enable
```

Enter a filter number from 1 to 256. Specify the IPv4 address and subnet mask of the network that you want to match. Enable the network filter. You can distribute up to 256 network filters among 64 route maps each containing 32 access lists.

2. (Optional) Define the criteria for the access list and enable it.

Specify the access list and associate the network filter number configured in Step 1.

```
RS8296(config)# route-map 1
RS8296(config-route-map)# access-list 1 match-address 1
RS8296(config-route-map)# access-list 1 metric <metric value>
RS8296(config-route-map)# access-list 1 action deny
RS8296(config-route-map)# access-list 1 enable
```

Steps 2 and 3 are optional, depending on the criteria that you want to match. In Step 2, the network filter number is used to match the subnets defined in the network filter. In Step 3, the autonomous system number is used to match the subnets. Or, you can use both (Step 2 and Step 3) criteria: access list (network filter) and access path (AS filter) to configure the route maps.

3. (Optional) Configure the AS filter attributes.

```
RS8296(config-route-map)# as-path-list 1 as 1
RS8296(config-route-map)# as-path-list 1 action deny
RS8296(config-route-map)# as-path-list 1 enable
```

4. (Optional) Set up the BGP attributes.

If you want to overwrite the attributes that the peer router is sending, define the following BGP attributes:

- Specify up to three AS numbers that you want to prepend to a matched route and the local preference for the matched route.

- Specify the metric [Multi Exit Discriminator (MED)] for the matched route.

```
RS8296(config-route-map)# as-path-preference <AS number> [<AS number>]
                                     [<AS number>]
RS8296(config-route-map)# local-preference <local preference value>
RS8296(config-route-map)# metric <metric value>
```

5. Enable the route map.

```
RS8296(config-route-map)# enable
RS8296(config-route-map)# exit
```

6. Turn BGP on.

```
RS8296(config)# router bgp
RS8296(config-router-bgp)# enable
```

7. Assign the route map to a peer router.

Select the peer router and then add the route map to the incoming route map list,

```
RS8296(config-router-bgp)# neighbor 1 route-map in <1-255>
```

or to the outgoing route map list.

```
RS8296(config-router-bgp)# neighbor 1 route-map out <1-255>
```

8. Exit Router BGP mode.

```
RS8296(config-router-bgp)# exit
```

Aggregating Routes

Aggregation is the process of combining several different routes in such a way that a single route can be advertised, which minimizes the size of the routing table. You can configure aggregate routes in BGP either by redistributing an aggregate route into BGP or by creating an aggregate entry in the BGP routing table.

To define an aggregate route in the BGP routing table, use the following commands:

```
>> # router bgp
>> (config-router-bgp)# aggregate-address <1-16> <IPv4 address> <mask>
>> (config-router-bgp)# aggregate-address <1-16> enable
```

The origin code for the aggregated route is either IGP or INCOMPLETE.

An example of creating a BGP aggregate route is shown in [“Default Redistribution and Route Aggregation Example” on page 441](#).

Redistributing Routes

In addition to running multiple routing protocols simultaneously, N/OS software can redistribute information from one routing protocol to another. For example, you can instruct the switch to use BGP to re-advertise static routes. This applies to all of the IP-based routing protocols.

You can also conditionally control the redistribution of routes between routing domains by defining a method known as route maps between the two domains. For more information on route maps, see [“What is a Route Map?” on page 430](#). Redistributing routes is another way of providing policy control over whether to export OSPF routes, fixed routes, and static routes. For an example configuration, see [“Default Redistribution and Route Aggregation Example” on page 441](#).

Default routes can be configured using the following methods:

- Import
- Originate—The router sends a default route to peers if it does not have any default routes in its routing table.
- Redistribute—Default routes are either configured through the default gateway or learned via other protocols and redistributed to peer routers. If the default routes are from the default gateway, enable the static routes because default routes from the default gateway are static routes. Similarly, if the routes are learned from another routing protocol, make sure you enable that protocol for redistribution.
- None

The origin code for the redistributed routes is INCOMPLETE.

BGP Communities

BGP communities are attribute tags that allow controlled distribution of routing information based on an agreement between BGP peers. Communities are commonly used by transit service providers to enable peering customers to choose specific routing destinations for their outgoing routes. The transit service provider would typically publish a list of well-known or proprietary communities along with their descriptions, and take it upon itself to advertise incoming routes accordingly. For instance, an ISP may advertise that incoming routes tagged with community XY:01 will be advertised only to European peers while incoming routes tagged with community XY:02 will be advertised only to Asian peers.

The RackSwitch G8296 can be configured to manage the community tags applied to the outgoing route updates. It does not, however, modify any routing decisions based on the community tags.

Up to 32 community tags can be applied to prefixes that pass a route-map. Valid values are between 0:0 and 65535:65535. Newly added communities will replace the existing communities. To set communities to prefixes that pass the route-map, use the following commands:

```
RS8296(config)# route-map <map number>  
RS8296(config-route-map)# set community {aa:nn [aa:nn]}
```

To remove all community tags from prefixes that pass the route-map, use the following command:

```
RS8296(config-route-map)# set community none
```

To propagate prefixes that pass the route-map with their original community settings, use the following command:

```
RS8296(config-route-map)# no set community
```

To enable or disable community tags forwarding for specific neighbors or neighbor groups, use the following commands:

```
RS8296(config)# router bgp  
RS8296(config-router-bgp)# neighbor 5 send-community  
RS8296(config-router-bgp)# no neighbor 6 send-community  
RS8296(config-router-bgp)# neighbor group 1 send-community
```

BGP Attributes

The following BGP attributes are discussed in this section: Local preference, metric (Multi-Exit Discriminator), and Next hop.

Local Preference Attribute

When there are multiple paths to the same destination, the local preference attribute indicates the preferred path. The path with the higher preference is preferred (the default value of the local preference attribute is 100). Unlike the weight attribute, which is only relevant to the local router, the local preference attribute is part of the routing update and is exchanged among routers in the same AS.

The local preference attribute can be set in one of two ways:

- The following commands use the BGP default local preference method, affecting the outbound direction only.

```
>> # router bgp
>> (config_router_bgp)# local-preference <0-4294967294>
>> (config_router_bgp)# exit
```

- The following commands use the route map local preference method, which affects both inbound and outbound directions.

```
>> # route-map 1
>> (config_route_map)# local-preference
>> (config_router_map)# exit
```

Metric (Multi-Exit Discriminator) Attribute

This attribute is a hint to external neighbors about the preferred path into an AS when there are multiple entry points. A lower metric value is preferred over a higher metric value. The default value of the metric attribute is 0.

Unlike local preference, the metric attribute is exchanged between ASs; however, a metric attribute that comes into an AS does not leave the AS.

When an update enters the AS with a certain metric value, that value is used for decision making within the AS. When BGP sends that update to another AS, the metric is reset to 0.

Unless otherwise specified, the router compares metric attributes for paths from external neighbors that are in the same AS.

Next Hop Attribute

BGP routing updates sent to a neighbor contain the next hop IP address used to reach a destination. In eBGP, the edge router, by default, sends its own IP address as the next hop address. However, this can sometimes cause routing path failures in Non-Broadcast Multiaccess Networks (NBMA) and when the edge router sends iBGP updates.

To avoid routing failures, you can manually configure the next hop IP address. In case of NBMA networks, you can configure the external BGP speaker to advertise its own IP address as the next hop. In case of iBGP updates, you can configure the edge iBGP router to send its IP address as the next hop.

Next hop can be configured on a BGP peer or a peer group. Use the following commands:

- Next Hop for a BGP Peer

```
RS8296(config)# router bgp  
RS8296(config-router-bgp)# neighbor <number> next-hop-self
```

- Next Hop for a BGP Peer Group:

```
RS8296(config)# router bgp  
RS8296(config-router-bgp)# neighbor group <number> next-hop-self
```

Selecting Route Paths in BGP

BGP selects only one path as the best path. It does not rely on metric attributes to determine the best path. When the same network is learned via more than one BGP peer, BGP uses its policy for selecting the best route to that network. The BGP implementation on the G8296 uses the following criteria to select a path when the same route is received from multiple peers.

1. Local fixed and static routes are preferred over learned routes.
2. With iBGP peers, routes with higher local preference values are selected.
3. In the case of multiple routes of equal preference, the route with lower AS path weight is selected.

AS path weight = 128 x AS path length (number of autonomous systems traversed).

4. In the case of equal weight and routes learned from peers that reside in the same AS, the lower metric is selected.

Note: A route with a metric is preferred over a route without a metric.

5. In the case of equal cost, the eBGP route is preferred over iBGP.
6. The lower cost to the next hop of routes is selected.
7. If all routes have same route type (eBGP or iBGP), the route with the lower router ID is selected.
8. The first learned route is selected if the same router ID is present.

When the path is selected, BGP puts the selected path in its routing table and propagates the path to its neighbors.

Equal Cost Multi-Path

BGP can be configured to load balance the traffic over multiple paths if first six steps in path selection are identical and the next-hop of the route differs.

Multipath Relax

BGP multipath relax functionality allows load balancing across different autonomous system paths that have equal AS path length. This functionality can be enabled using the command:

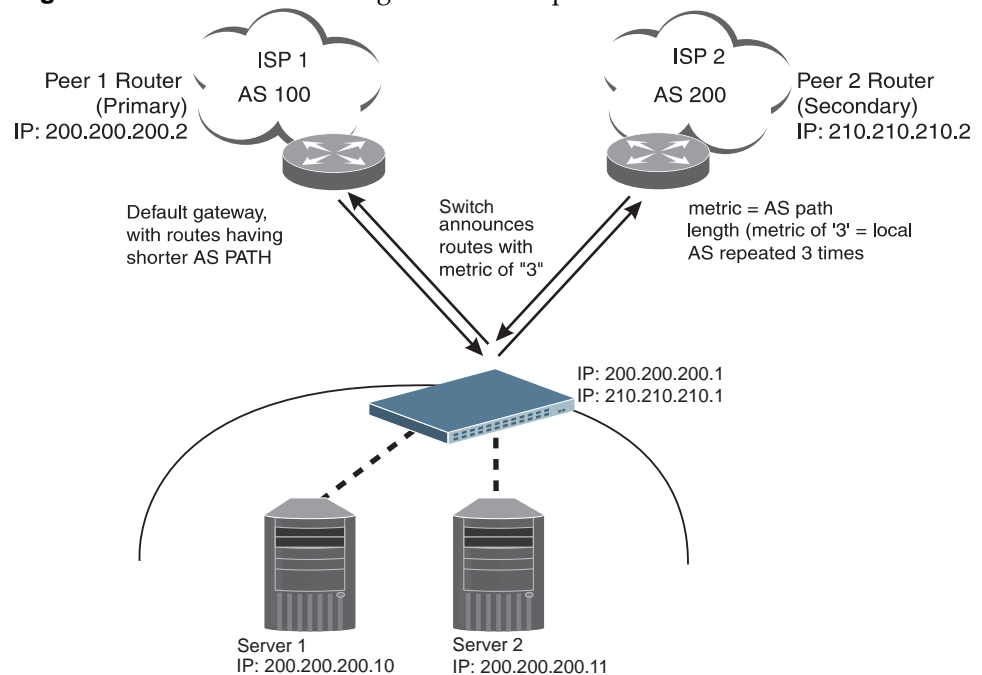
```
RS8296(config-router-bgp)# bestpath as-path multipath-relax
```

BGP Failover Configuration

Use the following example to create redundant default gateways for a G8296 at a Web Host/ISP site, eliminating the possibility, if one gateway goes down, that requests will be forwarded to an upstream router unknown to the switch.

As shown in [Figure 37](#), the switch is connected to ISP 1 and ISP 2. The customer negotiates with both ISPs to allow the switch to use their peer routers as default gateways. The ISP peer routers will then need to announce themselves as default gateways to the G8296.

Figure 37. BGP Failover Configuration Example



On the G8296, one peer router (the secondary one) is configured with a longer AS path than the other, so that the peer with the shorter AS path will be seen by the switch as the primary default gateway. ISP 2, the secondary peer, is configured with a metric of "3," thereby appearing to the switch to be three router *hops* away.

1. Define the VLANs.

For simplicity, both default gateways are configured in the same VLAN in this example. The gateways could be in the same VLAN or different VLANs.

```
RS8296(config)# vlan 1
```

2. Define the IP interfaces with IPv4 addresses.

The switch will need an IP interface for each default gateway to which it will be connected. Each interface must be placed in the appropriate VLAN. These interfaces will be used as the primary and secondary default gateways for the switch.

```
>> # interface ip 1
>> (config-ip-if)# ip address 200.200.200.1
>> (config-ip-if)# ip netmask 255.255.255.0
>> (config-ip-if)# enable
>> (config-ip-if)# exit
>> # interface ip 2
>> (config-ip-if)# ip address 210.210.210.1
>> (config-ip-if)# ip netmask 255.255.255.0
>> (config-ip-if)# enable
>> (config-ip-if)# exit
```

3. Enable IP forwarding.

IP forwarding is turned on by default and is used for VLAN-to-VLAN (non-BGP) routing. Make sure IP forwarding is on if the default gateways are on different subnets or if the switch is connected to different subnets and those subnets need to communicate through the switch (which they almost always do).

```
>> # ip routing
```

Note: To help eliminate the possibility for a Denial of Service (DoS) attack, the forwarding of directed broadcasts is disabled by default.

4. Configure BGP peer router 1 and 2 with IPv4 addresses.

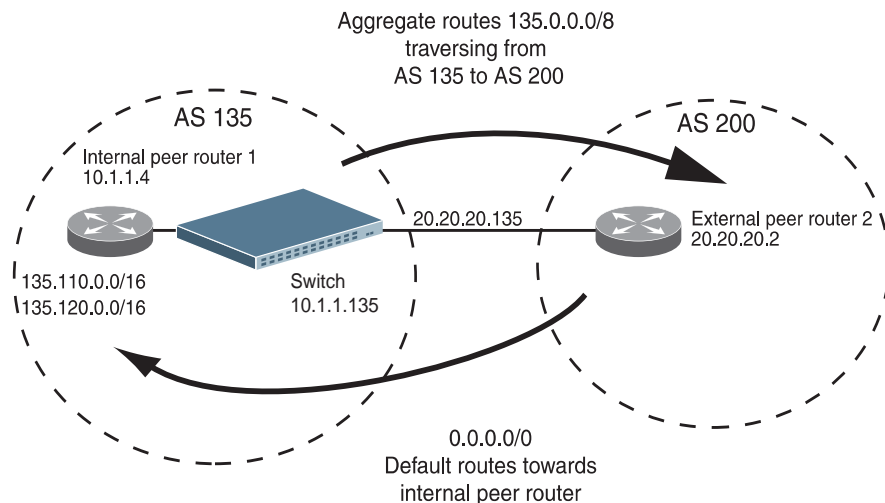
```
>> # router bgp
>> (config-router-bgp)# neighbor 1 remote-address 200.200.200.2
>> (config-router-bgp)# neighbor 1 remote-as 100
>> (config-router-bgp)# neighbor 2 remote-address 210.210.210.2
>> (config-router-bgp)# neighbor 2 remote-as 200
```


Default Redistribution and Route Aggregation Example

This example shows you how to configure the switch to redistribute information from one routing protocol to another and create an aggregate route entry in the BGP routing table to minimize the size of the routing table.

As illustrated in [Figure 38](#), you have two peer routers: an internal and an external peer router. Configure the G8296 to redistribute the default routes from AS 200 to AS 135. At the same time, configure for route aggregation to allow you to condense the number of routes traversing from AS 135 to AS 200.

Figure 38. Route Aggregation and Default Route Redistribution



1. Configure the IP interface.
2. Configure the AS number (AS 135) and router ID (10.1.1.135).

```
>> # router bgp
>> (config-router-bgp)# as 135
>> (config-router-bgp)# exit
>> # ip router-id 10.1.1.135
```

3. Configure internal peer router 1 and external peer router 2 with IPv4 addresses.

```
>> # router bgp
>> (config-router-bgp)# neighbor 1 remote-address 10.1.1.4
>> (config-router-bgp)# neighbor 1 remote-as 135
>> (config-router-bgp)# neighbor 2 remote-address 20.20.20.2
>> (config-router-bgp)# neighbor 2 remote-as 200
```

4. Configure redistribution for Peer 1.

```
>> (config-router-bgp)# neighbor 1 redistribute default-action
redistribute
>> (config-router-bgp)# neighbor 1 redistribute fixed
```

5. Configure aggregation policy control.

Configure the IPv4 routes that you want aggregated.

```
>> (config-router-bgp)# aggregate-address 1 135.0.0.0 255.0.0.0  
>> (config-router-bgp)# aggregate-address 1 enable
```

Chapter 32. OSPF

Lenovo Network Operating System supports the Open Shortest Path First (OSPF) routing protocol. The Networking OS implementation conforms to the OSPF version 2 specifications detailed in Internet RFC 1583, and OSPF version 3 specifications in RFC 5340. The following sections discuss OSPF support for the RackSwitch G8296:

- [“OSPFv2 Overview” on page 444](#). This section provides information on OSPFv2 concepts, such as types of OSPF areas, types of routing devices, neighbors, adjacencies, link state database, authentication, and internal versus external routing.
- [“OSPFv2 Implementation in Networking OS” on page 449](#). This section describes how OSPFv2 is implemented in N/OS, such as configuration parameters, electing the designated router, summarizing routes, defining route maps and so forth.
- [“OSPFv2 Configuration Examples” on page 459](#). This section provides step-by-step instructions on configuring different OSPFv2 examples:
 - Creating a simple OSPF domain
 - Creating virtual links
 - Summarizing routes
- [“OSPFv3 Implementation in Networking OS” on page 468](#). This section describes differences and additional features found in OSPFv3.

OSPFv2 Overview

OSPF is designed for routing traffic within a single IP domain called an Autonomous System (AS). The AS can be divided into smaller logical units known as *areas*.

All routing devices maintain link information in their own Link State Database (LSDB). OSPF allows networks to be grouped together into an area. The topology of an area is hidden from the rest of the AS, thereby reducing routing traffic. Routing within an area is determined only by the area's own topology, thus protecting it from bad routing data. An area can be generalized as an IP subnetwork.

The following sections describe key OSPF concepts.

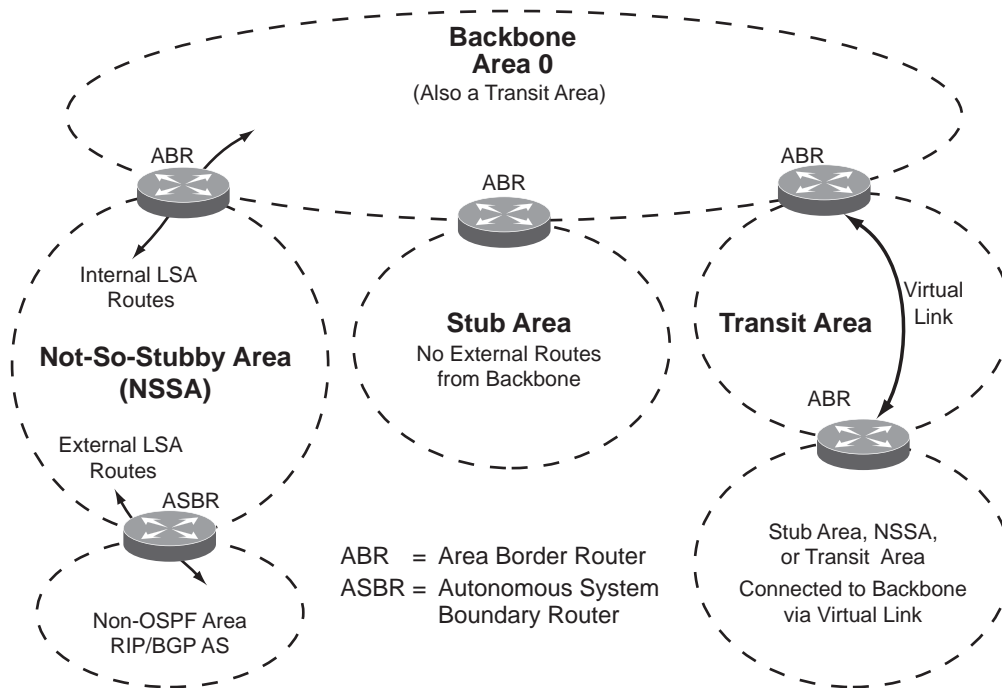
Types of OSPF Areas

An AS can be broken into logical units known as *areas*. In any AS with multiple areas, one area must be designated as area 0, known as the *backbone*. The backbone acts as the central OSPF area. All other areas in the AS must be connected to the backbone. Areas inject summary routing information into the backbone, which then distributes it to other areas as needed.

As shown in [Figure 39](#), OSPF defines the following types of areas:

- **Stub Area**—an area that is connected to only one other area. The stub area receives information from other areas, but external route information is not distributed into stub areas.
- **Not-So-Stubby-Area (NSSA)**—similar to a stub area with additional capabilities. Routes originating from within the NSSA can be propagated to adjacent transit and backbone areas. External routes from outside the AS can be advertised within the NSSA, but external routes coming from backbone area are not accepted.
- **Transit Area**—an area that also carries data traffic which neither originates nor terminates in the area itself.

Figure 39. OSPF Area Types

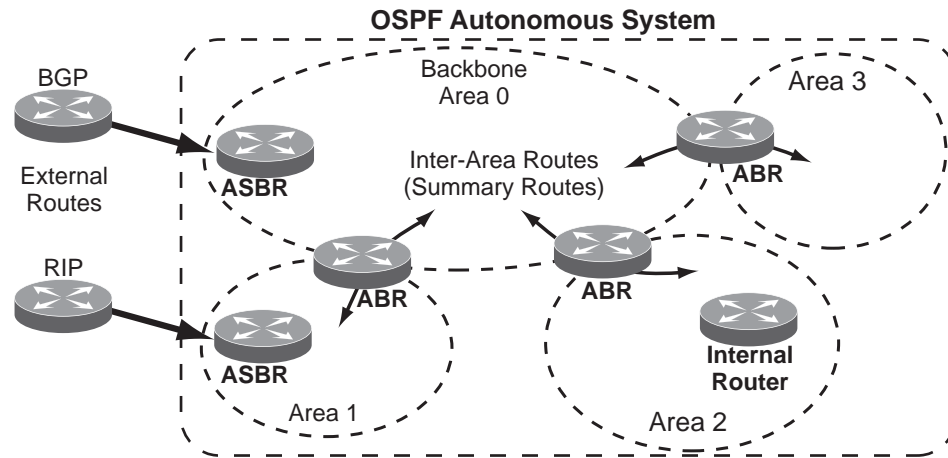


Types of OSPF Routing Devices

As shown in [Figure 40](#), OSPF uses the following types of routing devices:

- Internal Router (IR)—a router that has all of its interfaces within the same area. IRs maintain LSDBs identical to those of other routing devices within the local area.
- Area Border Router (ABR)—a router that has interfaces in multiple areas. ABRs maintain one LSDB for each connected area and disseminate routing information between areas.
- Autonomous System Boundary Router (ASBR)—a router that acts as a gateway between the OSPF domain and non-OSPF domains, such as RIP, BGP, and static routes.

Figure 40. OSPF Domain and an Autonomous System



Neighbors and Adjacencies

In areas with two or more routing devices, *neighbors* and *adjacencies* are formed.

Neighbors are routing devices that maintain information about each others' state. To establish neighbor relationships, routing devices periodically send hello packets on each of their interfaces. All routing devices that share a common network segment, appear in the same area, and have the same health parameters (hello and dead intervals) and authentication parameters respond to each other's hello packets and become neighbors. Neighbors continue to send periodic hello packets to advertise their health to neighbors. In turn, they listen to hello packets to determine the health of their neighbors and to establish contact with new neighbors.

The hello process is used for electing one of the neighbors as the network segment's Designated Router (DR) and one as the network segment's Backup Designated Router (BDR). The DR is adjacent to all other neighbors on that specific network segment and acts as the central contact for database exchanges. Each neighbor sends its database information to the DR, which relays the information to the other neighbors.

The BDR is adjacent to all other neighbors (including the DR). Each neighbor sends its database information to the BDR just as with the DR, but the BDR merely stores this data and does not distribute it. If the DR fails, the BDR will take over the task of distributing database information to the other neighbors.

The Link-State Database

OSPF is a link-state routing protocol. A *link* represents an interface (or routable path) from the routing device. By establishing an adjacency with other OSPF routers, each routing device in an OSPF area maintains an identical Link-State Database (LSDB) describing the network topology for its area.

Each routing device transmits a Link-State Advertisement (LSA) on each of its *active* interfaces. LSAs are entered into the LSDB of each routing device. OSPF uses *flooding* to distribute LSAs between routing devices. Interfaces may also be *passive*. Passive interfaces send LSAs to active interfaces, but do not receive LSAs, hello

packets, or any other OSPF protocol information from active interfaces. Passive interfaces behave as stub networks, allowing OSPF routing devices to be aware of devices that do otherwise participate in OSPF (either because they do not support it, or because the administrator chooses to restrict OSPF traffic exchange or transit).

When LSAs result in changes to the routing device's LSDB, the routing device forwards the changes to the adjacent neighbors (the DR and BDR) for distribution to the other neighbors.

OSPF routing updates occur only when changes occur, instead of periodically. For each new route, if a neighbor is interested in that route (for example, if configured to receive static routes and the new route is indeed static), an update message containing the new route is sent to the adjacency. For each route removed from the route table, if the route has already been sent to a neighbor, an update message containing the route to withdraw is sent.

The Shortest Path First Tree

The routing devices use a link-state algorithm (Dijkstra's algorithm) to calculate the shortest path to all known destinations, based on the cumulative *cost* required to reach the destination.

The cost of an individual interface in OSPF is an indication of the overhead required to send packets across it. The cost is inversely proportional to the bandwidth of the interface. A lower cost indicates a higher bandwidth.

Internal Versus External Routing

To ensure effective processing of network traffic, every routing device on your network needs to know how to send a packet (directly or indirectly) to any other location/destination in your network. This is referred to as *internal routing* and can be done with static routes or using active internal routing protocols, such as OSPF, RIP, or RIPv2.

It is also useful to tell routers outside your network (upstream providers or *peers*) about the routes you have access to in your network. Sharing of routing information between autonomous systems is known as *external routing*.

Typically, an AS will have one or more border routers (peer routers that exchange routes with other OSPF networks) as well as an internal routing system enabling every router in that AS to reach every other router and destination within that AS.

When a routing device *advertises* routes to boundary routers on other autonomous systems, it is effectively committing to carry data to the IP space represented in the route being advertised. For example, if the routing device advertises 192.204.4.0/24, it is declaring that if another router sends data destined for any address in the 192.204.4.0/24 range, it will carry that data to its destination.

OSPFv2 Implementation in Networking OS

N/OS supports a single instance of OSPF and up to 4K routes on the network. The following sections describe OSPF implementation in N/OS:

- [“Configurable Parameters” on page 449](#)
- [“Defining Areas” on page 450](#)
- [“Interface Cost” on page 452](#)
- [“Electing the Designated Router and Backup” on page 452](#)
- [“Summarizing Routes” on page 452](#)
- [“Default Routes” on page 453](#)
- [“Virtual Links” on page 454](#)
- [“Router ID” on page 454](#)
- [“Authentication” on page 455](#)

Configurable Parameters

In N/OS, OSPF parameters can be configured through the Command Line Interfaces (CLI/ISCLI), Browser-Based Interface (BBI), or through SNMP. For more information, see [“Switch Administration” on page 29](#).

The ISCLI supports the following parameters: interface output cost, interface priority, dead and hello intervals, retransmission interval, and interface transmit delay.

In addition to the preceding parameters, you can specify the following:

- Shortest Path First (SPF) interval—Time interval between successive calculations of the shortest path tree using the Dijkstra’s algorithm.
- Stub area metric—A stub area can be configured to send a numeric metric value such that all routes received via that stub area carry the configured metric to potentially influence routing decisions.
- Default routes—Default routes with weight metrics can be manually injected into transit areas. This helps establish a preferred route when multiple routing devices exist between two areas. It also helps route traffic to external networks.
- Passive—When enabled, the interface sends LSAs to upstream devices, but does not otherwise participate in OSPF protocol exchanges.
- Point-to-Point—For LANs that have only two OSPF routing agents (the G8296 and one other device), this option allows the switch to significantly reduce the amount of routing information it must carry and manage.

Defining Areas

If you are configuring multiple areas in your OSPF domain, one of the areas must be designated as area 0, known as the *backbone*. The backbone is the central OSPF area and is usually physically connected to all other areas. The areas inject routing information into the backbone which, in turn, disseminates the information into other areas.

Since the backbone connects the areas in your network, it must be a contiguous area. If the backbone is partitioned (possibly as a result of joining separate OSPF networks), parts of the AS will be unreachable, and you will need to configure *virtual links* to reconnect the partitioned areas (see “[Virtual Links](#)” on page 454).

Up to six OSPF areas can be connected to the G8296 with N/OS software. To configure an area, the OSPF number must be defined and then attached to a network interface on the switch. The full process is explained in the following sections.

An OSPF area is defined by assigning *two* pieces of information: an *area index* and an *area ID*. The commands to define and enable an OSPF area are as follows:

```
RS8296(config)# router ospf
RS8296(config-router-ospf)# area <area index> area-id <n.n.n.n>
RS8296(config-router-ospf)# area <area index> enable
RS8296(config-router-ospf)# exit
```

Note: The area option is an arbitrary index used only on the switch and does not represent the actual OSPF area number. The actual OSPF area number is defined in the area portion of the command as explained in the following sections.

Assigning the Area Index

The area <area index> option is actually just an arbitrary index (0-19) used only by the G8296. This index number does not necessarily represent the OSPF area number, though for configuration simplicity, it ought to where possible.

For example, both of the following sets of commands define OSPF area 0 (the backbone) and area 1 because that information is held in the area ID portion of the command. However, the first set of commands is easier to maintain because the arbitrary area indexes agree with the area IDs:

- Area index and area ID agree

```
area 0 area-id 0.0.0.0           (Use index 0 to set area 0 in ID octet
                                format)
area 1 area-id 0.0.0.1           (Use index 1 to set area 1 in ID octet
                                format)
```
- Area index set to an arbitrary value

```
area 1 area-id 0.0.0.0           (Use index 1 to set area 0 in ID octet
                                format)
area 2 area-id 0.0.0.1           (Use index 2 to set area 1 in ID octet
                                format)
```

Using the Area ID to Assign the OSPF Area Number

The OSPF area number is defined in the `area-id <IP address>` option. The octet format is used to be compatible with two different systems of notation used by other OSPF network vendors. There are two valid ways to designate an area ID:

- **Single Number**
Most common OSPF vendors express the area ID number as a single number. For example, the Cisco IOS-based router command “`network 1.1.1.0 0.0.0.255 area 1`” defines the area number simply as “`area 1`.”
- **Multi-octet (*IP address*):** Placing the area number in the last octet (`0.0.0.n`)
Some OSPF vendors express the area ID number in multi-octet format. For example, “`area 0.0.0.2`” represents OSPF area 2 and can be specified directly on the G8296 as “`area-id 0.0.0.2`”.

On the G8296, using the last octet in the area ID, “`area 1`” is equivalent to “`area-id 0.0.0.1`”.

Note: Although both types of area ID formats are supported, be sure that the area IDs are in the same format throughout an area.

Attaching an Area to a Network

Once an OSPF area has been defined, it must be associated with a network. To attach the area to a network, you must assign the OSPF area index to an IP interface that participates in the area. The format for the command is as follows:

```
RS8296(config)# interface ip <interface number>
RS8296(config-ip-if)# ip ospf area <area index>
RS8296(config-ip-if)# exit
```

For example, the following commands could be used to configure IP interface 14 to use 10.10.10.1 on the 10.10.10.0/24 network, to define OSPF area 1, and to attach the area to the network:

```
RS8296(config)# router ospf
RS8296(config-router-ospf)# area 1 area-id 0.0.0.1
RS8296(config-router-ospf)# area 1 enable
RS8296(config-router-ospf)# enable
RS8296(config-router-ospf)# exit
RS8296(config)# interface ip 14
RS8296(config-ip-if)# ip address 10.10.10.1
RS8296(config-ip-if)# ip netmask 255.255.255.0
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# ip ospf area 1
RS8296(config-ip-if)# ip ospf enable
```

Note: OSPFv2 supports IPv4 only. IPv6 is supported in OSPFv3 (see “[OSPFv3 Implementation in Networking OS](#)” on page 468).

Interface Cost

The OSPF link-state algorithm (Dijkstra's algorithm) places each routing device at the root of a tree and determines the cumulative *cost* required to reach each destination. Usually, the cost is inversely proportional to the bandwidth of the interface. Low cost indicates high bandwidth. You can manually enter the cost for the output route with the following command (Interface IP mode):

```
RS8296(config-ip-if)# ip ospf cost <cost value (1-65535)>
```

Electing the Designated Router and Backup

In any area with more than two routing devices, a Designated Router (DR) is elected as the central contact for database exchanges among neighbors, and a Backup Designated Router (BDR) is elected in case the DR fails.

DR and BDR elections are made through the `hello` process. The election can be influenced by assigning a priority value to the OSPF interfaces on the G8296. The command is as follows:

```
RS8296(config-ip-if)# ip ospf priority <priority value (0-255)>
```

A priority value of 255 is the highest, and 1 is the lowest. A priority value of 0 specifies that the interface cannot be used as a DR or BDR. In case of a tie, the routing device with the highest router ID wins. Interfaces configured as *passive* do not participate in the DR or BDR election process:

```
RS8296(config-ip-if)# ip ospf passive-interface  
RS8296(config-ip-if)# exit
```

Summarizing Routes

Route summarization condenses routing information. Without summarization, each routing device in an OSPF network would retain a route to every subnet in the network. With summarization, routing devices can reduce some sets of routes to a single advertisement, reducing both the load on the routing device and the perceived complexity of the network. The importance of route summarization increases with network size.

Summary routes can be defined for up to 16 IP address ranges using the following command:

```
RS8296(config)# router ospf  
RS8296(config-router-ospf)# area-range <range number> address  
                                  <IP address> <mask>
```

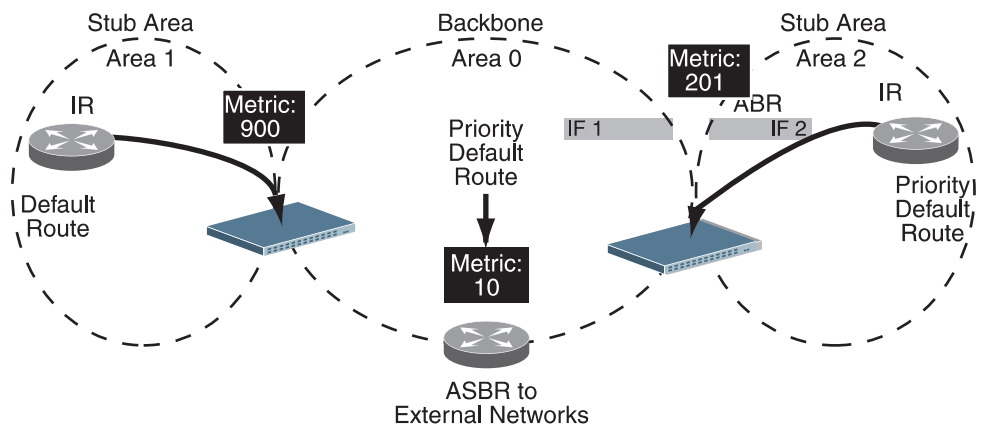
where <range number> is a number 1 to 16, <IP address> is the base IP address for the range, and <mask> is the IP address mask for the range. For a detailed configuration example, see [“Example 3: Summarizing Routes” on page 466](#).

Default Routes

When an OSPF routing device encounters traffic for a destination address it does not recognize, it forwards that traffic along the *default route*. Typically, the default route leads upstream toward the backbone until it reaches the intended area or an external router.

Each G8296 acting as an ABR automatically inserts a default route into each attached area. In simple OSPF stub areas or NSSAs with only one ABR leading upstream (see Area 1 in [Figure 41](#)), any traffic for IP address destinations outside the area is forwarded to the switch's IP interface, and then into the connected transit area (usually the backbone). Since this is automatic, no further configuration is required for such areas.

Figure 41. Injecting Default Routes



If the switch is in a transit area and has a configured default gateway, it can inject a default route into rest of the OSPF domain. Use the following command to configure the switch to inject OSPF default routes (Router OSPF mode):

```
RS8296(config-router-ospf)# default-information <metric value>  
                           <metric type (1 or 2)>
```

In this command, *<metric value>* sets the priority for choosing this switch for default route. The value *none* sets no default and 1 sets the highest priority for default route. Metric type determines the method for influencing routing decisions for external routes.

When the switch is configured to inject a default route, an AS-external LSA with link state ID 0.0.0.0 is propagated throughout the OSPF routing domain. This LSA is sent with the configured metric value and metric type.

The OSPF default route configuration can be removed with the command:

```
RS8296(config-router-ospf)# no default-information
```

Virtual Links

Usually, all areas in an OSPF AS are physically connected to the backbone. In some cases where this is not possible, you can use a *virtual link*. Virtual links are created to connect one area to the backbone through another non-backbone area (see [Figure 39 on page 445](#)).

The area which contains a virtual link must be a transit area and have full routing information. Virtual links cannot be configured inside a stub area or NSSA. The area type must be defined as `transit` using the following command:

```
RS8296(config-router-ospf)# area <area index> type transit
```

The virtual link must be configured on the routing devices at each endpoint of the virtual link, though they may traverse multiple routing devices. To configure a G8296 as one endpoint of a virtual link, use the following command:

```
RS8296(config-router-ospf)# area-virtual-link <link number> neighbor-router  
<router ID>
```

where <link number> is a value between 1 and 3, <area index> is the OSPF area index of the transit area, and <router ID> is the IP address of the virtual neighbor, the routing device at the target endpoint. Another router ID is needed when configuring a virtual link in the other direction. To provide the G8296 with a router ID, see the following section [Router ID](#).

For a detailed configuration example on Virtual Links, see [“Example 2: Virtual Links” on page 462](#).

Router ID

Routing devices in OSPF areas are identified by a router ID. The router ID is expressed in IP address format. The IP address of the router ID is not required to be included in any IP interface range or in any OSPF area, and may even use the G8296 loopback interface.

The router ID can be configured in one of the following two ways:

- Dynamically—OSPF protocol configures the lowest IP interface IP address as the router ID (loopback interface has priority over the IP interface). This is the default.
- Statically—Use the following command to manually configure the router ID:

```
RS8296(config-router-ospf)# ip router-id <IPv4 address>
```

If there is a loopback interface, its IP address is always preferred as the router ID, instead of an IP interface address. The `ip router-id` command is the preferred method to set the router ID and it is always used in preference to the other methods.

- To modify the router ID from static to dynamic, set the router ID to 0.0.0.0, save the configuration, and reboot the G8296.
- To view the router ID, use the following command:

```
RS8296(config-router-ospf)# show ip ospf
```

Authentication

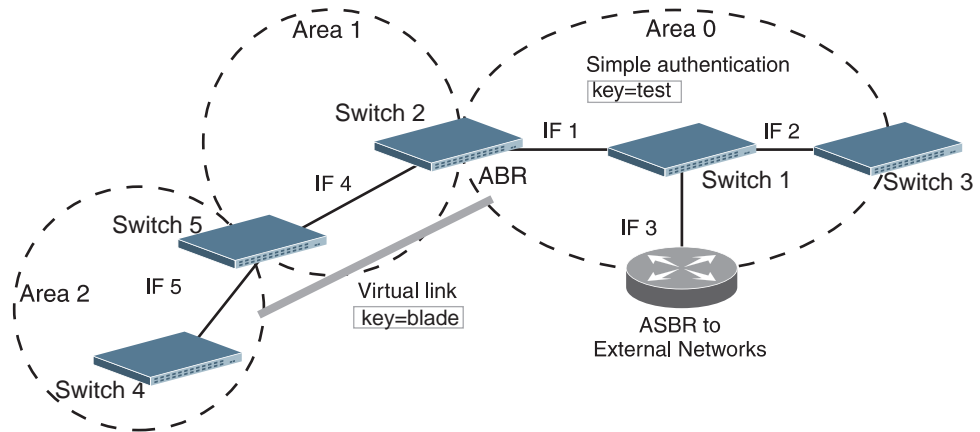
OSPF protocol exchanges can be authenticated so that only trusted routing devices can participate. This ensures less processing on routing devices that are not listening to OSPF packets.

OSPF allows packet authentication and uses IP multicast when sending and receiving packets. Routers participate in routing domains based on pre-defined passwords. N/OS supports simple password (type 1 plain text passwords) and MD5 cryptographic authentication. This type of authentication allows a password to be configured per area.

We strongly recommend that you implement MD5 cryptographic authentication as a best practice.

Figure 42 shows authentication configured for area 0 with the password test. Simple authentication is also configured for the virtual link between area 2 and area 0. Area 1 is not configured for OSPF authentication.

Figure 42. OSPF Authentication



Configuring Plain Text OSPF Passwords

To configure simple plain text OSPF passwords on the switches shown in [Figure 42](#) use the following commands:

1. Enable OSPF authentication for Area 0 on switches 1, 2, and 3.

```
RS8296(config-router-ospf)# area 0 authentication-type password
RS8296(config-router-ospf)# exit
```

2. Configure a simple text password up to eight characters for each OSPF IP interface in Area 0 on switches 1, 2, and 3.

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip ospf key test
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip ospf key test
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 3
RS8296(config-ip-if)# ip ospf key test
RS8296(config-ip-if)# exit
```

3. Enable OSPF authentication for Area 2 on switch 4.

```
RS8296(config)# router ospf
RS8296(config-router-ospf)# area 2 authentication-type password
```

4. Configure a simple text password up to eight characters for the virtual link between Area 2 and Area 0 on switches 2 and 4.

```
RS8296(config-router-ospf)# area-virtual-link 1 key blade
```

Configuring MD5 Authentication

Use the following commands to configure MD5 authentication on the switches shown in [Figure 42](#):

1. Enable OSPF MD5 authentication for Area 0 on switches 1, 2, and 3.

```
RS8296(config-router-ospf)# area 0 authentication-type md5
```

2. Configure MD5 key ID for Area 0 on switches 1, 2, and 3.

```
RS8296(config-router-ospf)# message-digest-key 1 md5-key test
RS8296(config-router-ospf)# exit
```


3. Assign MD5 key ID to OSPF interfaces on switches 1, 2, and 3.

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip ospf message-digest-key 1
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip ospf message-digest-key 1
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 3
RS8296(config-ip-if)# ip ospf message-digest-key 1
RS8296(config-ip-if)# exit
```

4. Enable OSPF MD5 authentication for Area 2 on switch 4.

```
RS8296(config)# router ospf
RS8296(config-router-ospf)# area 1 authentication-type md5
```

5. Configure MD5 key for the virtual link between Area 2 and Area 0 on switches 2 and 4.

```
RS8296(config-router-ospf)# message-digest-key 2 md5-key test
```

6. Assign MD5 key ID to OSPF virtual link on switches 2 and 4.

```
RS8296(config-router-ospf)# area-virtual-link 1 message-digest-key 2
RS8296(config-router-ospf)# exit
```

Host Routes for Load Balancing

N/OS implementation of OSPF includes host routes. Host routes are used for advertising network device IP addresses to external networks, accomplishing the following goals:

- ABR Load Sharing

As a form of load balancing, host routes can be used for dividing OSPF traffic among multiple ABRs. To accomplish this, each switch provides identical services but advertises a host route for a different IP address to the external network. If each IP address serves a different and equal portion of the external world, incoming traffic from the upstream router must be split evenly among ABRs.

- ABR Failover

Complementing ABR load sharing, identical host routes can be configured on each ABR. These host routes can be given different costs so that a different ABR is selected as the preferred route for each server and the others are available as backups for failover purposes.

- Equal Cost Multipath (ECMP)

With equal cost multipath, a router potentially has several available next hops towards any given destination. ECMP allows separate routes to be calculated for each IP Type of Service. All paths of equal cost to a given destination are calculated, and the next hops for all equal-cost paths are inserted into the routing table.

Loopback Interfaces in OSPF

A loopback interface is an IP interface which has an IP address, but is not associated with any particular physical port. The loopback interface is thus always available to the general network, regardless of which specific ports are in operation. Because loopback interfaces are always available on the switch, loopback interfaces may present an advantage when used as the router ID.

If dynamic router ID selection is used (see [“Router ID” on page 454](#)) loopback interfaces can be used to force router ID selection. If a loopback interface is configured, its IP address is automatically selected as the router ID, even if other IP interfaces have lower IP addresses. If more than one loopback interface is configured, the lowest loopback interface IP address is selected.

Loopback interfaces can be advertised into the OSPF domain by specifying an OSPF host route with the loopback interface IP address.

To enable OSPF on an existing loopback interface:

```
RS8296(config)# interface loopback <1-5>
RS8296(config-ip-loopback)# ip ospf area <area ID> enable
RS8296(config-ip-loopback)# exit
```

OSPF Features Not Supported in This Release

The following OSPF features are not supported in this release:

- Summarizing external routes
- Filtering OSPF routes
- Using OSPF to forward multicast routes
- Configuring OSPF on non-broadcast multi-access networks (such as frame relay, X.25, or ATM)

OSPFv2 Configuration Examples

A summary of the basic steps for configuring OSPF on the G8296 is listed here. Detailed instructions for each of the steps is covered in the following sections:

1. Configure IP interfaces.

One IP interface is required for each desired network (range of IP addresses) being assigned to an OSPF area on the switch.

2. (Optional) Configure the router ID.

The router ID is required only when configuring virtual links on the switch.

3. Enable OSPF on the switch.

4. Define the OSPF areas.

5. Configure OSPF interface parameters.

IP interfaces are used for attaching networks to the various areas.

6. (Optional) Configure route summarization between OSPF areas.

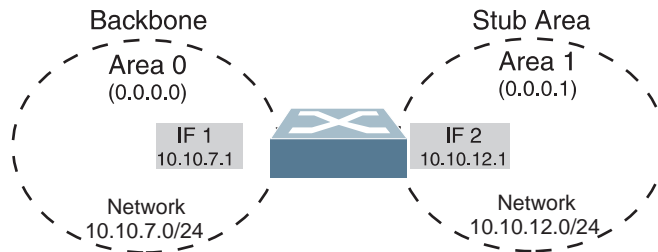
7. (Optional) Configure virtual links.

8. (Optional) Configure host routes.

Example 1: Simple OSPF Domain

In this example, two OSPF areas are defined—one area is the backbone and the other is a stub area. A stub area does not allow advertisements of external routes, thus reducing the size of the database. Instead, a default summary route of IP address 0.0.0.0 is automatically inserted into the stub area. Any traffic for IP address destinations outside the stub area will be forwarded to the stub area's IP interface, and then into the backbone.

Figure 43. A Simple OSPF Domain



Follow this procedure to configure OSPF support as shown in [Figure 43](#):

1. Configure IP interfaces on each network that will be attached to OSPF areas.

In this example, two IP interfaces are needed:

- Interface 1 for the backbone network on 10.10.7.0/24
- Interface 2 for the stub area network on 10.10.12.0/24

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip address 10.10.7.1
RS8296(config-ip-if)# ip netmask 255.255.255.0
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip address 10.10.12.1
RS8296(config-ip-if)# ip netmask 255.255.255.0
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
```

Note: OSPFv2 supports IPv4 only. IPv6 is supported in OSPFv3 (see [“OSPFv3 Implementation in Networking OS” on page 468](#)).

2. Enable OSPF.

```
RS8296(config)# router ospf
RS8296(config-router-ospf)# enable
```

3. Define the backbone.

The backbone is always configured as a transit area using `areaid 0.0.0.0`.

```
RS8296(config-router-ospf)# area 0 area-id 0.0.0.0
RS8296(config-router-ospf)# area 0 type transit
RS8296(config-router-ospf)# area 0 enable
```

4. Define the stub area.

```
RS8296(config-router-ospf)# area 1 area-id 0.0.0.1  
RS8296(config-router-ospf)# area 1 type stub  
RS8296(config-router-ospf)# area 1 enable  
RS8296(config-router-ospf)# exit
```

5. Attach the network interface to the backbone.

```
RS8296(config)# interface ip 1  
RS8296(config-ip-if)# ip ospf area 0  
RS8296(config-ip-if)# ip ospf enable  
RS8296(config-ip-if)# exit
```

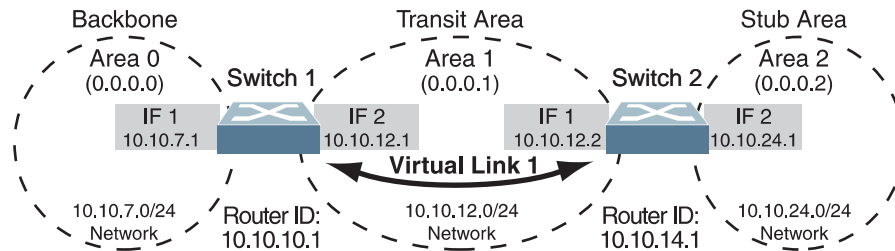
6. Attach the network interface to the stub area.

```
RS8296(config)# interface ip 2  
RS8296(config-ip-if)# ip ospf area 1  
RS8296(config-ip-if)# ip ospf enable  
RS8296(config-ip-if)# exit
```

Example 2: Virtual Links

In the example shown in [Figure 44](#), area 2 is not physically connected to the backbone as is usually required. Instead, area 2 will be connected to the backbone via a virtual link through area 1. The virtual link must be configured at each endpoint.

Figure 44. Configuring a Virtual Link



Note: OSPFv2 supports IPv4 only. IPv6 is supported in OSPFv3 (see [“OSPFv3 Implementation in Networking OS”](#) on page 468).

Configuring OSPF for a Virtual Link on Switch #1

1. Configure IP interfaces on each network that will be attached to the switch.

In this example, two IP interfaces are needed:

- Interface 1 for the backbone network on 10.10.7.0/24
- Interface 2 for the transit area network on 10.10.12.0/24

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip address 10.10.7.1
RS8296(config-ip-if)# ip netmask 255.255.255.0
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip address 10.10.12.1
RS8296(config-ip-if)# ip netmask 255.255.255.0
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
```

2. Configure the router ID.

A router ID is required when configuring virtual links. Later, when configuring the other end of the virtual link on Switch 2, the router ID specified here will be used as the target virtual neighbor (nbr) address.

```
RS8296(config)# ip router-id 10.10.10.1
```

3. Enable OSPF.

```
RS8296(config)# router ospf
RS8296(config-router-ospf)# enable
```

4. Define the backbone.

```
RS8296(config-router-ospf)# area 0 area-id 0.0.0.0
RS8296(config-router-ospf)# area 0 type transit
RS8296(config-router-ospf)# area 0 enable
```

5. Define the transit area.

The area that contains the virtual link must be configured as a transit area.

```
RS8296(config-router-ospf)# area 1 area-id 0.0.0.1
RS8296(config-router-ospf)# area 1 type transit
RS8296(config-router-ospf)# area 1 enable
RS8296(config-router-ospf)# exit
```

6. Attach the network interface to the backbone.

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip ospf area 0
RS8296(config-ip-if)# ip ospf enable
RS8296(config-ip-if)# exit
```

7. Attach the network interface to the transit area.

```
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip ospf area 1
RS8296(config-ip-if)# ip ospf enable
RS8296(config-ip-if)# exit
```

8. Configure the virtual link.

The `nbr` router ID configured in this step must be the same as the router ID that will be configured for Switch #2 in [Step 2 on page 464](#).

```
RS8296(config)# router ospf
RS8296(config-router-ospf)# area-virtual-link 1 area 1
RS8296(config-router-ospf)# area-virtual-link 1 neighbor-router
10.10.14.1
RS8296(config-router-ospf)# area-virtual-link 1 enable
```

Configuring OSPF for a Virtual Link on Switch #2

1. Configure IP interfaces on each network that will be attached to OSPF areas.

In this example, two IP interfaces are needed:

- Interface 1 for the transit area network on 10.10.12.0/24

- Interface 2 for the stub area network on 10.10.24.0/24

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip address 10.10.12.2
RS8296(config-ip-if)# ip netmask 255.255.255.0
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip address 10.10.24.1
RS8296(config-ip-if)# ip netmask 255.255.255.0
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
```

2. Configure the router ID.

A router ID is required when configuring virtual links. This router ID must be the same one specified as the target virtual neighbor (nbr) on switch 1 in [Step 8 on page 463](#).

```
RS8296(config)# ip router-id 10.10.14.1
```

3. Enable OSPF.

```
RS8296(config)# router ospf
RS8296(config-router-ospf)# enable
```

4. Define the backbone.

This version of N/OS requires that a backbone index be configured on the non-backbone end of the virtual link as follows:

```
RS8296(config-router-ospf)# area 0 area-id 0.0.0.0
RS8296(config-router-ospf)# area 0 enable
```

5. Define the transit area.

```
RS8296(config-router-ospf)# area 1 area-id 0.0.0.1
RS8296(config-router-ospf)# area 1 type transit
RS8296(config-router-ospf)# area 1 enable
```

6. Define the stub area.

```
RS8296(config-router-ospf)# area 2 area-id 0.0.0.2
RS8296(config-router-ospf)# area 2 type stub
RS8296(config-router-ospf)# area 2 enable
RS8296(config-router-ospf)# exit
```

7. Attach the network interface to the transmit area.

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip ospf area 1
RS8296(config-ip-if)# ip ospf enable
RS8296(config-ip-if)# exit
```


8. Attach the network interface to the stub area.

```
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip ospf area 2
RS8296(config-ip-if)# ip ospf enable
RS8296(config-ip-if)# exit
```

9. Configure the virtual link.

The `nbr` router ID configured in this step must be the same as the router ID that was configured for switch #1 in [Step 2 on page 462](#).

```
RS8296(config)# router ospf
RS8296(config-router-ospf)# area-virtual-link 1 area 1
RS8296(config-router-ospf)# area-virtual-link 1 neighbor-router
10.10.10.1
RS8296(config-router-ospf)# area-virtual-link 1 enable
```

Other Virtual Link Options

- You can use redundant paths by configuring multiple virtual links.
- Only the endpoints of the virtual link are configured. The virtual link path may traverse multiple routers in an area as long as there is a routable path between the endpoints.

Example 3: Summarizing Routes

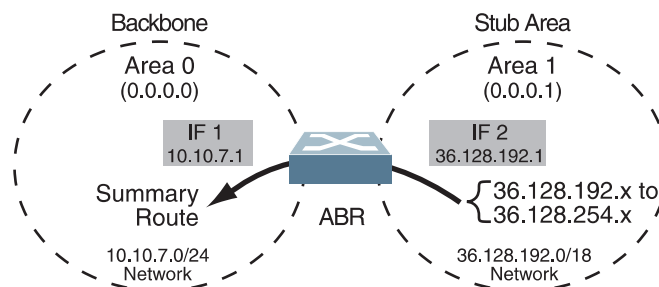
By default, ABRs advertise all the network addresses from one area into another area. Route summarization can be used for consolidating advertised addresses and reducing the perceived complexity of the network.

If network IP addresses in an area are assigned to a contiguous subnet range, you can configure the ABR to advertise a single summary route that includes all individual IP addresses within the area.

The following example shows one summary route from area 1 (stub area) injected into area 0 (the backbone). The summary route consists of all IP addresses from 36.128.192.0 through 36.128.254.255 except for the routes in the range 36.128.200.0 through 36.128.200.255.

Note: OSPFv2 supports IPv4 only. IPv6 is supported in OSPFv3 (see [“OSPFv3 Implementation in Networking OS” on page 468](#)).

Figure 45. Summarizing Routes



Note: You can specify a range of addresses to prevent advertising by using the `hide` option. In this example, routes in the range 36.128.200.0 through 36.128.200.255 are kept private.

Use the following procedure to configure OSPF support as shown in [Figure 45](#):

1. Configure IP interfaces for each network which will be attached to OSPF areas.

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip address 10.10.7.1
RS8296(config-ip-if)# ip netmask 255.255.255.0
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip address 36.128.192.1
RS8296(config-ip-if)# ip netmask 255.255.255.0
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
```

2. Enable OSPF.

```
RS8296(config)# router ospf
RS8296(config-router-ospf)# enable
```

3. Define the backbone.

```
RS8296(config-router-ospf)# area 0 area-id 0.0.0.0
RS8296(config-router-ospf)# area 0 type transit
RS8296(config-router-ospf)# area 0 enable
```

4. Define the stub area.

```
RS8296(config-router-ospf)# area 1 area-id 0.0.0.1
RS8296(config-router-ospf)# area 1 type stub
RS8296(config-router-ospf)# area 1 enable
RS8296(config-router-ospf)# exit
```

5. Attach the network interface to the backbone.

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip ospf area 0
RS8296(config-ip-if)# ip ospf enable
RS8296(config-ip-if)# exit
```

6. Attach the network interface to the stub area.

```
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip ospf area 1
RS8296(config-ip-if)# ip ospf enable
RS8296(config-ip-if)# exit
```

7. Configure route summarization by specifying the starting address and mask of the range of addresses to be summarized.

```
RS8296(config)# router ospf
RS8296(config-router-ospf)# area-range 1 address 36.128.192.0
255.255.192.0
RS8296(config-router-ospf)# area-range 1 area 1
RS8296(config-router-ospf)# area-range 1 enable
RS8296(config-router-ospf)# exit
```

8. Use the hide command to prevent a range of addresses from advertising to the backbone.

```
RS8296(config)# router ospf
RS8296(config-router-ospf)# area-range 2 address 36.128.200.0
255.255.255.0
RS8296(config-router-ospf)# area-range 2 area 1
RS8296(config-router-ospf)# area-range 2 hide
RS8296(config-router-ospf)# exit
```

Verifying OSPF Configuration

Use the following commands to verify the OSPF configuration on your switch:

- **show ip ospf**
- **show ip ospf neighbor**
- **show ip ospf database database-summary**
- **show ip ospf routes**

Refer to the *Lenovo Network Operating System Command Reference* for information on the preceding commands.

OSPFv3 Implementation in Networking OS

OSPF version 3 is based on OSPF version 2, but has been modified to support IPv6 addressing. In most other ways, OSPFv3 is similar to OSPFv2: They both have the same packet types and interfaces, and both use the same mechanisms for neighbor discovery, adjacency formation, LSA flooding, aging, and so on. The administrator must be familiar with the OSPFv2 concepts covered in the preceding sections of this chapter before implementing the OSPFv3 differences as described in the following sections.

Although OSPFv2 and OSPFv3 are very similar, they represent independent features on the G8296. They are configured separately, and both can run in parallel on the switch with no relation to one another, serving different IPv6 and IPv4 traffic, respectively.

The Networking OS implementation conforms to the OSPF version 3 authentication/confidentiality specifications in RFC 4552.

OSPFv3 Differences from OSPFv2

Note: When OSPFv3 is enabled, the OSPF backbone area (0.0.0.0) is created by default and is always active.

OSPFv3 Requires IPv6 Interfaces

OSPFv3 is designed to support IPv6 addresses. This requires IPv6 interfaces to be configured on the switch and assigned to OSPF areas, in much the same way IPv4 interfaces are assigned to areas in OSPFv2. This is the primary configuration difference between OSPFv3 and OSPFv2.

See [“Internet Protocol Version 6” on page 357](#) for configuring IPv6 interfaces.

OSPFv3 Uses Independent Command Paths

Though OSPFv3 and OSPFv2 are very similar, they are configured independently. OSPFv3 command paths are located as follows:

- In the ISCLI

```
RS8296(config)# ipv6 router ospf           (OSPFv3 router config mode)
RS8296(config-router-ospf3)# ?

RS8296(config)# interface ip <Interface number>(Configure OSPFv3)
RS8296(config-ip-if)# ipv6 ospf ?         (OSPFv3 interface config)

RS8296# show ipv6 ospf ?                 (Show OSPFv3 information)
```

OSPFv3 Identifies Neighbors by Router ID

Where OSPFv2 uses a mix of IPv4 interface addresses and Router IDs to identify neighbors, depending on their type, OSPFv3 configuration consistently uses a Router ID to identify all neighbors.

Although Router IDs are written in dotted decimal notation, and may even be based on IPv4 addresses from an original OSPFv2 network configuration, it is important to realize that Router IDs are not IP addresses in OSPFv3, and can be assigned independently of IP address space. However, maintaining Router IDs consistent with any legacy OSPFv2 IPv4 addressing allows for easier implementation of both protocols.

Other Internal Improvements

OSPFv3 has numerous improvements that increase the protocol efficiency in addition to supporting IPv6 addressing. These improvements change some of the behaviors in the OSPFv3 network and may affect topology consideration, but have little direct impact on configuration. For example:

- Addressing fields have been removed from Router and Network LSAs.
- Flexible treatment of unknown LSA types to make integration of OSPFv3 easier.
- Interface network type can be specified using the command:
RS8296(config-ip-if)# ipv6 ospf network
{broadcast|non-broadcast|point-to-multipoint|point-to-point}
- For an interface network type that is not broadcast or NBMA, link LSA suppression can be enabled so link LSA is not originated for the interface. Use the command: RS8296(config-ip-if)# ipv6 ospf linklsasuppress

OSPFv3 Limitations

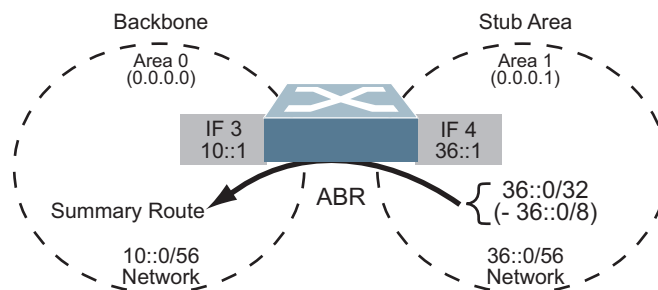
N/OS 8.2 does not currently support multiple interfaces of OSPFv3 on the same link.

OSPFv3 Configuration Example

The following example depicts the OSPFv3 equivalent configuration of [“Example 3: Summarizing Routes” on page 466](#) for OSPFv2.

In this example, one summary route from area 1 (stub area) is injected into area 0 (the backbone). The summary route consists of all IP addresses from the 36::0/32 portion of the 36::0/56 network, except for the routes in the 36::0/8 range.

Figure 46. Summarizing Routes



Note: You can specify a range of addresses to prevent advertising by using the hide option. In this example, routes in the 36::0/8 range are kept private.

Use the following procedure to configure OSPFv3 support as shown in [Figure 45](#):

1. Configure IPv6 interfaces for each link which will be attached to OSPFv3 areas.

```
RS8296(config)# interface ip 3
RS8296(config-ip-if)# ipv6 address 10:0:0:0:0:1
RS8296(config-ip-if)# ipv6 prefixlen 56
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 4
RS8296(config-ip-if)# ip address 36:0:0:0:0:1
RS8296(config-ip-if)# ipv6 prefixlen 56
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
```

This is equivalent to configuring the IP address and netmask for IPv4 interfaces.

2. Enable OSPFv3.

```
RS8296(config)# ipv6 router ospf
RS8296(config-router-ospf3)# enable
```

This is equivalent to the OSPFv2 enable option in the router ospf command path.

3. Define the backbone.

```
RS8296(config-router-ospf3)# area 0 area-id 0.0.0.0
RS8296(config-router-ospf3)# area 0 type transit
RS8296(config-router-ospf3)# area 0 enable
```

This is identical to OSPFv2 configuration.

4. Define the stub area.

```
RS8296(config-router-ospf3)# area 1 area-id 0.0.0.1
RS8296(config-router-ospf3)# area 1 type stub
RS8296(config-router-ospf3)# area 1 enable
RS8296(config-router-ospf3)# exit
```

This is identical to OSPFv2 configuration.

5. Attach the network interface to the backbone.

```
RS8296(config)# interface ip 3
RS8296(config-ip-if)# ipv6 ospf area 0
RS8296(config-ip-if)# ipv6 ospf enable
RS8296(config-ip-if)# exit
```

The ipv6 command path is used instead of the OSPFv2 ip command path

6. Attach the network interface to the stub area.

```
RS8296(config)# interface ip 4
RS8296(config-ip-if)# ipv6 ospf area 1
RS8296(config-ip-if)# ipv6 ospf enable
RS8296(config-ip-if)# exit
```

The `ipv6` command path is used instead of the OSPFv2 `ip` command path

7. Configure route summarization by specifying the starting address and prefix length of the range of addresses to be summarized.

```
RS8296(config)# ipv6 router ospf  
RS8296(config-router-ospf3)# area-range 1 address 36:0:0:0:0:0:0 32  
RS8296(config-router-ospf3)# area-range 1 area 0  
RS8296(config-router-ospf3)# area-range 1 enable
```

This differs from OSPFv2 only in that the OSPFv3 command path is used, and the address and prefix are specified in IPv6 format.

8. Use the `hide` command to prevent a range of addresses from advertising to the backbone.

```
RS8296(config-router-ospf)# area-range 2 address 36:0:0:0:0:0:0 8  
RS8296(config-router-ospf)# area-range 2 area 0  
RS8296(config-router-ospf)# area-range 2 hide  
RS8296(config-router-ospf)# exit
```

This differs from OSPFv2 only in that the OSPFv3 command path is used, and the address and prefix are specified in IPv6 format.

Neighbor Configuration Example

When using NBMA or point to multipoint interfaces, you must manually configure neighbors. Following example includes the steps for neighbor configuration.

1. Configure IPv6 interface parameters:

```
RS8296(config# interface ip 10
RS8296(config-ip-if)# ipv6 address 10:0:0:0:0:0:12 64
RS8296(config-ip-if)# vlan 10
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# ipv6 ospf area 0
RS8296(config-ip-if)# ipv6 ospf retransmit-interval 5
RS8296(config-ip-if)# ipv6 ospf transmit-delay 1
RS8296(config-ip-if)# ipv6 ospf priority 1
RS8296(config-ip-if)# ipv6 ospf hello-interval 10
RS8296(config-ip-if)# ipv6 ospf dead-interval 40
RS8296(config-ip-if)# ipv6 ospf network point-to-multipoint
RS8296(config-ip-if)# ipv6 ospf poll-interval 120
RS8296(config-ip-if)# ipv6 ospf enable
RS8296(config-ip-if)# exit
```

2. Enable OSPFv3:

```
RS8296(config# ipv6 router ospf
RS8296(config-router-ospf3)# router-id 12.12.12.12
RS8296(config-router-ospf3)# enable
```

3. Define the backbone.

```
RS8296(config-router-ospf3)# area 0 area-id 0.0.0.0
RS8296(config-router-ospf3)# area 0 stability-interval 40
RS8296(config-router-ospf3)# area 0 default-metric 1
RS8296(config-router-ospf3)# area 0 default-metric type 1
RS8296(config-router-ospf3)# area 0 translation-role candidate
RS8296(config-router-ospf3)# area 0 type transit
RS8296(config-router-ospf3)# area 0 enable
```

4. Configure neighbor entry:

```
RS8296(config-router-ospf3)# neighbor 1 address fe80:0:0:0:dceb:ff:fe00:9
RS8296(config-router-ospf3)# neighbor 1 interface 10
RS8296(config-router-ospf3)# neighbor 1 priority 1
RS8296(config-router-ospf3)# neighbor 1 enable
```

Chapter 33. Protocol Independent Multicast

Lenovo Network Operating System supports Protocol Independent Multicast (PIM) in Sparse Mode (PIM-SM) and Dense Mode (PIM-DM).

Note: Networking OS 8.2 does not support IPv6 for PIM.

The following sections discuss PIM support for the RackSwitch G8296:

- [“PIM Overview” on page 474](#)
- [“Supported PIM Modes and Features” on page 475](#)
- [“Basic PIM Settings” on page 476](#)
- [“Additional Sparse Mode Settings” on page 479](#)
- [“Using PIM with Other Features” on page 482](#)
- [“PIM Configuration Examples” on page 483](#)

PIM Overview

PIM is designed for efficiently routing multicast traffic across one or more IPv4 domains. This has benefits for application such as IP television, collaboration, education, and software delivery, where a single source must deliver content (a multicast) to a group of receivers that span both wide-area and inter-domain networks.

Instead of sending a separate copy of content to each receiver, a multicast derives efficiency by sending only a single copy of content toward its intended receivers. This single copy only becomes duplicated when it reaches the target domain that includes multiple receivers, or when it reaches a necessary bifurcation point leading to different receiver domains.

PIM is used by multicast source stations, client receivers, and intermediary routers and switches, to build and maintain efficient multicast routing trees. PIM is protocol independent; It collects routing information using the existing unicast routing functions underlying the IPv4 network, but does not rely on any particular unicast protocol. For PIM to function, a Layer 3 routing protocol (such as BGP, OSPF, RIP, or static routes) must first be configured on the switch.

PIM-SM is a reverse-path routing mechanism. Client receiver stations advertise their willingness to join a multicast group. The local routing and switching devices collect multicast routing information and forward the request toward the station that will provide the multicast content. When the join requests reach the sending station, the multicast data is sent toward the receivers, flowing in the opposite direction of the original join requests.

Some routing and switching devices perform special PIM-SM functions. Within each receiver domain, one router is elected as the Designated Router (DR) for handling multicasts for the domain. DRs forward information to a similar device, the Rendezvous Point (RP), which holds the root tree for the particular multicast group.

Receiver join requests as well as sender multicast content initially converge at the RP, which generates and distributes multicast routing data for the DRs along the delivery path. As the multicast content flows, DRs use the routing tree information obtained from the RP to optimize the paths both to and from send and receive stations, bypassing the RP for the remainder of content transactions if a more efficient route is available.

DRs continue to share routing information with the RP, modifying the multicast routing tree when new receivers join, or pruning the tree when all the receivers in any particular domain are no longer part of the multicast group.

Supported PIM Modes and Features

For each interface attached to a PIM network component, PIM can be configured to operate either in PIM Sparse Mode (PIM-SM) or PIM Dense Mode (PIM-DM).

- PIM-SM is used in networks where multicast senders and receivers comprise a relatively small (sparse) portion of the overall network. PIM-SM uses a more complex process than PIM-DM for collecting and optimizing multicast routes, but minimizes impact on other IP services and is more commonly used.
- PIM-DM is used where multicast devices are a relatively large (dense) portion of the network, with very frequent (or constant) multicast traffic. PIM-DM requires less configuration on the switch than PIM-SM, but uses broadcasts that can consume more bandwidth in establishing and optimizing routes.

The following PIM modes and features are *not* currently supported in N/OS 8.2:

- Hybrid Sparse-Dense Mode (PIM-SM/DM). Sparse Mode and Dense Mode may be configured on separate IP interfaces on the switch, but are not currently supported simultaneously on the same IP interface.
- PIM Source-Specific Multicast (PIM-SSM)
- Anycast RP
- PIM RP filters
- Only configuration via the switch CLI is supported. PIM configuration is currently not available using the BBI or via SNMP.

Basic PIM Settings

To use PIM the following is required:

- The PIM feature must be enabled globally on the switch.
- PIM network components and PIM modes must be defined.
- IP interfaces must be configured for each PIM component.
- PIM neighbor filters may be defined (optional).
- If PIM-SM is used, define additional parameters:
 - Rendezvous Point
 - Designated Router preferences (optional)
 - Bootstrap Router preferences (optional)

Each of these tasks is covered in the following sections.

Note: In N/OS 8.2, PIM can be configured through the switch CLI only. PIM configuration and information are not available using the BBI or via SNMP.

Globally Enabling or Disabling the PIM Feature

By default, PIM is disabled on the switch. PIM can be globally enabled or disabled using the following commands:

```
RS8296(config)# [no] ip pim enable
```

Defining a PIM Network Component

The G8296 can be attached to a maximum of two independent PIM network components. Each component represents a different PIM network, and can be defined for either PIM-SM or PIM-DM operation. Basic PIM component configuration is performed using the following commands:

```
RS8296(config)# ip pim component <1-2>  
RS8296(config-ip-pim-comp)# mode {sparse|dense}  
RS8296(config-ip-pim-comp)# exit
```

The sparse option will place the component in Sparse Mode (PIM-SM). The dense option will place the component in Dense Mode (PIM-DM). By default, PIM component 1 is configured for Sparse Mode. PIM component 2 is unconfigured by default.

Note: A component using PIM-SM must also be configured with a dynamic or static Rendezvous Point (see [“Specifying the Rendezvous Point” on page 479](#)).

Defining an IP Interface for PIM Use

Each network attached to an IP interface on the switch may be assigned one of the available PIM components. The same PIM component can be assigned to multiple IP interfaces. The interfaces may belong to the same VLAN, and they may also belong to different VLANs as long as their member IP addresses do not overlap.

To define an IP interface for use with PIM, first configured the interface with an IPv4 address and VLAN as follows:

```
RS8296(config)# interface ip <Interface number>
RS8296(config-ip-if)# ip address <IPv4 address> <IPv4 mask>
RS8296(config-ip-if)# vlan <VLAN number>
RS8296(config-ip-if)# enable
```

Note: The PIM feature currently supports only one VLAN for each IP interface. Configurations where different interfaces on different VLANs share IP addresses are not supported.

Next, PIM must be enabled on the interface, and the PIM network component ID must be specified:

```
RS8296(config-ip-if)# ip pim enable
RS8296(config-ip-if)# ip pim component-id <1-2>
RS8296(config-ip-if)# exit
```

By default, PIM component 1 is automatically assigned when PIM is enabled on the IP interface.

Note: While PIM is enabled on the interface, the interface VLAN cannot be changed. To change the VLAN, first disable PIM on the interface.

PIM Neighbor Filters

The G8296 accepts connection to up to 24 PIM interfaces. By default, the switch accepts all PIM neighbors attached to the PIM-enabled interfaces, up to the maximum number (72 neighbors). Once the maximum is reached, the switch will deny further PIM neighbors.

To ensure that only the appropriate PIM neighbors are accepted by the switch, the administrator can use PIM neighbor filters to specify which PIM neighbors may be accepted or denied on a per-interface basis.

To turn PIM neighbor filtering on or off for a particular IP interface, use the following commands:

```
RS8296(config)# interface ip <Interface number>
RS8296(config-ip-if)# [no] ip pim neighbor-filter
```

When filtering is enabled, all PIM neighbor requests on the specified IP interface will be denied by default. To allow a specific PIM neighbor, use the following command:

```
RS8296(config-ip-if)# ip pim neighbor-addr <neighbor IPv4 address> allow
```

To remove a PIM neighbor from the accepted list, use the following command.

```
RS8296(config-ip-if)# ip pim neighbor-addr <neighbor IPv4 address> deny
RS8296(config-ip-if)# exit
```

You can view configured PIM neighbor filters globally or for a specific IP interface using the following commands:

```
RS8296(config)# show ip pim neighbor-filters  
RS8296(config)# show ip pim interface <Interface number> neighbor-filters
```

Additional Sparse Mode Settings

Specifying the Rendezvous Point

Using PIM-SM, at least one PIM-capable router must be a candidate for use as a Rendezvous Point (RP) for any given multicast group. If desired, the G8296 can act as an RP candidate. To assign a configured switch IP interface as a candidate, use the following procedure.

1. Select the PIM component that will represent the RP candidate:

```
RS8296(config)# ip pim component <1-2>
```

2. Configure the IPv4 address of the switch interface which will be advertised as a candidate RP for the specified multicast group:

```
RS8296(config-ip-pim-comp)# rp-candidate rp-address <group address>  
<group address mask> <candidate IPv4 address>
```

The switch interface will participate in the election of the RP that occurs on the Bootstrap Router, or BSR (see [“Specifying a Bootstrap Router”](#) on page 480).

3. If using dynamic RP candidates, configure the amount of time that the elected interface will remain the RP for the group before a re-election is performed:

```
RS8296(config-ip-pim-comp)# rp-candidate holdtime <0-255>  
RS8296(config-ip-pim-comp)# exit
```

Static RP

If RP no election is desired, the switch can provide a static RP. Use the following commands:

1. Enable static RP configuration.

```
RS8296(config)# ip pim static-rp enable
```

2. Select the PIM component that will represent the RP candidate:

```
RS8296(config)# ip pim component <1-2>
```

3. Configure the static IPv4 address.

```
RS8296(config-ip-pim-comp)# rp-static rp-address <group address> <group address  
mask> <static IPv4 address>
```

Influencing the Designated Router Selection

Using PIM-SM, All PIM-enabled IP interfaces are considered as potential Designate Routers (DR) for their domain. By default, the interface with the highest IP address on the domain is selected. However, if an interface is configured with a

DR priority value, it overrides the IP address selection process. If more than one interface on a domain is configured with a DR priority, the one with the highest number is selected.

Use the following commands to configure the DR priority value (Interface IP mode):

```
RS8296(config)# interface ip <Interface number>
RS8296(config-ip-if)# ip pim dr-priority <value (0-4294967294)>
RS8296(config-ip-if)# exit
```

Note: A value of 0 (zero) specifies that the G8296 will not act as the DR. This setting requires the G8296 to be connected to a peer that has a DR priority setting of 1 or higher to ensure that a DR will be present in the network.

Specifying a Bootstrap Router

Using PIM-SM, a Bootstrap Router (BSR) is a PIM-capable router that hosts the election of the RP from available candidate routers. For each PIM-enabled IP interface, the administrator can set the preference level for which the local interface becomes the BSR:

```
RS8296(config)# interface ip <Interface number>
RS8296(config-ip-if)# ip pim cbsr-preference <0 to 255>
RS8296(config-ip-if)# exit
```

A value of 255 highly prefers the local interface as a BSR. A value of 0 indicates that the PIM CBSR preference is not configured on the switch.

Configuring a Loopback Interface

Loopback interfaces can be used in PIM Sparse Mode for Rendezvous Points (RPs) and Bootstrap Routers (BSRs). For example:

- As a static RP

```
interface loopback 1
  ip address 55.55.1.1 255.255.255.0
  enable
  exit

ip pim static-rp enable

ip pim component 1
  rp-static rp-address 224.0.0.0 240.0.0.0 55.55.1.1

interface loopback 1
  ip pim enable
  exit
```


- As a candidate RP

```
interface loopback 1
  ip address 55.55.1.1 255.255.255.0
  enable
  exit

ip pim component 1
  rp-candidate holdtime 60
  rp-candidate rp-address 224.0.0.0 240.0.0.0 55.55.1.1

interface loopback 1
  ip pim enable
  exit
```

- As a BSR

```
interface loopback 1
  ip address 55.55.1.1 255.255.255.0
  enable
  exit

interface loopback 1
  ip pim enable
  ip pim cbsr-preference 2
  exit
```

Using PIM with Other Features

PIM with ACLs or VMAPs

If using ACLs or VMAPs, be sure to permit traffic for local hosts and routers.

PIM with IGMP

If using IGMP (see [“Internet Group Management Protocol” on page 387](#)):

- IGMP static joins can be configured with a PIM-SM or PIM-DM multicast group IPv4 address. Using the ISCLI:

```
RS8296(config)# ip mroute <multicast group IPv4 address> <VLAN> <port>
```

- IGMP Querier is disabled by default. If IGMP Querier is needed with PIM, be sure to enable the IGMP Query feature globally, as well as on each VLAN where it is needed.
- If the switch is connected to multicast receivers and/or hosts, be sure to enable IGMP snooping globally, as well as on each VLAN where PIM receivers are attached.

PIM with VLAG

If using VLAG, see [“VLAG with PIM” on page 187](#).

PIM Configuration Examples

Example 1: PIM-SM with Dynamic RP

This example configures PIM Sparse Mode for one IP interface, with the switch acting as a candidate for dynamic Rendezvous Point (RP) selection.

1. Globally enable the PIM feature:

```
RS8296(config)# ip pim enable
```

2. Configure a PIM network component with dynamic RP settings, and set it for PIM Sparse Mode:

```
RS8296(config)# ip pim component 1
RS8296(config-ip-pim-comp)# mode sparse
RS8296(config-ip-pim-comp)# rp-candidate rp-address 225.1.0.0 255.255.0.0
10.10.1.1
RS8296(config-ip-pim-comp)# rp-candidate holdtime <0-255>
RS8296(config-ip-pim-comp)# exit
```

Where 225.1.0.0 is the multicast group base IP address, 255.255.0.0 is the multicast group address mask, and 10.10.1.1 is the switch RP candidate address.

Note: Because, Sparse Mode is set by default for PIM component 1, the `mode` command is needed only if the mode has been previously changed.

3. Define an IP interface for use with PIM:

```
RS8296(config)# interface ip 111
RS8296(config-ip-if)# ip address 10.10.1.1 255.255.255.255
RS8296(config-ip-if)# vlan 11
RS8296(config-ip-if)# enable
```

The IP interface represents the PIM network being connected to the switch. The IPv4 addresses in the defined range must not be included in another IP interface on the switch under a different VLAN.

4. Enable PIM on the IP interface and assign the PIM component:

```
RS8296(config-ip-if)# ip pim enable
RS8296(config-ip-if)# ip pim component-id 1
```

Note: Because, PIM component 1 is assigned to the interface by default, the `component -id` command is needed only if the setting has been previously changed.

5. Set the Bootstrap Router (BSR) preference:

```
RS8296(config-ip-if)# ip pim cbsr-preference 135
RS8296(config-ip-if)# exit
```

Example 2: PIM-SM with Static RP

The following commands can be used to modify the prior example configuration to use a static RP:

```
RS8296(config)# ip pim static-rp enable
RS8296(config)# ip pim component 1
RS8296(config-ip-pim-comp)# rp-static rp-address 225.1.0.0 255.255.0.0
10.10.1.1
RS8296(config-ip-pim-comp)# exit
```

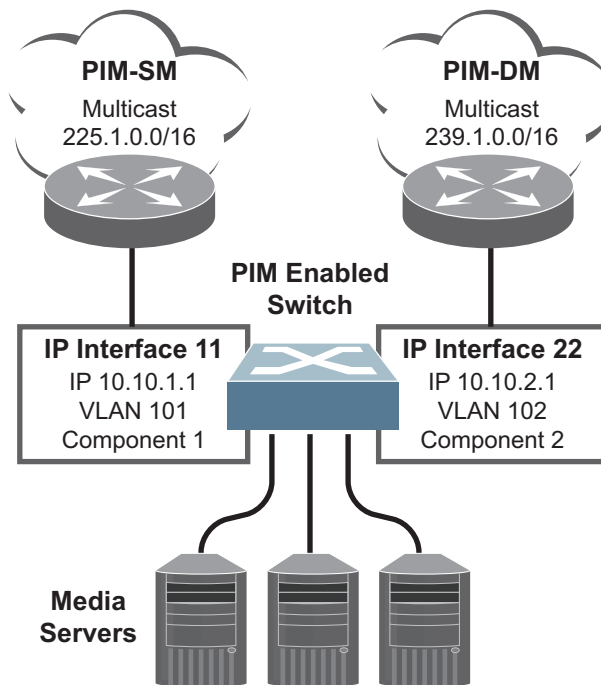
Where 225.1.0.0 255.255.0.0 is the multicast group base address and mask, and 10.10.1.1 is the static RP address.

Note: The same static RP address must be configured for all switches in the group.

Example 3: PIM-DM

This example configures PIM Dense Mode (PIM-DM) on one IP interface. PIM-DM can be configured independently, or it can be combined with the prior PIM-SM examples (which are configured on a different PIM component) as shown in [Figure 47](#).

Figure 47. Network with both PIM-DM and PIM-SM Components



1. Configure the PIM-SM component as shown in the prior examples, or if using PIM-DM independently, enable the PIM feature.

```
RS8296(config)# ip pim enable
```

2. Configure a PIM component and set the PIM mode:

```
RS8296(config)# ip pim component 2
RS8296(config-ip-pim-comp)# mode dense
RS8296(config-ip-pim-comp)# exit
```

3. Define an IP interface for use with PIM:

```
RS8296(config)# interface ip 22
RS8296(config-ip-if)# ip address 10.10.2.1 255.255.255.255
RS8296(config-ip-if)# vlan 102
RS8296(config-ip-if)# enable
```

4. Enable PIM on the IP interface and assign the PIM component:

```
RS8296(config-ip-if)# ip pim enable
RS8296(config-ip-if)# ip pim component-id 2
RS8296(config-ip-if)# exit
```

5. (Optional) Configure PIM border router if the IPMC traffic is flowing between PIM domains:

```
RS8296(config)# ip pim pmbr enable
RS8296(config)# interface ip 22
RS8296(config-ip-if)# ip pim border-bit
RS8296(config-ip-if)# exit
```

Note: For PIM Dense Mode, the DR, RP, and BSR settings do not apply.

Part 6: High Availability Fundamentals

Internet traffic consists of myriad services and applications which use the Internet Protocol (IP) for data delivery. However, IP is not optimized for all the various applications. High Availability goes beyond IP and makes intelligent switching decisions to provide redundant network configurations.

Chapter 34. Basic Redundancy

Lenovo Network Operating System 8.2 includes various features for providing basic link or device redundancy:

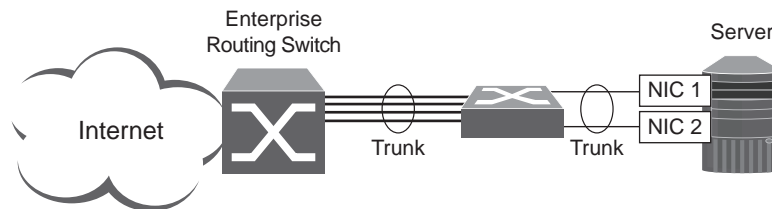
- [“Trunking for Link Redundancy” on page 490](#)
- [“Virtual Link Aggregation” on page 490](#)
- [“Hot Links” on page 491](#)

Trunking for Link Redundancy

Multiple switch ports can be combined together to form robust, high-bandwidth trunks to other devices. Since trunks are comprised of multiple physical links, the trunk group is inherently fault tolerant. As long as one connection between the switches is available, the trunk remains active.

In [Figure 48](#), four ports are trunked together between the switch and the enterprise routing device. Connectivity is maintained as long as one of the links remain active. The links to the server are also trunked, allowing the secondary NIC to take over in the event that the primary NIC link fails.

Figure 48. Trunking Ports for Link Redundancy



For more information on trunking, see [“Ports and Trunking” on page 131](#).

Virtual Link Aggregation

Using the VLAG feature, switches can be paired as VLAG peers. The peer switches appear to the connecting device as a single virtual entity for the purpose of establishing a multi-port trunk. The VLAG-capable switches synchronize their logical view of the access layer port structure and internally prevent implicit loops. The VLAG topology also responds more quickly to link failure and does not result in unnecessary MAC flooding.

VLAGs are useful in multi-layer environments for both uplink and downlink redundancy to any regular LAG-capable device. They can also be used in for active-active VRRP connections.

For more information on VLAGs, see [“Virtual Link Aggregation Groups” on page 167](#).

Hot Links

Hot Links provides basic link redundancy with fast recovery.

Hot Links consists of up to 25 triggers. A trigger consists of a pair of layer 2 interfaces, each containing an individual port, trunk, or LACP adminkey. One interface is the Master, and the other is a Backup. While the Master interface is set to the active state and forwards traffic, the Backup interface is set to the standby state and blocks traffic until the Master interface fails. If the Master interface fails, the Backup interface is set to active and forwards traffic. Once the Master interface is restored, it regains its active state and forwards traffic. The Backup transitions to the standby state.

You may select a physical port, static trunk, or an LACP adminkey as a Hot Link interface.

Forward Delay

The Forward Delay timer allows Hot Links to monitor the Master and Backup interfaces for link stability before selecting one interface to transition to the active state. Before the transition occurs, the interface must maintain a stable link for the duration of the Forward Delay interval.

For example, if you set the Forward delay timer to 10 seconds, the switch will select an interface to become active only if a link remained stable for the duration of the Forward Delay period. If the link is unstable, the Forward Delay period starts again.

Preemption

You can configure the Master interface to resume the active state whenever it becomes available. With Hot Links preemption enabled, the Master interface transitions to the active state immediately upon recovery. The Backup interface immediately transitions to the standby state. If Forward Delay is enabled, the transition occurs when an interface has maintained link stability for the duration of the Forward Delay period.

FDB Update

Use the FDB update option to notify other devices on the network about updates to the Forwarding Database (FDB). When you enable FDB update, the switch sends multicasts of addresses in the forwarding database (FDB) over the active interface, so that other devices on the network can learn the new path. The Hot Links FDB update option uses the station update rate to determine the rate at which to send FDB packets.

Configuration Guidelines

The following configuration guidelines apply to Hot links:

- When Hot Links is turned on, STP must be turned off on the Hot Links ports.

- A port that is a member of the Master interface cannot be a member of the Backup interface. A port that is a member of one Hot Links trigger cannot be a member of another Hot Links trigger.
- An individual port that is configured as a Hot Link interface cannot be a member of a trunk.

Configuring Hot Links

Use the following commands to configure Hot Links.

```
RS8296(config)# hotlinks trigger 1 enable (Enable Hot Links Trigger 1)  
RS8296(config)# hotlinks trigger 1 master port 1 (Add port to Master interface)  
RS8296(config)# hotlinks trigger 1 backup port 2 (Add port to Backup interface)  
RS8296(config)# hotlinks enable (Turn on Hot Links)
```

Chapter 35. Layer 2 Failover

The primary application for Layer 2 Failover is to support Network Adapter Teaming. With Network Adapter Teaming, all the NICs on each server share the same IP address, and are configured into a team. One NIC is the primary link, and the other is a standby link. For more details, refer to the documentation for your Ethernet adapter.

Note: Only two links per server can be used for Layer 2 Trunk Failover (one primary and one backup). Network Adapter Teaming allows only one backup NIC for each server blade.

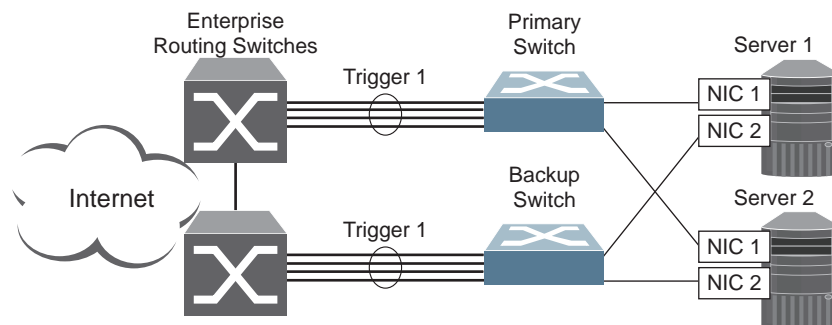
Monitoring Trunk Links

Layer 2 Failover can be enabled on any trunk group in the G8296, including LACP trunks. Trunks can be added to failover trigger groups. Then, if some specified number of monitor links fail, the switch disables all the control ports in the switch. When the control ports are disabled, it causes the NIC team on the affected servers to failover from the primary to the backup NIC. This process is called a failover event.

When the appropriate number of links in a monitor group return to service, the switch enables the control ports. This causes the NIC team on the affected servers to fail back to the primary switch (unless Auto-Fallback is disabled on the NIC team). The backup switch processes traffic until the primary switch's control links come up, which can take up to five seconds.

Figure 49 is a simple example of Layer 2 Failover. One G8296 is the primary, and the other is used as a backup. In this example, all ports on the primary switch belong to a single trunk group, with Layer 2 Failover enabled, and Failover Limit set to 2. If two or fewer links in trigger 1 remain active, the switch temporarily disables all control ports. This action causes a failover event on Server 1 and Server 2.

Figure 49. Basic Layer 2 Failover



Setting the Failover Limit

The failover limit lets you specify the minimum number of operational links required within each trigger before the trigger initiates a failover event. For example, if the limit is two, a failover event occurs when the number of operational links in the trigger is two or fewer. When you set the limit to zero, the switch triggers a failover event only when no links in the trigger are operational.

Manually Monitoring Port Links

The Manual Monitor allows you to configure a set of ports and trunks to monitor for link failures (a monitor list), and another set of ports and trunks to disable when the trigger limit is reached (a control list). When the switch detects a link failure on the monitor list, it automatically disables the items in control list. When server ports are disabled, the corresponding server's network adapter can detect the disabled link, and trigger a network-adapter failover to another port or trunk on the switch, or another switch.

The switch automatically enables the control list items when the monitor list items return to service.

Monitor Port State

A monitor port is considered operational as long as the following conditions are true:

- The port must be in the `Link Up` state.
- If STP is enabled, the port must be in the `Forwarding` state.
- If the port is part of an LACP trunk, the port must be in the `Aggregated` state.

If any of these conditions is false, the monitor port is considered to have failed.

Control Port State

A control port is considered Operational if the monitor trigger is up. As long as the trigger is up, the port is considered operational from a teaming perspective, even if the port itself is actually in the `Down` state, `Blocking` state (if STP is enabled on the port), or `Not Aggregated` state (if part of an LACP trunk).

A control port is considered to have failed only if the monitor trigger is in the `Down` state.

To view the state of any port, use one of the following commands:

```
>> # show interface link                (View port link status)
>> # show interface port <x> spanning-tree stp <x> (View port STP status)
>> # show lacp information              (View port LACP status)
```

L2 Failover with Other Features

L2 Failover works together with static trunks, Link Aggregation Control Protocol (LACP), and with Spanning Tree Protocol (STP), as described in the next sections.

Static Trunks

When you add a portchannel (static trunk group) to a failover trigger, any ports in that trunk become members of the trigger. You can add up to 64 static trunks to a failover trigger, using manual monitoring.

LACP

Link Aggregation Control Protocol allows the switch to form dynamic trunks. You can use the *admin key* to add up to two LACP trunks to a failover trigger using automatic monitoring. When you add an *admin key* to a trigger, any LACP trunk with that *admin key* becomes a member of the trigger.

Spanning Tree Protocol

If Spanning Tree Protocol (STP) is enabled on the ports in a failover trigger, the switch monitors the port STP state rather than the link state. A port failure results when STP is not in a Forwarding state (such as Learning, Discarding, or No Link) in all the Spanning Tree Groups (STGs) to which the port belongs. The switch automatically disables the appropriate control ports.

When the switch determines that ports in the trigger are in STP Forwarding state in any one of the STGs it belongs to, then it automatically enables the appropriate control ports. The switch *fails back* to normal operation.

For example, if a monitor port is a member of STG1, STG2, and STG3, a failover will be triggered only if the port is not in a forwarding state in all the three STGs. When the port state in any of the three STGs changes to forwarding, then the control port is enabled and normal switch operation is resumed.

Configuration Guidelines

This section provides important information about configuring Layer 2 Failover.

- Any specific failover trigger can monitor ports only, static trunks only, or LACP trunks only. The different types cannot be combined in the same trigger.
- A maximum of 64 LACP keys can be added per trigger.
- Management ports cannot be monitored.
- Control ports for different triggers must not overlap. Monitor ports may overlap.

Configuring Layer 2 Failover

Use the following procedure to configure a Layer 2 Failover Manual Monitor.

1. Specify the links to monitor.

```
RS8296(config)# failover trigger 1 mmon monitor member 1-5
```

2. Specify the links to disable when the failover limit is reached.

```
RS8296(config)# failover trigger 1 mmon control member 6-10
```

3. Configure general Failover parameters.

```
RS8296(config)# failover enable  
RS8296(config)# failover trigger 1 enable  
RS8296(config)# failover trigger 1 limit 2
```

Chapter 36. Virtual Router Redundancy Protocol

The RackSwitch G8296 (G8296) supports IPv4 high-availability network topologies through an enhanced implementation of the Virtual Router Redundancy Protocol (VRRP).

Note: Lenovo Network Operating System 8.2 does not support IPv6 for VRRP.

The following topics are discussed in this chapter:

- [“VRRP Overview” on page 500](#). This section discusses VRRP operation and Networking OS redundancy configurations.
- [“Failover Methods” on page 503](#). This section describes the three modes of high availability.
- [“Networking OS Extensions to VRRP” on page 504](#). This section describes VRRP enhancements implemented in N/OS.
- [“Virtual Router Deployment Considerations” on page 505](#). This section describes issues to consider when deploying virtual routers.
- [“High Availability Configurations” on page 507](#). This section discusses the more useful and easily deployed redundant configurations.

VRRP Overview

In a high-availability network topology, no device can create a single point-of-failure for the network or force a single point-of-failure to any other part of the network. This means that your network will remain in service despite the failure of any single device. To achieve this usually requires redundancy for all vital network components.

VRRP enables redundant router configurations within a LAN, providing alternate router paths for a host to eliminate single points-of-failure within a network. Each participating VRRP-capable routing device is configured with the same virtual router IPv4 address and ID number. One of the virtual routers is elected as the master, based on a number of priority criteria, and assumes control of the shared virtual router IPv4 address. If the master fails, one of the backup virtual routers will take control of the virtual router IPv4 address and actively process traffic addressed to it.

With VRRP, Virtual Interface Routers (VIR) allow two VRRP routers to share an IP interface across the routers. VIRs provide a single Destination IPv4 (DIP) address for upstream routers to reach various servers, and provide a virtual default Gateway for the servers.

VRRP Components

Each physical router running VRRP is known as a *VRRP router*.

Virtual Router

Two or more VRRP routers can be configured to form a *virtual router* (RFC 2338). Each VRRP router may participate in one or more virtual routers. Each virtual router consists of a user-configured *virtual router identifier* (VRID) and an IPv4 address.

Virtual Router MAC Address

The VRID is used to build the *virtual router MAC Address*. The five highest-order octets of the virtual router MAC Address are the standard MAC prefix (00-00-5E-00-01) defined in RFC 2338. The VRID is used to form the lowest-order octet.

Owners and Renters

Only one of the VRRP routers in a virtual router may be configured as the IPv4 address owner. This router has the virtual router's IPv4 address as its real interface address. This router responds to packets addressed to the virtual router's IPv4 address for ICMP pings, TCP connections, and so on.

There is no requirement for any VRRP router to be the IPv4 address owner. Most VRRP installations choose not to implement an IPv4 address owner. For the purposes of this chapter, VRRP routers that are not the IPv4 address owner are called *renters*.

Master and Backup Virtual Router

Within each virtual router, one VRRP router is selected to be the virtual router master. See [“Selecting the Master VRRP Router” on page 502](#) for an explanation of the selection process.

Note: If the IPv4 address owner is available, it will always become the virtual router master.

The virtual router master forwards packets sent to the virtual router. It also responds to Address Resolution Protocol (ARP) requests sent to the virtual router's IPv4 address. Finally, the virtual router master sends out periodic advertisements to let other VRRP routers know it is alive and its priority.

Within a virtual router, the VRRP routers not selected to be the master are known as virtual router backups. If the virtual router master fails, one of the virtual router backups becomes the master and assumes its responsibilities.

Virtual Interface Router

At Layer 3, a Virtual Interface Router (VIR) allows two VRRP routers to share an IP interface across the routers. VIRs provide a single Destination IPv4 (DIP) address for upstream routers to reach various destination networks, and provide a virtual default Gateway.

Note: Every VIR must be assigned to an IP interface, and every IP interface must be assigned to a VLAN. If no port in a VLAN has link up, the IP interface of that VLAN is down, and if the IP interface of a VIR is down, that VIR goes into INIT state.

VRRP Operation

Only the virtual router master responds to ARP requests. Therefore, the upstream routers only forward packets destined to the master. The master also responds to ICMP ping requests. The backup does not forward any traffic, nor does it respond to ARP requests.

If the master is not available, the backup becomes the master and takes over responsibility for packet forwarding and responding to ARP requests.

Selecting the Master VRRP Router

Each VRRP router is configured with a priority between 1–254. A bidding process determines which VRRP router is or becomes the master—the VRRP router with the highest priority.

The master periodically sends advertisements to an IPv4 multicast address. As long as the backups receive these advertisements, they remain in the backup state. If a backup does not receive an advertisement for three advertisement intervals, it initiates a bidding process to determine which VRRP router has the highest priority and takes over as master. In addition to the three advertisement intervals, a manually set holdoff time can further delay the backups from assuming the master status.

If, at any time, a backup determines that it has higher priority than the current master does, it can preempt the master and become the master itself, unless configured not to do so. In preemption, the backup assumes the role of master and begins to send its own advertisements. The current master sees that the backup has higher priority and will stop functioning as the master.

A backup router can stop receiving advertisements for one of two reasons—the master can be down, or all communications links between the master and the backup can be down. If the master has failed, it is clearly desirable for the backup (or one of the backups, if there is more than one) to become the master.

Note: If the master is healthy but communication between the master and the backup has failed, there will then be two masters within the virtual router. To prevent this from happening, configure redundant links to be used between the switches that form a virtual router.

Failover Methods

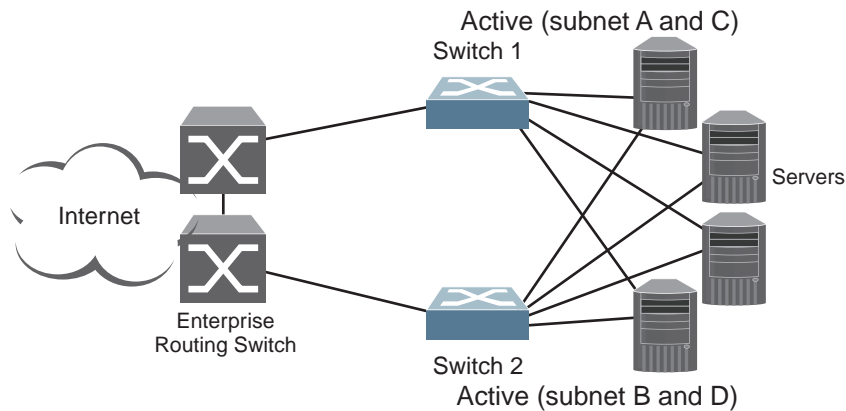
With service availability becoming a major concern on the Internet, service providers are increasingly deploying Internet traffic control devices, such as application switches, in redundant configurations. N/OS high availability configurations are based on VRRP. The N/OS implementation of VRRP includes proprietary extensions.

Active-Active Redundancy

In an active-active configuration, shown in [Figure 50](#), two switches provide redundancy for each other, with both active at the same time. Each switch processes traffic on a different subnet. When a failure occurs, the remaining switch can process traffic on all subnets.

For a configuration example, see [“High Availability Configurations” on page 507](#).

Figure 50. Active-Active Redundancy



Virtual Router Group

The virtual router group ties all virtual routers on the switch together as a single entity. As members of a group, all virtual routers on the switch (and therefore the switch itself), are in either a master or standby state.

A VRRP group has the following characteristics:

- When enabled, all virtual routers behave as one entity, and all group settings override any individual virtual router settings.
- All individual virtual routers, once the VRRP group is enabled, assume the group's tracking and priority.
- When one member of a VRRP group fails, the priority of the group decreases, and the state of the entire switch changes from Master to Standby.

Each VRRP advertisement can include up to 16 addresses. All virtual routers are advertised within the same packet, conserving processing and buffering resources.

Networking OS Extensions to VRRP

This section describes VRRP enhancements that are implemented in N/OS.

N/OS supports a tracking function that dynamically modifies the priority of a VRRP router, based on its current state. The objective of tracking is to have, whenever possible, the master bidding processes for various virtual routers in a LAN converge on the same switch. Tracking ensures that the selected switch is the one that offers optimal network performance. For tracking to have any effect on virtual router operation, preemption must be enabled.

N/OS can track the attributes listed in [Table 42](#) (Router VRRP mode):

Table 42. VRRP Tracking Parameters

Parameter	Description
Number of IP interfaces on the switch that are active (“up”) <code>tracking-priority-increment interfaces</code>	Helps elect the virtual routers with the most available routes as the master. (An IP interface is considered active when there is at least one active port on the same VLAN.) This parameter influences the VRRP router's priority in virtual interface routers.
Number of active ports on the same VLAN <code>tracking-priority-increment ports</code>	Helps elect the virtual routers with the most available ports as the master. This parameter influences the VRRP router's priority in virtual interface routers.
Number of virtual routers in master mode on the switch <code>tracking-priority-increment virtual-routers</code>	Useful for ensuring that traffic for any particular client/server pair is handled by the same switch, increasing routing efficiency. This parameter influences the VRRP router's priority in virtual interface routers.
Next hop tracking <code>virtual-router <x> next-hop <IP address> [ICMP ARP] [<interval>] [<retries>]</code>	A health check mechanism that is used to track the next hop. This parameter influences the VRRP router's priority based on the availability of the next hop.

Each tracked parameter has a user-configurable weight associated with it. As the count associated with each tracked item increases (or decreases), so does the VRRP router's priority, subject to the weighting associated with each tracked item. If the priority level of a standby is greater than that of the current master, then the standby can assume the role of the master.

See [“Configuring the Switch for Tracking” on page 505](#) for an example on how to configure the switch for tracking VRRP priority.

Virtual Router Deployment Considerations

Assigning VRRP Virtual Router ID

During the software upgrade process, VRRP virtual router IDs will be automatically assigned if failover is enabled on the switch. When configuring virtual routers at any point after upgrade, virtual router ID numbers must be assigned. The virtual router ID may be configured as any number between 1 and 128. Use the following command to configure the virtual router ID:

```
RS8296(config)# router vrrp
RS8296(config-vrrp)# virtual-router 1 virtual-router-id <1-128>
```

Configuring the Switch for Tracking

Tracking configuration largely depends on user preferences and network environment. Consider the configuration shown in [Figure 50 on page 503](#). Assume the following behavior on the network:

- Switch 1 is the master router upon initialization.
- If switch 1 is the master and it has one fewer active servers than switch 2, then switch 1 remains the master.
This behavior is preferred because running one server down is less disruptive than bringing a new master online and severing all active connections in the process.
- If switch 1 is the master and it has two or more active servers fewer than switch 2, then switch 2 becomes the master.
- If switch 2 is the master, it remains the master even if servers are restored on switch 1 such that it has one fewer or an equal number of servers.
- If switch 2 is the master and it has one active server fewer than switch 1, then switch 1 becomes the master.

You can implement this behavior by configuring the switch for tracking as follows:

1. Set the priority for switch 1 to 101.
2. Leave the priority for switch 2 at the default value of 100.
3. On both switches, enable tracking based on ports, interfaces, next hops, or virtual routers. You can choose any combination of tracking parameters, based on your network configuration.

Note: There is no shortcut to setting tracking parameters. The goals must first be set and the outcomes of various configurations and scenarios analyzed to find settings that meet the goals.

VRRP - Next Hop Tracking

VRRP can be configured to track next hops. A health check mechanism, using ICMP ping or ARP requests, is used to track the next hop. If the health check succeeds or fails, the priority of the virtual router, for which the next hop tracking was configured, changes based on the configured tracking-priority-increment value.

A VRRP group's priority is calculated as the sum of all priorities for all active next hops. If an active next hop belongs to two virtual routers, then that next-hop's priority is added twice.

VRRP next hop tracking can be configured using the following commands:

Enable VRRP tracking on next hop:

```
RS8296(config-vrrp)# virtual-router <x> track next-hops
```

Configure next hop IP address:

```
RS8296(config-vrrp)# virtual-router <x> next-hop <IP address> [ICMP|ARP]
[<interval>] [<retries>]
```

Default Values:
Health check protocol: ICMP
Interval: 2 seconds
Retries: 3

Note: A maximum of four unique next-hop IP addresses can be configured.

Configure priority for each next hop.

The value is added to the virtual router's priority when the next hop is active, and subtracted, when the next hop goes down:

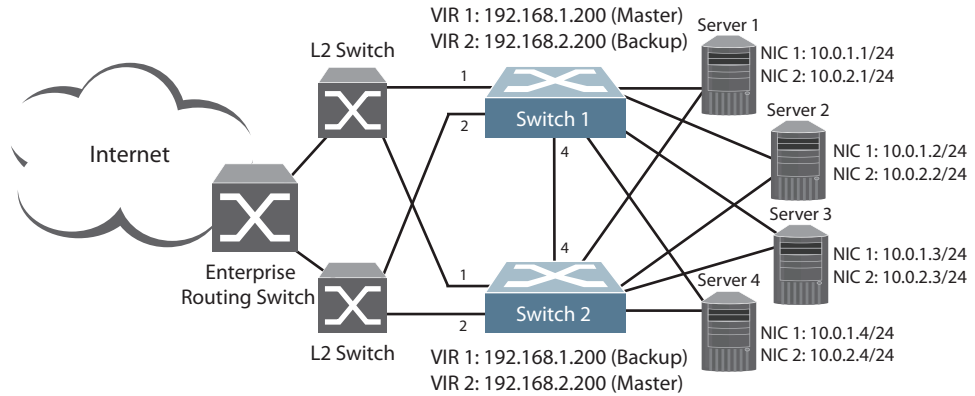
```
RS8296(config-vrrp)# tracking-priority-increment next-hop <0-254>
```

High Availability Configurations

VRRP High-Availability Using Multiple VIRs

Figure 51 shows an example configuration where two G8296s are used as VRRP routers in an active-active configuration. In this configuration, both switches respond to packets.

Figure 51. Active-Active Configuration using VRRP



Although this example shows only two switches, there is no limit on the number of switches used in a redundant configuration. It is possible to implement an active-active configuration across all the VRRP-capable switches in a LAN.

Each VRRP-capable switch in an active-active configuration is autonomous. Switches in a virtual router need not be identically configured.

In the scenario illustrated in Figure 51, traffic destined for IPv4 address 10.0.1.1 is forwarded through the Layer 2 switch at the top of the drawing, and ingresses G8296 1 on port 1. Return traffic uses default gateway 1 (192.168.1.1), which must be in the same network.

If the link between G8296 1 and the Layer 2 switch fails, G8296 2 becomes the Master because it has a higher priority. Traffic is forwarded to G8296 2, which forwards it to G8296 1 through port 4. Return traffic uses default gateway 2 (192.168.2.2), and is forwarded through the Layer 2 switch at the bottom of the drawing.

To implement the active-active example, perform the following switch configuration.

Task 1: Configure G8296 1

1. Configure client and server interfaces.

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip address 192.168.1.100 255.255.255.0
RS8296(config-ip-if)# vlan 10
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip address 192.168.2.101 255.255.255.0
RS8296(config-ip-if)# vlan 20
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 3
RS8296(config-ip-if)# ip address 10.0.1.100 255.255.255.0
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 4
RS8296(config-ip-if)# ip address 10.0.2.101 255.255.255.0
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
```

2. Configure the default gateways. Each default gateway points to a Layer 3 router.

```
RS8296(config)# ip gateway 1 address 192.168.1.1
RS8296(config)# ip gateway 1 enable
RS8296(config)# ip gateway 2 address 192.168.2.1
RS8296(config)# ip gateway 2 enable
```

3. Turn on VRRP and configure two Virtual Interface Routers.

```
RS8296(config)# router vrrp
RS8296(config-vrrp)# enable
RS8296(config-vrrp)# virtual-router 1 virtual-router-id 1
RS8296(config-vrrp)# virtual-router 1 interface 1
RS8296(config-vrrp)# virtual-router 1 address 192.168.1.200
RS8296(config-vrrp)# virtual-router 1 enable
RS8296(config-vrrp)# virtual-router 2 virtual-router-id 2
RS8296(config-vrrp)# virtual-router 2 interface 2
RS8296(config-vrrp)# virtual-router 2 address 192.168.2.200
RS8296(config-vrrp)# virtual-router 2 enable
```

4. Enable tracking on ports. Set the priority of Virtual Router 1 to 101, so that it becomes the Master.

```
RS8296(config-vrrp)# virtual-router 1 track ports
RS8296(config-vrrp)# virtual-router 1 priority 101
RS8296(config-vrrp)# virtual-router 2 track ports
RS8296(config-vrrp)# exit
```

5. Configure ports.

```
RS8296(config)# vlan 10
RS8296(config-vlan)# exit
RS8296(config)# interface port 1
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 10
RS8296(config-if)# exit

RS8296(config)# vlan 20
RS8296(config-vlan)# exit
RS8296(config)# interface port 2
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 20
RS8296(config-if)# exit
```

6. Turn off Spanning Tree Protocol globally.

```
RS8296(config)# no spanning-tree stp 1
```

Task 2: Configure G8296 2

1. Configure client and server interfaces.

```
RS8296(config)# interface ip 1
RS8296(config-ip-if)# ip address 192.168.1.101 255.255.255.0
RS8296(config-ip-if)# vlan 10
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 2
RS8296(config-ip-if)# ip address 192.168.2.100 255.255.255.0
RS8296(config-ip-if)# vlan 20
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 3
RS8296(config-ip-if)# ip address 10.0.1.101 255.255.255.0
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
RS8296(config)# interface ip 4
RS8296(config-ip-if)# ip address 10.0.2.100 255.255.255.0
RS8296(config-ip-if)# enable
RS8296(config-ip-if)# exit
```

2. Configure the default gateways. Each default gateway points to a Layer 3 router.

```
RS8296(config)# ip gateway 1 address 192.168.2.1
RS8296(config)# ip gateway 1 enable
RS8296(config)# ip gateway 2 address 192.168.1.1
RS8296(config)# ip gateway 2 enable
```

3. Turn on VRRP and configure two Virtual Interface Routers.

```
RS8296(config)# router vrrp
RS8296(config-vrrp)# enable
RS8296(config-vrrp)# virtual-router 1 virtual-router-id 1
RS8296(config-vrrp)# virtual-router 1 interface 1
RS8296(config-vrrp)# virtual-router 1 address 192.168.1.200
RS8296(config-vrrp)# virtual-router 1 enable
RS8296(config-vrrp)# virtual-router 2 virtual-router-id 2
RS8296(config-vrrp)# virtual-router 2 interface 2
RS8296(config-vrrp)# virtual-router 2 address 192.168.2.200
RS8296(config-vrrp)# virtual-router 2 enable
```

4. Enable tracking on ports. Set the priority of Virtual Router 2 to 101, so that it becomes the Master.

```
RS8296(config-vrrp)# virtual-router 1 track ports
RS8296(config-vrrp)# virtual-router 2 track ports
RS8296(config-vrrp)# virtual-router 2 priority 101
RS8296(config-vrrp)# exit
```

5. Configure ports.

```
RS8296(config)# vlan 10
RS8296(config-vlan)# exit
RS8296(config)# interface port 1
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 10
RS8296(config-if)# exit

RS8296(config)# vlan 20
RS8296(config-vlan)# exit
RS8296(config)# interface port 2
RS8296(config-if)# switchport mode trunk
RS8296(config-if)# switchport trunk allowed vlan add 20
RS8296(config-if)# exit
```

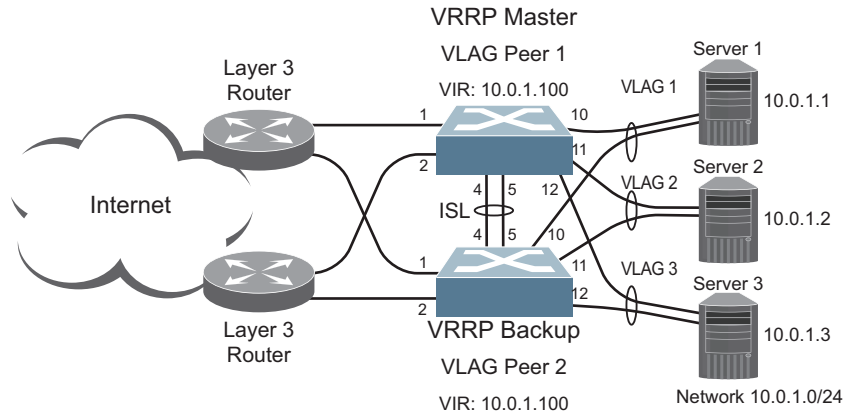
6. Turn off Spanning Tree Protocol globally.

```
RS8296(config)# no spanning-tree stp 1
```

VRRP High-Availability Using VLAGs

VRRP can be used in conjunction with VLAGs and LACP-capable servers and switches to provide seamless redundancy.

Figure 52. Active-Active Configuration using VRRP and VLAGs



See [“VLAGs with VRRP”](#) on page 178 for a detailed configuration example.

Part 7: Network Management

Chapter 37. Link Layer Discovery Protocol

The Lenovo Network Operating System software support Link Layer Discovery Protocol (LLDP). This chapter discusses the use and configuration of LLDP on the switch:

- [“LLDP Overview” on page 516](#)
- [“Enabling or Disabling LLDP” on page 517](#)
- [“LLDP Transmit Features” on page 518](#)
- [“LLDP Receive Features” on page 522](#)
- [“LLDP Example Configuration” on page 526](#)

LLDP Overview

Link Layer Discovery Protocol (LLDP) is an IEEE 802.1AB-2005 standard for discovering and managing network devices. LLDP uses Layer 2 (the data link layer), and allows network management applications to extend their awareness of the network by discovering devices that are direct neighbors of already known devices.

With LLDP, the G8296 can advertise the presence of its ports, their major capabilities, and their current status to other LLDP stations in the same LAN. LLDP transmissions occur on ports at regular intervals or whenever there is a relevant change to their status. The switch can also receive LLDP information advertised from adjacent LLDP-capable network devices.

In addition to discovery of network resources, and notification of network changes, LLDP can help administrators quickly recognize a variety of common network configuration problems, such as unintended VLAN exclusions or mis-matched port aggregation membership.

The LLDP transmit function and receive function can be independently configured on a per-port basis. The administrator can allow any given port to transmit only, receive only, or both transmit and receive LLDP information.

The LLDP information to be distributed by the G8296 ports, and that which has been collected from other LLDP stations, is stored in the switch's Management Information Base (MIB). Network Management Systems (NMS) can use Simple Network Management Protocol (SNMP) to access this MIB information. LLDP-related MIB information is read-only.

Changes, either to the local switch LLDP information or to the remotely received LLDP information, are flagged within the MIB for convenient tracking by SNMP-based management systems.

For LLDP to provide expected benefits, all network devices that support LLDP must be consistent in their LLDP configuration.

Enabling or Disabling LLDP

Global LLDP Setting

By default, LLDP is enabled on the G8296. To turn LLDP on or off, use the following command:

```
RS8296(config)# [no] lldp enable      (Turn LLDP on or off globally)
```

Transmit and Receive Control

The G8296 can also be configured to transmit or receive LLDP information on a port-by-port basis. By default, when LLDP is globally enabled on the switch, G8296 ports transmit and receive LLDP information (see the `tx_rx` option in the following example). To change the LLDP transmit and receive state, the following commands are available:

```
RS8296(config)# interface port 1      (Select a switch port)
RS8296(config-if)# lldp admin-status tx_rx (Transmit and receive LLDP)
RS8296(config-if)# lldp admin-status tx_only (Only transmit LLDP)
RS8296(config-if)# lldp admin-status rx_only (Only receive LLDP)
RS8296(config-if)# no lldp admin-status (Do not participate in LLDP)
RS8296(config-if)# exit (Exit port mode)
```

To view the LLDP transmit and receive status, use the following commands:

```
RS8296(config)# show lldp port (status of all ports)
RS8296(config)# show interface port <n> lldp (status of selected port)
```

LLDP Transmit Features

Numerous LLDP transmit options are available, including scheduled and minimum transmit interval, expiration on remote systems, SNMP trap notification, and the types of information permitted to be shared.

Scheduled Interval

The G8296 can be configured to transmit LLDP information to neighboring devices once each 5 to 32768 seconds. The scheduled interval is global; the same interval value applies to all LLDP transmit-enabled ports. However, to help balance LLDP transmissions and keep them from being sent simultaneously on all ports, each port maintains its own interval clock, based on its own initialization or reset time. This allows switch-wide LLDP transmissions to be spread out over time, though individual ports comply with the configured interval.

The global transmit interval can be configured using the following command:

```
RS8296(config)# lldp refresh-interval <interval>
```

where *interval* is the number of seconds between LLDP transmissions. The range is 5 to 32768. The default is 30 seconds.

Minimum Interval

In addition to sending LLDP information at scheduled intervals, LLDP information is also sent when the G8296 detects relevant changes to its configuration or status (such as when ports are enabled or disabled). To prevent the G8296 from sending multiple LLDP packets in rapid succession when port status is in flux, a transmit delay timer can be configured.

The transmit delay timer represents the minimum time permitted between successive LLDP transmissions on a port. Any interval-driven or change-driven updates will be consolidated until the configured transmit delay expires.

The minimum transmit interval can be configured using the following command:

```
RS8296(config)# lldp transmission-delay <interval>
```

where *interval* is the minimum number of seconds permitted between successive LLDP transmissions on any port. The range is 1 to one-quarter of the scheduled transmit interval (`lldp refresh-interval <value>`), up to 8192. The default is 2 seconds.

Time-to-Live for Transmitted Information

The transmitted LLDP information is held by remote systems for a limited time. A time-to-live parameter allows the switch to determine how long the transmitted data is held before it expires. The hold time is configured as a multiple of the configured transmission interval.

```
RS8296(config)# lldp holdtime-multiplier <multiplier>
```

where *multiplier* is a value between 2 and 10. The default value is 4, meaning that remote systems will hold the port's LLDP information for 4 x the 30-second transmission interval value, or 120 seconds, before removing it from their MIB.

Trap Notifications

If SNMP is enabled on the G8296 (see [“Using Simple Network Management Protocol” on page 37](#)), each port can be configured to send SNMP trap notifications whenever LLDP transmissions are sent. By default, trap notification is disabled for each port. The trap notification state can be changed using the following commands (Interface Port mode):

```
RS8296(config)# interface port 1
RS8296(config-if)# [no] lldp trap-notification
RS8296(config-if)# exit
```

In addition to sending LLDP information at scheduled intervals, LLDP information is also sent when the G8296 detects relevant changes to its configuration or status (such as when ports are enabled or disabled). To prevent the G8296 from sending multiple trap notifications in rapid succession when port status is in flux, a global trap delay timer can be configured.

The trap delay timer represents the minimum time permitted between successive trap notifications on any port. Any interval-driven or change-driven trap notices from the port will be consolidated until the configured trap delay expires.

The minimum trap notification interval can be configured using the following command:

```
RS8296(config)# lldp trap-notification-interval <interval>
```

where *interval* is the minimum number of seconds permitted between successive LLDP transmissions on any port. The range is 1 to 3600. The default is 5 seconds.

If SNMP trap notification is enabled, the notification messages can also appear in the system log. This is enabled by default. To change whether the SNMP trap notifications for LLDP events appear in the system log, use the following command:

```
RS8296(config)# [no] logging log lldp
```

Changing the LLDP Transmit State

When the port is disabled, or when LLDP transmit is turned off for the port using the LLDP admin-status command options (see [“Transmit and Receive Control” on page 517](#)), a final LLDP packet is transmitted with a time-to-live value of 0. Neighbors that receive this packet will remove the LLDP information associated with the G8296 port from their MIB.

In addition, if LLDP is fully disabled on a port and then later re-enabled, the G8296 will temporarily delay resuming LLDP transmissions on the port to allow the port LLDP information to stabilize. The reinitialization delay interval can be globally configured for all ports using the following command:

```
RS8296(config)# lldp reinit-delay <interval>
```

where *interval* is the number of seconds to wait before resuming LLDP transmissions. The range is between 1 and 10. The default is 2 seconds.

Types of Information Transmitted

When LLDP transmission is permitted on the port (see [“Enabling or Disabling LLDP” on page 517](#)), the port advertises the following required information in type/length/value (TLV) format:

- Chassis ID
- Port ID
- LLDP Time-to-Live

LLDP transmissions can also be configured to enable or disable inclusion of optional information, using the following command (Interface Port mode):

```
RS8296(config)# interface port 1
RS8296(config-if)# [no] lldp tlv <type>
RS8296(config-if)# exit
```

where *type* is an LLDP information option from [Table 43](#):

Table 43. LLDP Optional Information Types

Type	Description	Default
portdesc	Port Description	Enabled
sysname	System Name	Enabled
sysdescr	System Description	Enabled
syscap	System Capabilities	Enabled
mgmtaddr	Management Address	Enabled
portvid	IEEE 802.1 Port VLAN ID	Disabled
portprot	IEEE 802.1 Port and Protocol VLAN ID	Disabled
vlanname	IEEE 802.1 VLAN Name	Disabled
protid	IEEE 802.1 Protocol Identity	Disabled
macphy	IEEE 802.3 MAC/PHY Configuration/Status, including the auto-negotiation, duplex, and speed status of the port.	Disabled

Table 43. *LLDP Optional Information Types (continued)*

Type	Description	Default
powermdi	IEEE 802.3 Power via MDI, indicating the capabilities and status of devices that require or provide power over twisted-pair copper links.	Disabled
linkaggr	IEEE 802.3 Link Aggregation status for the port.	Disabled
framesz	IEEE 802.3 Maximum Frame Size for the port.	Disabled
all	Select all optional LLDP information for inclusion or exclusion.	Disabled

LLDP Receive Features

Types of Information Received

When the LLDP receive option is enabled on a port (see [“Enabling or Disabling LLDP” on page 517](#)), the port may receive the following information from LLDP-capable remote systems:

- Chassis Information
- Port Information
- LLDP Time-to-Live
- Port Description
- System Name
- System Description
- System Capabilities Supported/Enabled
- Remote Management Address

The G8296 stores the collected LLDP information in the MIB. Each remote LLDP-capable device is responsible for transmitting regular LLDP updates. If the received updates contain LLDP information changes (to port state, configuration, LLDP MIB structures, deletion), the switch will set a change flag within the MIB for convenient notification to SNMP-based management systems.

Viewing Remote Device Information

LLDP information collected from neighboring systems can be viewed in numerous ways:

- Using a centrally-connected LLDP analysis server
- Using an SNMP agent to examine the G8296 MIB
- Using the G8296 Browser-Based Interface (BBI)
- Using commands on the G8296

The following command displays remote LLDP information:

```
RS8296(config)# show lldp remote-device [<index number>]
```

To view a summary of remote information, omit the *Index number* parameter. For example:

```
RS8296(config)# show lldp remote-device
LLDP Remote Devices Information
Legend(possible values in DMAC column) :
NB - Nearest Bridge - 01-80-C2-00-00-0E
NnTB - Nearest non-TPMR Bridge - 01-80-C2-00-00-03
NCB - Nearest Customer Bridge - 01-80-C2-00-00-00
Total number of current entries: 2

LocalPort|Index|Remote Chassis ID |Remote Port|Remote System Name|DMAC
-----|-----|-----|-----|-----|-----
MGT | 1 | 00 00 00 00 11 00 | 5 | C12 | NB
20 | 2 | 74 99 75 1c 75 00 | 44 | C3+ | NB
```

To view detailed information for a remote device, specify the *Index number* as found in the summary. For example, in keeping with the sample summary, to list details for the first remote device (with an Index value of 1), use the following command:

```
RS8296(config)# show lldp remote-device 1
Local Port Alias: 3
  Remote Device Index      : 1
  Remote Device TTL       : 99
  Remote Device RxChanges : false
  Chassis Type            : Mac Address
  Chassis Id              : 00-18-b1-33-1d-00
  Port Type               : Locally Assigned
  Port Id                 : 23
  Port Description        : 7

  System Name             :
  System Description      : Lenovo Networking Operating
System RackSwitch G8296, Lenovo Networking OS: version 8.2.1, Boot Image:
version 8.2.1

  System Capabilities Supported : bridge, router
  System Capabilities Enabled  : bridge, router

  Remote Management Address:
    Subtype                 : IPV4
    Address                  : 10.100.120.181
    Interface Subtype       : ifIndex
    Interface Number        : 128
    Object Identifier       :
```

Note: Received LLDP information can change very quickly. When using show commands, it is possible that flags for some expected events may be too short-lived to be observed in the output.

To view detailed information of all remote devices, use the following command:

```
RS8296# show lldp remote-device detail

Local Port Alias: MGTA
  Remote Device Index      : 1
  Remote Device TTL       : 4678
  Remote Device RxChanges : false
  Chassis Type            : Mac Address
  Chassis Id              : 08-17-f4-a1-db-00
  Port Type               : Locally Assigned
  Port Id                 : 25
  Port Description        : MGTA

  System Name             :
  System Description      : Lenovo Networking Operating
System RackSwitch G8296, Lenovo Networking OS: version 8.2.1, Boot Image:
version 8.2.1
  System Capabilities Supported : bridge, router
  System Capabilities Enabled   : bridge, router

  Remote Management Address:
    Subtype                  : IPv4
    Address                   : 10.38.22.23
    Interface Subtype        : ifIndex
    Interface Number         : 127
    Object Identifier        :

Local Port Alias: 2
  Remote Device Index      : 2
  Remote Device TTL       : 4651
  Remote Device RxChanges : false
  Chassis Type            : Mac Address
  Chassis Id              : 08-17-f4-a1-db-00
  Port Type               : Locally Assigned
  Port Id                 : 2
  Port Description        : 2

  System Name             :
  System Description      : Lenovo Networking Operating
System RackSwitch G8296, Lenovo Networking OS: version 8.2.1, Boot Image:
version 8.2.1
  System Capabilities Supported : bridge, router
  System Capabilities Enabled   : bridge, router

  Remote Management Address:
    Subtype                  : IPv4
    Address                   : 10.38.22.23
    Interface Subtype        : ifIndex
    Interface Number         : 127
    Object Identifier        :

Total entries displayed: 2
```

Time-to-Live for Received Information

Each remote device LLDP packet includes an expiration time. If the switch port does not receive an LLDP update from the remote device before the time-to-live clock expires, the switch will consider the remote information to be invalid, and will remove all associated information from the MIB.

Remote devices can also intentionally set their LLDP time-to-live to 0, indicating to the switch that the LLDP information is invalid and must be immediately removed.

LLDP Example Configuration

1. Turn LLDP on globally.

```
RS8296(config)# lldp enable
```

2. Set the global LLDP timer features.

```
RS8296(config)# lldp refresh-interval 30           (Transmit each 30 seconds)
RS8296(config)# lldp transmission-delay 2         (No more often than 2 sec.)
RS8296(config)# lldp holdtime-multiplier 4        (Remote hold 4 intervals)
RS8296(config)# lldp reinit-delay 2               (Wait 2 sec. after reinit.)
RS8296(config)# lldp trap-notification-interval 5 (Minimum 5 sec. between)
```

3. Set LLDP options for each port.

```
RS8296(config)# interface port <n>               (Select a switch port)
RS8296(config-if)# lldp admin-status tx_rx        (Transmit and receive LLDP)
RS8296(config-if)# lldp trap-notification         (Enable SNMP trap notifications)
RS8296(config-if)# lldp tlv all                   (Transmit all optional information)
RS8296(config-if)# exit
```

4. Enable syslog reporting.

```
RS8296(config)# logging log lldp
```

5. Verify the configuration settings:

```
RS8296(config)# show lldp
```

6. View remote device information as needed.

```
RS8296(config)# show lldp remote-device
  or
RS8296(config)# show lldp remote-device <index number>
  or
RS8296(config)# show lldp remote-devices detail
```

Chapter 38. Simple Network Management Protocol

Lenovo Network Operating System provides Simple Network Management Protocol (SNMP) version 1, version 2, and version 3 support for access through any network management software, such as IBM Director or HP-OpenView.

Note: SNMP read and write functions are enabled by default. For best security practices, if SNMP is not needed for your network, it is recommended that you disable these functions prior to connecting the switch to the network.

SNMP Version 1 & Version 2

To access the SNMP agent on the G8296, the read and write community strings on the SNMP manager must be configured to match those on the switch. The default read community string on the switch is `public` and the default write community string is `private`.

The read and write community strings on the switch can be changed using the following commands on the CLI:

```
RS8296(config)# snmp-server read-community <1-32 characters>
-and-
RS8296(config)# snmp-server write-community <1-32 characters>
```

The SNMP manager must be able to reach the management interface or any one of the IP interfaces on the switch.

For the SNMP manager to receive the SNMPv1 traps sent out by the SNMP agent on the switch, configure the trap host on the switch with the following command:

```
RS8296(config)# snmp-server trap-source <trap source IP interface>
RS8296(config)# snmp-server host <IPv4 address> <trap host community string>
```

Note: You can use a loopback interface to set the source IP address for SNMP traps. Use the following command to apply a configured loopback interface:

```
RS8296(config)# snmp-server trap-source loopback <1-5>
```

SNMP Version 3

SNMP version 3 (SNMPv3) is an enhanced version of the Simple Network Management Protocol, approved by the Internet Engineering Steering Group in March, 2002. SNMPv3 contains additional security and authentication features that provide data origin authentication, data integrity checks, timeliness indicators and encryption to protect against threats such as masquerade, modification of information, message stream modification and disclosure.

SNMPv3 allows clients to query the MIBs securely.

SNMPv3 configuration is managed using the following command path menu:

```
RS8296(config)# snmp-server ?
```

For more information on SNMP MIBs and the commands used to configure SNMP on the switch, see the *Lenovo Network Operating System 8.2 Command Reference*.

Default Configuration

Networking OS has three SNMPv3 users by default. All the three users have access to all the MIBs supported by the switch:

- User 1 name is `adminmd5` (password `adminmd5`). Authentication used is MD5. Privacy protocol used is DES.
- User 2 name is `adminsha` (password `adminsha`). Authentication used is SHA. Privacy protocol used is DES.
- User 3 name is `adminshaaes` (password `Edpq132x!#9Zpx432w`). Authentication used is SHA. Privacy protocol used is AES-128.

In boot strict mode (See [“Boot Strict Mode” on page 43](#)), Networking OS has one SNMPv3 user:

- User 1 name is `adminshaaes` (password `Edpq132x!#9Zpx432w`). Authentication used is SHA. Privacy protocol used is AES-128.

Up to 16 SNMP users can be configured on the switch. To modify an SNMP user, enter the following commands:

```
RS8296(config)# snmp-server user <1-16> name <1-32 characters>
```

Users can be configured to use the authentication/privacy options. The G8296 support two authentication algorithms: MD5 and SHA, as specified in the following command:

```
RS8296(config)# snmp-server user <1-16> authentication-protocol {md5|sha}
authentication-password
-or-
RS8296(config)# snmp-server user <1-16> authentication-protocol none
```

User Configuration Example

1. To configure a user with name “admin,” authentication type MD5, and authentication password of “admin,” privacy option DES with privacy password of “admin,” use the following CLI commands.

```
RS8296(config)# snmp-server user 5 name admin
RS8296(config)# snmp-server user 5 authentication-protocol md5
authentication-password
Changing authentication password; validation required:
Enter current admin password:          <admin. password>
Enter new authentication password:
                                         <auth. password; Should be more than 18 characters >
Re-enter new authentication password: <auth. password>
New authentication password accepted.

RS8296(config)# snmp-server user 5 privacy-protocol des privacy-password
Changing privacy password; validation required:
Enter current admin password:          <admin. password>
Enter new privacy password:
                                         <privacy password; Should be more than 18 characters>
Re-enter new privacy password:        <privacy password>
New privacy password accepted.
```

2. Configure a user access group, along with the views the group may access. Use the access table to configure the group’s access level.

```
RS8296(config)# snmp-server access 5 name admingrp
RS8296(config)# snmp-server access 5 level authpriv
RS8296(config)# snmp-server access 5 read-view iso
RS8296(config)# snmp-server access 5 write-view iso
RS8296(config)# snmp-server access 5 notify-view iso
```

Because the read view, write view, and notify view are all set to “iso,” the user type has access to all private and public MIBs.

3. Assign the user to the user group. Use the group table to link the user to a particular access group.

```
RS8296(config)# snmp-server group 5 user-name admin
RS8296(config)# snmp-server group 5 group-name admingrp
```

Configuring SNMP Trap Hosts

SNMPv1 Trap Host

1. Configure a user with no authentication and password.

```
RS8296(config)# snmp-server user 10 name v1trap
```

2. Configure an access group and group table entries for the user. Use the following menu to specify which traps can be received by the user:

```
RS8296(config)# snmp-server access <user number>
```

In the following example the user will receive the traps sent by the switch.

```
RS8296(config)# snmp-server access 10          (Access group to view SNMPv1 traps)
name v1trap
security snmpv1
notify-view iso
RS8296(config)# snmp-server group 10          (Assign user to the access group)
security snmpv1
user-name v1trap
group-name v1trap
```

3. Configure an entry in the notify table.

```
RS8296(config)# snmp-server notify 10 name v1trap
RS8296(config)# snmp-server notify 10 tag v1trap
```

4. Specify the IPv4 address and other trap parameters in the `targetAddr` and `targetParam` tables. Use the following commands to specify the user name associated with the `targetParam` table:

```
RS8296(config)# snmp-server target-address 10 name v1trap address
10.70.70.190
RS8296(config)# snmp-server target-address 10 parameters-name v1param
RS8296(config)# snmp-server target-address 10 taglist v1param
RS8296(config)# snmp-server target-parameters 10 name v1param
RS8296(config)# snmp-server target-parameters 10 user-name v1trap
RS8296(config)# snmp-server target-parameters 10 message snmpv1
```

Note: N/OS 8.2 supports only IPv4 addresses for SNMPv1 and SNMPv2 trap hosts.

5. Use the community table to specify which community string is used in the trap.

```
RS8296(config)# snmp-server community 10      (Define the community string)
index v1trap
name public
user-name v1trap
```

SNMPv2 Trap Host Configuration

The SNMPv2 trap host configuration is similar to the SNMPv1 trap host configuration. Wherever you specify the model, use `snmpv2` instead of `snmpv1`.

```
RS8296(config)# snmp-server user 10 name v2trap

RS8296(config)# snmp-server group 10 security snmpv2
RS8296(config)# snmp-server group 10 user-name v2trap
RS8296(config)# snmp-server group 10 group-name v2trap
RS8296(config)# snmp-server access 10 name v2trap
RS8296(config)# snmp-server access 10 security snmpv2
RS8296(config)# snmp-server access 10 notify-view iso

RS8296(config)# snmp-server notify 10 name v2trap
RS8296(config)# snmp-server notify 10 tag v2trap

RS8296(config)# snmp-server target-address 10 name v2trap
                address 100.10.2.1
RS8296(config)# snmp-server target-address 10 taglist v2trap
RS8296(config)# snmp-server target-address 10 parameters-name
                v2param
RS8296(config)# snmp-server target-parameters 10 name v2param
RS8296(config)# snmp-server target-parameters 10 message snmpv2c
RS8296(config)# snmp-server target-parameters 10 user-name v2trap
RS8296(config)# snmp-server target-parameters 10 security snmpv2

RS8296(config)# snmp-server community 10 index v2trap
RS8296(config)# snmp-server community 10 user-name v2trap
```

Note: N/OS 8.2 supports only IPv4 addresses for SNMPv1 and SNMPv2 trap hosts.

SNMPv3 Trap Host Configuration

To configure a user for SNMPv3 traps, you can choose to send the traps with both privacy and authentication, with authentication only, or without privacy or authentication.

This is configured in the access table using the following commands:

```
RS8296(config)# snmp-server access <1-32> level
RS8296(config)# snmp-server target-parameters <1-16> level
```

Configure the user in the user table accordingly.

It is not necessary to configure the community table for SNMPv3 traps because the community string is not used by SNMPv3.

The following example shows how to configure an SNMPv3 user `v3trap` with authentication only:

```
RS8296(config)# snmp-server user 11 name v3trap
RS8296(config)# snmp-server user 11 authentication-protocol md5
authentication-password
Changing authentication password; validation required:
Enter current admin password: <admin. password>
Enter new authentication password: <auth. password>
Re-enter new authentication password: <auth. password>
New authentication password accepted.
RS8296(config)# snmp-server access 11 notify-view iso
RS8296(config)# snmp-server access 11 level authNoPriv
RS8296(config)# snmp-server access 11 name v3trap
RS8296(config)# snmp-server group 11 user-name v3trap
RS8296(config)# snmp-server group 11 group-name v3trap
RS8296(config)# snmp-server notify 11 name v3trap
RS8296(config)# snmp-server notify 11 tag v3trap
RS8296(config)# snmp-server target-address 11 name v3trap address
47.81.25.66
RS8296(config)# snmp-server target-address 11 taglist v3trap
RS8296(config)# snmp-server target-address 11 parameters-name v3param
RS8296(config)# snmp-server target-parameters 11 name v3param
RS8296(config)# snmp-server target-parameters 11 user-name v3trap
RS8296(config)# snmp-server target-parameters 11 level authNoPriv
```

SNMP MIBs

The N/OS SNMP agent supports SNMP version 3. Security is provided through SNMP community strings. The default community strings are “public” for SNMP GET operation and “private” for SNMP SET operation. The community string can be modified only through the Command Line Interface (CLI). Detailed SNMP MIBs and trap definitions of the N/OS SNMP agent are contained in the N/OS enterprise MIB document.

The N/OS SNMP agent supports the following standard MIBs:

- dot1x.mib
- ieee8021ab.mib
- ieee8023ad.mib
- rfc1213.mib
- rfc1215.mib
- rfc1493.mib
- rfc1573.mib
- rfc1643.mib
- rfc1657.mib
- rfc1757.mib
- rfc1850.mib
- rfc1907.mib
- rfc2037.mib
- rfc2233.mib
- rfc2465.mib
- rfc2571.mib
- rfc2572.mib
- rfc2573.mib
- rfc2574.mib
- rfc2575.mib
- rfc2576.mib
- rfc2790.mib
- rfc3176.mib
- rfc4133.mib
- rfc4363.mib

The N/OS SNMP agent supports the following generic traps as defined in RFC 1215:

- ColdStart
- WarmStart
- LinkDown
- LinkUp
- AuthenticationFailure

The SNMP agent also supports two Spanning Tree traps as defined in RFC 1493:

- NewRoot
- TopologyChange

The following are the enterprise SNMP traps supported in N/OS:

Table 44. *Networking OS-Supported Enterprise SNMP Traps*

Trap Name	Description
swLoginFailure	Signifies that someone failed to enter a valid username/password combination. swTrapDisplayString specifies whether the login attempt was from CONSOLE or TELNET. In case of TELNET login it also specifies the IP address of the host from which the attempt was made.
swValidLogin	Signifies that a user login has occurred.
swApplyComplete	Signifies that new configuration has been applied.
swSaveComplete	Signifies that new configuration has been saved.
swFwDownloadSucess	Signifies that firmware has been downloaded to [image1 image2 boot image].
swFwDownloadFailure	Signifies that firmware downloaded failed to [image1 image2 boot image].
swValidLogout	Signifies that a user logout has occurred.
swPrimaryPowerSupplyFailure	Signifies that the primary power supply has failed.
swPrimaryPowerSupplyFixed	Signifies that the primary power supply has been restored.
swDefAdminDisable	Signifies that the default admin account has been disabled.
swAcntStrngPswdNotMet	Signifies that the configured password does not match strong password complexity.
swAcntLocked	Signifies that account has been locked.
swAcntUnlocked	Signifies that account has been unlocked.
swStgNewRoot	Signifies that the bridge has become the new root of the STG.

Table 44. *Networking OS-Supported Enterprise SNMP Traps (continued)*

Trap Name	Description
swCistNewRoot	Signifies that the bridge has become the new root of the CIST.
swStgTopologyChanged	Signifies that there was a STG topology change.
swCistTopologyChanged	Signifies that there was a CIST topology change.
swHotlinksMasterUp	Signifies that the Master interface is active.
swHotlinksMasterDn	Signifies that the Master interface is not active.
swHotlinksBackupUp	Signifies that the Backup interface is active.
swHotlinksBackupDn	Signifies that the Backup interface is not active.
swHotlinksNone	Signifies that there are no active interfaces.
swTeamingCtrlUp	Signifies that the teaming is up.
swTeamingCtrlDown	Signifies that the teaming control is down.
swTeamingCtrlDownTearDownBlked	Signifies that the teaming control is down but teardown is blocked.
swTeamingCtrlError	Signifies error, action is undefined.
swLACPPortBlocked	Signifies that LACP is operationally down on a port, and traffic is blocked on the port.
swLACPPortUnblocked	Signifies that LACP is operationally up on a port, and traffic is no longer blocked on the port.
swLFDPortErrdisabled	Signifies that a port is error-disabled due to excessive link flaps.
swVlagInstanceUp	Signifies that VLAG instance is up identified in the trap message.
swVlagInstanceRemoteUp	Signifies that VLAG is down but instance on the remote instance is up.
swVlagInstanceLocalUp	Signifies that VLAG is down but local instance is up.

Table 44. *Networking OS-Supported Enterprise SNMP Traps (continued)*

Trap Name	Description
swVlagInstanceDown	Signifies that VLAG instance is down identified in the trap message.
swVlagIslUp	Signifies that connection between VLAG switches is up.
swVlagIslDown	Signifies that connection between VLAG switches is down.
sw8021x	Signifies 802.1x feature is enabled.
swDefGwUp	Signifies that the default gateway is alive. ipCurCfgGwIndex is the index of the Gateway in ipCurCfgGwTable. The range for ipCurCfgGwIndex is from 1 to ipGatewayTableMax. ipCurCfgGwAddr is the IP address of the default gateway.
swDefGwDown	Signifies that the default gateway is down. ipCurCfgGwIndex is the index of the Gateway in ipCurCfgGwTable. The range for ipCurCfgGwIndex is from 1 to ipGatewayTableMax. ipCurCfgGwAddr is the IP address of the default gateway.
swDefGwInService	Signifies that the default gateway is up and in service. ipCurCfgGwIndex is the index of the Gateway in ipCurCfgGwTable. The range for ipCurCfgGwIndex is from 1 to ipGatewayTableMax. ipCurCfgGwAddr is the IP address of the default gateway.
swDefGwNotInService	Signifies that the default gateway is alive but not in service. ipCurCfgGwIndex is the index of the Gateway in ipCurCfgGwTable. The range for ipCurCfgGwIndex is from 1 to ipGatewayTableMax. ipCurCfgGwAddr is the IP address of the default gateway.

Table 44. *Networking OS-Supported Enterprise SNMP Traps (continued)*

Trap Name	Description
swVrrpNewMaster	Indicates that the sending agent has transitioned to “Master” state. vrrpCurCfgVirtRtrIndx is the VRRP virtual router table index referenced in vrrpCurCfgVirtRtrTable. The range is from 1 to vrrpVirtRtrTableMaxSize. vrrpCurCfgVirtRtrAddr is the VRRP virtual router IP address.
swVrrpNewBackup	Indicates that the sending agent has transitioned to “Backup” state. vrrpCurCfgVirtRtrIndx is the VRRP virtual router table index referenced in vrrpCurCfgVirtRtrTable. The range is from 1 to vrrpVirtRtrTableMaxSize. vrrpCurCfgVirtRtrAddr is the VRRP virtual router IP address.
swVrrpAuthFailure	Signifies that a packet has been received from a router whose authentication key or authentication type conflicts with this router's authentication key or authentication type. vrrpCurCfgIfIndx is the VRRP interface index. This is equivalent to ifIndex in RFC 1213 mib. The range is from 1 to vrrpIfTableMaxSize. vrrpCurCfgIfPasswd is the password for authentication. It is a DisplayString of 0 to 7 characters.
swTcpHoldDown	Signifies that new TCP connection requests from a particular client will be blocked for a pre-determined amount of time since the rate of new TCP connections from that client has reached a pre-determined threshold. The fltCurCfgSrcIp is the client source IP address for which new TCP connection requests will be blocked. The swTrapRate specifies the amount of time in minutes that the particular client will be blocked.

Table 44. *Networking OS-Supported Enterprise SNMP Traps (continued)*

Trap Name	Description
swNTPSyncFailed	Signifies that synchronization with the NTP server has failed.
swNTPUpdateClock	Signifies that the system clock is updated with NTP server.
swECMPGatewayUp	Signifies that the ECMP gateway is up.
swECMPGatewayDown	Signifies that the ECMP gateway is down.
swTempExceedThreshold	Signifies that the switch temperature has exceeded maximum safety limits.
swFanFailure	Signifies that the fan failure has been detected.
swFanFailureFixed	Signifies that the fan failure has been fixed.
swTempReturnThreshold	Signifies that the switch temperature has returned to under maximum safety limits.
swVMGroupVMotion	Signifies that a virtual machine has moved from a port to another.
swVMGroupVMOnline	Signifies that an advance provisioned virtual machine has come online.
swVMGroupVMVlanChange	Signifies that a virtual machine has entered a VLAN, or changed the VLAN.
swVmCheckSpoofedvm	Signifies that a spoofed VM MAC was found.
swOfFlowConnectControllerFailure	Signifies that the connection to OpenFlow controller is broken.
swOfFlowConnectControllerSuccess	Signifies that the connection to OpenFlow controller is successful.
swOfFlowEmergencyState	Signifies that the OpenFlow switch has moved to emergency state.
swOfFlowNoEmergencyState	Signifies that the OpenFlow switch is out of emergency state.
swOfFlowSwitchOfFlowMode	Signifies that the device is working in OpenFlow switch mode.

Table 44. *Networking OS-Supported Enterprise SNMP Traps (continued)*

Trap Name	Description
swOflowSwitchNomalMode	Signifies that the device is working in normal switch mode.
swOflowSwitchFlowtableClr	Signifies that the flow table entries are cleared from the device.
swOflowSwitchStatsClr	Signifies that the OpenFlow statistics are cleared from the device.
swOflowFlowtableEntryUnavail	Signifies that the device is unable to add flow entry in OpenFlow due to unavailability of system resources.
swOflowFlowtableLimit	Signifies that the device has reached its configured flow limit.
swOflowPortDownController	Signifies that the port is operationally down by OpenFlow controller.
swOflowPortUpController	Signifies that the port is operationally up by OpenFlow controller.
swOflowVersionNegotiationError	Signifies that the device is unable to connect to controller either due to bad version, or negotiation/hello failure.

Switch Images and Configuration Files

This section describes how to use MIB calls to work with switch images and configuration files. You can use a standard SNMP tool to perform the actions, using the MIBs listed in [Table 45](#).

[Table 45](#) lists the MIBs used to perform operations associated with the Switch Image and Configuration files.

Table 45. *MIBs for Switch Image and Configuration Files*

MIB Name	MIB OID
agTransferServer	1.3.6.1.4.1.26534.2.7.6.1.1.7.1
agTransferImage	1.3.6.1.4.1.26534.2.7.6.1.1.7.2
agTransferImageFileName	1.3.6.1.4.1.26534.2.7.6.1.1.7.3
agTransferCfgFileName	1.3.6.1.4.1.26534.2.7.6.1.1.7.4
agTransferDumpFileName	1.3.6.1.4.1.26534.2.7.6.1.1.7.5
agTransferAction	1.3.6.1.4.1.26534.2.7.6.1.1.7.6
agTransferLastActionStatus	1.3.6.1.4.1.26534.2.7.6.1.1.7.7
agTransferPort	1.3.6.1.4.1.26534.2.7.6.1.1.7.8
agTransferUserName	1.3.6.1.4.1.26534.2.7.6.1.1.7.9
agTransferPassword	1.3.6.1.4.1.26534.2.7.6.1.1.7.10
agTransferTSDumpFileName	1.3.6.1.4.1.26534.2.7.6.1.1.7.11

The following SNMP actions can be performed using the MIBs listed in [Table 45](#).

- Load a new Switch image (boot or running) from a FTP/TFTP server
- Load a previously saved switch configuration from a FTP/TFTP server
- Save the switch configuration to a FTP/TFTP server
- Save a switch dump to a FTP/TFTP server

Loading a New Switch Image

To load a new switch image with the name "MyNewImage-1.img" into image2, follow these steps. This example shows an FTP/TFTP server with the IPv4 address 192.168.10.10, though IPv6 is also supported.

1. Set the FTP/TFTP server address where the switch image resides:

```
Set agTransferServer.0 "192.168.10.10"
```

2. Set the area where the new image will be loaded:

```
Set agTransferImage.0 "image2"
```

3. Set the name of the image:

```
Set agTransferImageFileName.0 "MyNewImage-1.img"
```

4. If you are using an FTP server, enter a username:

```
Set agTransferUserName.0 "MyName"
```

5. If you are using an FTP server, enter a password:

```
Set agTransferPassword.0 "MyPassword"
```

6. Initiate the transfer. To transfer a switch image, enter 2 (gting):

```
Set agTransferAction.0 "2"
```

Loading a Saved Switch Configuration

To load a saved switch configuration with the name "MyRunningConfig.cfg" into the switch, follow these steps. This example shows a TFTP server at IPv4 address 192.168.10.10, though IPv6 is also supported.

1. Set the FTP/TFTP server address where the switch configuration file resides:

```
Set agTransferServer.0 "192.168.10.10"
```

2. Set the name of the configuration file:

```
Set agTransferCfgFileName.0 "MyRunningConfig.cfg"
```

3. If you are using an FTP server, enter a username:

```
Set agTransferUserName.0 "MyName"
```

4. If you are using an FTP server, enter a password:

```
Set agTransferPassword.0 "MyPassword"
```

5. Initiate the transfer. To restore a running configuration, enter 3:

```
Set agTransferAction.0 "3"
```

Saving the Switch Configuration

To save the switch configuration to a FTP/TFTP server follow these steps. This example shows a FTP/TFTP server with the IPv4 address 192.168.10.10, though IPv6 is also supported.

1. Set the FTP/TFTP server address where the configuration file is saved:

```
Set agTransferServer.0 "192.168.10.10"
```

2. Set the name of the configuration file:

```
Set agTransferCfgFileName.0 "MyRunningConfig.cfg"
```

3. If you are using an FTP server, enter a username:

```
Set agTransferUserName.0 "MyName"
```

4. If you are using an FTP server, enter a password:

```
Set agTransferPassword.0 "MyPassword"
```

5. Initiate the transfer. To save a running configuration file, enter 4:

```
Set agTransferAction.0 "4"
```

Saving a Switch Dump

To save a switch dump to a FTP/TFTP server, follow these steps. This example shows an FTP/TFTP server with the 192.168.10.10, though IPv6 is also supported.

1. Set the FTP/TFTP server address where the dump file will be saved:

```
Set agTransferServer.0 "192.168.10.10"
```

2. Set the name of the dump file:

```
Set agTransferDumpFileName.0 "MyDumpFile.dmp"
```

3. If you are using an FTP server, enter a username:

```
Set agTransferUserName.0 "MyName"
```

4. If you are using an FTP server, enter a password:

```
Set agTransferPassword.0 "MyPassword"
```

5. Initiate the transfer. To save a dump file, enter 5:

```
Set agTransferAction.0 "5"
```

Chapter 39. NETCONF

The Network Configuration Protocol (NETCONF) provides a mechanism to manage the G8296, retrieve or modify existing configuration data, and upload new configuration data. See RFC 4741 for details on NETCONF.

NETCONF operates in a client/server model. The NETCONF client establishes a session with the switch (acting as a NETCONF server) using a Remote Procedure Call (RPC). NETCONF is based on the Extensible Markup Language (XML) for encoding data and for exchanging configuration and protocol messages.

The following topics are discussed in this section:

- [“NETCONF Overview” on page 546](#)
- [“XML Requirements” on page 547](#)
- [“Installing the NETCONF Client” on page 548](#)
- [“Using Juniper Perl Client” on page 550](#)
- [“Establishing a NETCONF Session” on page 551](#)
- [“NETCONF Operations” on page 553](#)
- [“Protocol Operations Examples” on page 554](#)

NETCONF Overview

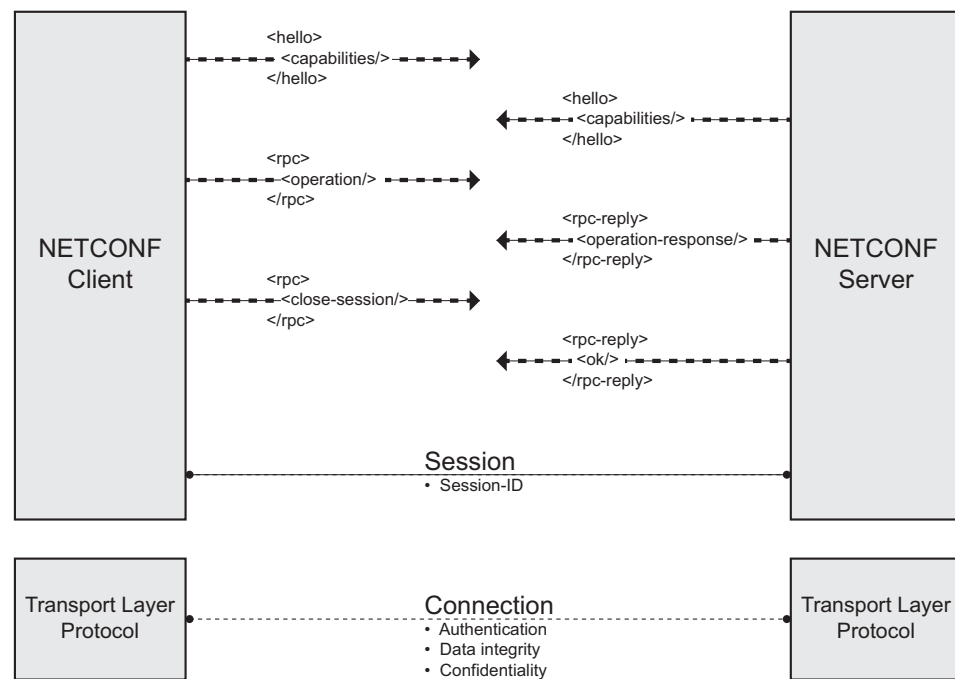
NETCONF provides a method to quickly configure the switch. It also allows you to implement a configuration across multiple switches, thereby saving time and reducing the chances of configuration errors.

The NETCONF protocol defines basic operations that are equivalent to the switch ISCLI commands.

Note: The current implementation of NETCONF supports only ISCLI commands.

NETCONF is a connection-oriented protocol. See [Figure 53](#) for an overview of NETCONF operation.

Figure 53. NETCONF Operations Procedure



1. The client establishes a transport layer connection to the switch (acting as a NETCONF server).
2. The client and switch exchange hello messages to declare their capabilities.
3. The client sends a request via rpc message to the switch.
4. The switch sends a response via rpc-reply message to the client.

Note: Steps 3 and 4 must be repeated for each request that the client sends to the switch.

5. The client sends a close-session message to the switch to end the NETCONF session and the transport layer connection.
6. The switch sends an OK response.

XML Requirements

XML is the encoding format used within NETCONF. When using XML for NETCONF:

- All NETCONF protocol elements are defined in the following namespace:
`urn:ietf:params:xml:ns:netconf:base:1.0`
- NETCONF capability names must be Uniform Resource Identifiers (URIs):
`urn:ietf:params:netconf:capability:{name}:1.0`
where *{name}* is the name of the capability.
- Document type declarations must not appear in the NETCONF content.
- For Secure Shell (SSH), you must use a special message termination sequence of six characters to provide message framing:
`]]>]]>`

Installing the NETCONF Client

You can download the required NETCONF Client installation files from www.ibm.com. Select **Support & downloads > Fixes, updates and drivers**. Follow instructions on the IBM Support Portal page to find the files.

Before installing the NETCONF client, ensure you have completed the following tasks:

- Install a supported version of Python (Python 2.6 or higher, up to but not including Python 3.0) in the folder C:\.
- Install the PyCrypto application appropriate to the Python version you are using.

Note: The following steps are for the Windows operating systems.

Follow these steps to install the Blade NETCONF Python Client (BNClient):

1. Extract the file `blade-netconf-python-client-v0.1.zip` to the root (C:\) folder.

You will see two folders under the root folder
C:\blade-netconf-python-client-v0.1:

- `blade-netconf-python-client`
- `python-library`

Note: Make sure Paramiko version 1.7.4 or higher is in the folder
C:\blade-netconf-python-client-v0.1\python-ssh-library\

2. Open the command prompt (**Select Start > Run > cmd**).
3. Enter the following command to install the SSH library:

```
python C:\blade-netconf-python-client-v0.1\python-ssh-library\  
paramiko-1.7.6\setup.py install
```

Note: If the `python` command does not work from the command prompt, you may need to add a system variable to set the path to the directory where you have installed Python. You can add the system variable at the following location: **My Computer > Properties > Advanced > Environment Variables**

Follow these steps to install BNClient as a python script:

1. Enter the following command for help:

```
python C:\blade-netconf-python-client-v0.1\blade-netconf-python-  
client\bnclient\bnclient.py -h
```

2. Enter the following command to establish a NETCONF session:

- Using SSH to connect to default port 830:

```
python C:\blade-netconf-python-client-v0.1\blade-netconf-python-  
client\bnclient\bnclient.py {switch IP address} -u admin -p admin -o get
```

- Using SSH to connect to port 22:

```
python C:\blade-netconf-python-client-v0.1\blade-netconf-python-client\bnclient\bnclient.py {switch IP address}:22 -u admin -p admin -o get
```

Follow these steps to install BNClient as a python library:

1. Open the file `C:\blade-netconf-python-client-v0.1\blade-netconf-python-client\example\get.py` in a Python editor (For example, IDLE).
2. Change the IP address in the *hostname* field to the switch IP address, and save the file.
3. Enter the following command to establish a session:

```
python C:\blade-netconf-python-client-v0.1\blade-netconf-python-client\setup.py install
```

4. Enter the following command to get the running configuration:

```
python C:\blade-netconf-python-client-v0.1\blade-netconf-python-client\example\get.py
```

Note: The file `get.py` is an example of a NETCONF operation python script. You can edit the script or write a new script as per your requirements.

Using Juniper Perl Client

You can use Juniper Perl client instead of BNClient to communicate with the NETCONF feature on the switch. Follow these steps to use the Juniper Perl client.

Note: You must use the Linux operating system for the Juniper Perl client.

1. Extract the file `juniper-netconf-perl-client.zip` to the folder `/home/user/`.

You will see two folders:

- `juniper-netconf-perl-client`
- `blade-netconf-perl-scripts`

2. Follow these steps to install the Juniper Perl client:

As a Perl library:

- a. Change to the following directory:

```
/home/user/juniper-netconf-perl-client
```

- b. Extract the following file:

```
netconf-perl-10.0R2.10.tar.gz
```

- c. Change to the following directory:

```
/home/user/juniper-netconf-perl-client/netconf-perl-10.0R2.10
```

- d. Install the client as per the instructions in the README file.

Note: If the prerequisites package installation fails, manually install each file in `/home/user/juniper-netconf-perl-client/netconf-perl-prereqs-patch`.

As a Perl script:

- a. Change to the following directory:

```
/home/user/blade-netconf-perl-scripts/
```

- b. Enter the following command:

```
perl get/get.pl -l admin -p admin {switch IP address}
```

Note: The file `get.pl` is an example of a NETCONF operation Perl script. You can edit the script or write a new script as per your requirement.

Establishing a NETCONF Session

SSH is the widely used protocol for NETCONF sessions. The default SSH port for NETCONF is 830. The client may also connect to the switch through SSH port 22.

Follow these steps to establish a NETCONF session. Enter commands in the client Linux Shell.

Note: You can open a maximum of four simultaneous sessions.

1. Enter the following command to open an SSH connection:

```
ssh admin@[switch IP address] -p 830 -s netconf
```

2. Type or paste the following hello message:

```
<hello>
  <capabilities>
    <capability>urn:ietf:params:netconf:base:1.0</capability>
  </capabilities>
</hello>
]]>]]>
```

The switch returns a hello message:

```
<hello xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <capabilities>
    <capability>urn:ietf:params:netconf:base:1.0</capability>
    <capability>urn:ietf:params:netconf:capability:writable-running:1.0</capability>
    <capability>urn:ietf:params:netconf:capability:rollback-on-error:1.0</capability>
    <capability>urn:ietf:params:netconf:capability:startup:1.0</capability>
  </capabilities>
  <session-id>102</session-id>
</hello>
]]>]]>
```

3. Type or paste the following rpc message. The get operation is used as an example.

```
<rpc message-id="100">
  <get>
    <filter type="subtree">
      <configuration-text/>
    </filter>
  </get>
</rpc>
]]>]]>
```

The switch sends an rpc-reply message:

```
<rpc-reply message-id="100">
  <data>
    <configuration-text
      xmlns="http://www.ibm.com/netconf/1.0/config-text"> version "8.2.1"
      switch-type "Lenovo Networking Operating System RackSwitch
      G8296"
      !
      !
      no system dhcp mgta
      !
      !
      interface ip 127
      ip address 172.31.36.51
      enable
      exit
      !
      ip gateway 3 address 172.31.1.1
      ip gateway 3 enable
      !
      !
      end
    </configuration-text>
  </data>
</rpc-reply>
]]>]]>
```

Note: Repeat Step 3 for each request you need to send to the switch.

4. Type or paste the following close-session message to close the NETCONF session and terminate the SSH connection.

```
<rpc message-id="101">
  <close-session/>
</rpc>
]]>]]>
```

The switch sends the following response:

```
<rpc-reply message-id="101">
  <ok/>
</rpc-reply>
]]>]]>
```

NETCONF Operations

The NETCONF protocol provides a set of operations to manage and retrieve switch configuration. [Table 46](#) provides a list of protocol operations supported by the switch.

Table 46. *Protocol Operations*

Operation	Description
get-config	Retrieve all or part of the running or startup configuration.
edit-config	Load all or part of a specified configuration to the running or startup configuration.
copy-config	Replace the target running or startup configuration with a source running or startup configuration.
delete-config	Delete startup configuration.
lock	Lock the running configuration to prevent other users (via another NETCONF session) from changing it.
unlock	Release a locked running configuration.
get	Retrieve running configuration and device state information.
close-session	Request graceful termination of a NETCONF session.
kill-session	Force the termination of a NETCONF session.
get-configuration	Retrieve configuration data from the switch.
get-interface-information	Retrieve interface status information.

Protocol Operations Examples

Following are examples of the NETCONF protocol operations supported by the G8296.

<get-config>

Usage:

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-config>
    <source>
      <running/>
    </source>
    <filter type="subtree">
      <configuration-text
        xmlns="http://www.ibm.com/netconf/1.0/config-text"/>
    </filter>
  </get-config>
</rpc>
```

Response from the switch:

```
<rpc-reply message-id="101"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    <configuration-text
      xmlns="http://www.ibm.com/netconf/1.0/config-text">
      <!-- configuration text... -->
    </configuration-text>
  </data>
</rpc-reply>
```

See [Table 47](#) for the tag elements and their values.

Table 47. get-config Tag Element Values

Tag Element	Description	Value
source	The configuration text you want to retrieve.	running/ or startup/
filter type="subtree"	The filter type.	subtree
!--configuration text...--	Contains the running configuration in ISCLI format.	

<edit-config>

Usage:

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <edit-config>
    <target>
      <running/>
    </target>
    <default-operation>
      <merge/>
    </default-operation>
    <error-option>
      <stop-on-error/>
    </error-option>
    <config-text xmlns="http://www.ibm.com/netconf/1.0/config-text">
      <configuration-text>hostname Router</configuration-text>
    </config-text>
  </edit-config>
</rpc>
```

Response from the switch:

```
<rpc-reply message-id="101"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

See [Table 48](#) for the tag elements and their values.

Table 48. edit-config *Tag Element Values*

Tag Element	Description	Value
target	The configuration you want to edit.	running/ or startup/

Table 48. edit-config Tag Element Values

Tag Element	Description	Value
default-operation	Set the default operation for the edit-config request.	<ul style="list-style-type: none">● merge: The new configuration is merged with the target configuration at the corresponding level.● replace: The new configuration replaces the target configuration.● none: The target configuration does not change unless the configuration data in the configuration-text parameter uses the operation attribute to request a different operation.
error-option	Set the option to handle configuration error.	<ul style="list-style-type: none">● stop-on-error: Abort the edit-config operation on first error. This is the default error-option.● continue-on-error: Continue to process configuration data on error.● rollback-on-error: Abort the edit-config operation on first error and discard the requested configuration changes.

<copy-config>

Usage:

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <copy-config>
    <target>
      <startup/>
    </target>
    <source>
      <running/>
    </source>
  </copy-config>
</rpc>
```

Response from the switch:

```
<rpc-reply message-id="101"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

See [Table 49](#) for the tag elements and their values.

Table 49. copy-config *Tag Element Values*

Tag Element	Description	Value
target	Configuration that needs to be changed.	running/ or startup/
source	Source configuration.	running/ or startup/

<delete-config>

Usage:

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <delete-config>
    <target>
      <startup/>
    </target>
  </delete-config>
</rpc>
```

Response from the switch:

```
<rpc-reply message-id="101"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

See [Table 50](#) for the tag elements and their values.

Table 50. delete-config Tag Element Values

Tag Element	Description	Value
target	Configuration that needs to be deleted.	startup/

<lock>

Usage:

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <lock>
    <target>
      <running/>
    </target>
  </lock>
</rpc>
```

Response from the switch:

```
<rpc-reply message-id="101"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

See [Table 51](#) for the tag elements and their values.

Table 51. lock Tag Element Values

Tag Element	Description	Value
target	Configuration that needs to be edited.	running/

<unlock>

Usage:

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">  
  <unlock>  
    <target>  
      <running/>  
    </target>  
  </unlock>  
</rpc>
```

Response from the switch:

```
<rpc-reply message-id="101"  
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">  
  <ok/>  
</rpc-reply>
```

See [Table 52](#) for the tag elements and their values.

Table 52. unlock Tag Element Values

Tag Element	Description	Value
target	Configuration being edited.	running/

<get>

Usage:

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get>
    <filter type="subtree">
      <!-- request a text version of the configuration -->
      <configuration-text
        xmlns="http://www.ibm.com/netconf/1.0/config-text"/>
    </filter>
  </get>
</rpc>
```

Response from the switch:

```
<rpc-reply message-id="101"
  xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    <configuration-text
      xmlns="http://www.ibm.com/netconf/1.0/config-text">
      <!-- configuration text... -->
    </configuration-text>
  </data>
</rpc-reply>
```

See [Table 53](#) for the tag elements and their values.

Table 53. *get Tag Element Values*

Tag Element	Description	Value
filter	Filter type.	subtree
configuration-text	Configuration in ISCLI format.	

<close-session>

Usage:

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <close-session/>
</rpc>
```

Response from the switch:

```
<rpc-reply message-id="101"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

<kill-session>

Usage:

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <kill-session>
    <session-id>4</session-id>
  </kill-session>
</rpc>
```

Response from the switch:

```
<rpc-reply message-id="101"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <ok/>
</rpc-reply>
```

See [Table 54](#) for the tag elements and their values.

Table 54. kill-session *Tag Element Values*

Tag Element	Description
session-id	ID number of the session to be killed

<get-configuration>

Usage:

```
<rpc message-id="101" xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <get-configuration database="committed" format="text"/>
</rpc>
```

Response from the switch:

```
<rpc-reply message-id="101"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <data>
    <configuration-text
      xmlns="http://www.ibm.com/netconf/1.0/config-text">
      <!-- configuration text... -->
    </configuration-text>
  </data>
</rpc-reply>
```

See [Table 55](#) for the tag elements and their values.

Table 55. get-configuration *Tag Element Values*

Tag Element	Description	Attributes
get-configuration	Retrieve the configuration.	database - supports only committed format - supports only text

<get-interface-information>

Usage:

```
<rpc message-id="101">
  <get-interface-information>
    <interface-name> port xx </interface-name>
    <brief/>
  </get-interface-information>
</rpc>
```

Response from switch:

- Port detail information

```
<rpc-reply message-id="101"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <interface-information>
    <physical-interface>
      <name></name>
      <admin-status></admin-status>
      <oper-status></oper-status>
      <local-index></local-index>
      <if-type></if-type>
      <link-level-type></link-level-type>
      <mtu></mtu>
      <speed></speed>
      <link-type></link-type>
      <traffic-statistics>
        <input-bytes></input-bytes>
        <output-bytes></output-bytes>
        <input-packets></input-packets>
        <output-packets></output-packets>
      </traffic-statistics>
      <input-error-list>
        <input-errors></input-errors>
        <framing-errors></framing-errors>
        <input-giants></input-giants>
        <input-discards></input-discards>
      </input-error-list>
      <output-error-list>
        <output-collisions></output-collisions>
        <output-errors></output-errors>
        <output-drops></output-drops>
      </output-error-list>
    </physical-interface>
  </interface-information>
</rpc-reply>
```

- IP detail information

```

<rpc-reply message-id="101"
xmlns="urn:ietf:params:xml:ns:netconf:base:1.0">
  <interface-information>
    <physical-interface>
      <logical-interface>
        <name></name>
        <local-index></local-index>
        <address-family>
          <address-family-name></address-family-name>
          <mtu></mtu>
          <interface-address>
            <ifa-destination></ifa-destination>
            <ifa-local></ifa-local>
            <ifa-broadcast></ifa-broadcast>
          </interface-address>
          </address-family>
        </logical-interface>
      </physical-interface>
    </interface-information>
  </rpc-reply>

```

See [Table 56](#) for the tag elements and their values.

Table 56. get-interface-information *Tag Element Values*

Tag Element	Description
interface-name	Interface name or number. You can use the tags <code>brief/</code> or <code>detail/</code> to specify the amount of information you need.
name	Name of the port or IP interface.
admin-status	Administration status of port interface; shutdown or no shutdown.
oper-status	Operational status of port interface; link-up or link-down.
local-index	Local index of port.
if-type	Type of port; GE, XGE.
link-level-type	Ethernet
mtu	9216 for port; 1500 for IP interface.
speed	Speed of port; 1000M, 10000M.
link-type	Type of duplex port; full, half.
input-bytes	Number of bytes received at the port.
output-bytes	Number of bytes sent from the port.
input-packets	Number of frames received at port.
output-packets	Number of frames sent out from the port.

Table 56. get-interface-information Tag Element Values

Tag Element	Description
input-errors	Sum of discarded frames and FCS Errors.
framing-errors	Number of failed frames received.
input-giants	Number of frames that are too long.
input-discards	Number of frames in discarding state.
output-collisions	Number of Ethernet collisions.
output-errors	Sum of the outgoing frame aborts and FCS errors.
output-drops	Number of frames dropped.
address-family-name	inet
ifa-destination	Protocol network address of the interface.
ifa-local	Protocol host address on the interface.
ifa-broadcast	Network broadcast address.

Part 8: Monitoring

The ability to monitor traffic passing through the G8296 can be invaluable for troubleshooting some types of networking problems. This sections cover the following monitoring features:

- Remote Monitoring (RMON)
- sFlow
- Port Mirroring

Chapter 40. Remote Monitoring

Remote Monitoring (RMON) allows network devices to exchange network monitoring data.

RMON allows the switch to perform the following functions:

- Track events and trigger alarms when a threshold is reached.
- Notify administrators by issuing a syslog message or SNMP trap.

RMON Overview

The RMON MIB provides an interface between the RMON agent on the switch and an RMON management application. The RMON MIB is described in RFC 1757.

The RMON standard defines objects that are suitable for the management of Ethernet networks. The RMON agent continuously collects statistics and proactively monitors switch performance. RMON allows you to monitor traffic flowing through the switch.

The switch supports the following RMON Groups, as described in RFC 1757:

- Group 1: Statistics
- Group 2: History
- Group 3: Alarms
- Group 9: Events

RMON Group 1—Statistics

The switch supports collection of Ethernet statistics as outlined in the RMON statistics MIB, in reference to etherStatsTable. You can configure RMON statistics on a per-port basis.

RMON statistics are sampled every second, and new data overwrites any old data on a given port.

Note: RMON port statistics must be enabled for the port before you can view RMON statistics.

Example Configuration

1. Enable RMON on a port.

```
RS8296(config)# interface port 1
RS8296(config-if)# rmon
```

2. View RMON statistics for the port.

```
RS8296(config-if)# show interface port 1 rmon-counters
-----
RMON statistics for port 1:
etherStatsDropEvents:                NA
etherStatsOctets:                    7305626
etherStatsPkts:                      48686
etherStatsBroadcastPkts:             4380
etherStatsMulticastPkts:             6612
etherStatsCRCAlignErrors:            22
etherStatsUndersizePkts:             0
etherStatsOversizePkts:              0
etherStatsFragments:                 2
etherStatsJabbers:                   0
etherStatsCollisions:                0
etherStatsPkts64Octets:               27445
etherStatsPkts65to127Octets:         12253
etherStatsPkts128to255Octets:        1046
etherStatsPkts256to511Octets:        619
etherStatsPkts512to1023Octets:       7283
etherStatsPkts1024to1518Octets:      38
```

RMON Group 2—History

The RMON History Group allows you to sample and archive Ethernet statistics for a specific interface during a specific time interval. History sampling is done per port.

Note: RMON port statistics must be enabled for the port before an RMON History Group can monitor the port.

Data is stored in *buckets*, which store data gathered during discreet sampling intervals. At each configured interval, the History index takes a sample of the current Ethernet statistics, and places them into a bucket. History data buckets reside in dynamic memory. When the switch is re-booted, the buckets are emptied.

Requested buckets are the number of buckets, or data slots, requested by the user for each History Group. Granted buckets are the number of buckets granted by the system, based on the amount of system memory available. The system grants a maximum of 50 buckets.

You can use an SNMP browser to view History samples.

History MIB Object ID

The type of data that can be sampled must be of an `ifIndex` object type, as described in RFC 1213 and RFC 1573. The most common data type for the History sample is as follows:

```
1.3.6.1.2.1.2.2.1.1.<x>
```

The last digit (*x*) represents the number of the port to monitor.

Configuring RMON History

Perform the following steps to configure RMON History on a port.

1. Enable RMON on a port.

```
RS8296(config)# interface port 1
RS8296(config-if)# rmon
RS8296(config-if)# exit
```

2. Configure the RMON History parameters for a port.

```
RS8296(config)# rmon history 1 interface-oid 1.3.6.1.2.1.2.2.1.1.<x>
RS8296(config)# rmon history 1 requested-buckets 30
RS8296(config)# rmon history 1 polling-interval 120
RS8296(config)# rmon history 1 owner "rmon port 1 history"
```

where `<x>` is the number of the port to monitor. For example, the full OID for port 1 would be:

```
1.3.6.1.2.1.2.2.1.1.129
```

3. View RMON history for the port.

```
RS8296(config)# show rmon history
RMON History group configuration:

Index          IFOID          Interval    Rbnum    Gbnum
-----
   1  1.3.6.1.2.1.2.2.1.1.129    120      30      30

Index          Owner
-----
   1  rmon port 1 history
```

RMON Group 3—Alarms

The RMON Alarm Group allows you to define a set of thresholds used to determine network performance. When a configured threshold is crossed, an alarm is generated. For example, you can configure the switch to issue an alarm if more than 1,000 CRC errors occur during a 10-minute time interval.

Each Alarm index consists of a variable to monitor, a sampling time interval, and parameters for rising and falling thresholds. The Alarm Group can be used to track rising or falling values for a MIB object. The object must be a counter, gauge, integer, or time interval.

Use one of the following commands to correlate an Alarm index to an Event index:

```
RS8296(config)# rmon alarm <alarm number> rising-crossing-index <event number>
RS8296(config)# rmon alarm <alarm number> falling-crossing-index <event number>
```

When the alarm threshold is reached, the corresponding event is triggered.

Alarm MIB objects

The most common data types used for alarm monitoring are `ifStats: errors`, `drops`, `bad CRCs`, and so on. These MIB Object Identifiers (OIDs) correlate to the ones tracked by the History Group. An example statistic follows:

```
1.3.6.1.2.1.2.2.1.1.129 - mgmt.icmp.icmpInMsgs
```

This value represents the alarm's MIB OID, as a string. Note that for non-tables, you must supply a `.0` to specify end node.

Configuring RMON Alarms

Configure the RMON Alarm parameters to track ICMP messages.

```
RS8296(config)# rmon alarm 1 oid 1.3.6.1.2.1.5.8.0
RS8296(config)# rmon alarm 1 alarm-type rising
RS8296(config)# rmon alarm 1 rising-crossing-index 110
RS8296(config)# rmon alarm 1 interval 60
RS8296(config)# rmon alarm 1 rising-limit 200
RS8296(config)# rmon alarm 1 sample delta
RS8296(config)# rmon alarm 1 owner "Alarm for icmpInEchos"
```

This configuration creates an RMON alarm that checks `icmpInEchos` on the switch once every minute. If the statistic exceeds 200 within a 60 second interval, an alarm is generated that triggers event index 110.

RMON Group 9—Events

The RMON Event Group allows you to define events that are triggered by alarms. An event can be a log message, an SNMP trap, or both.

When an alarm is generated, it triggers a corresponding event notification. Use the following commands to correlate an Event index to an alarm:

```
RS8296(config)# rmon alarm <alarm number> rising-crossing-index <event number>
RS8296(config)# rmon alarm <alarm number> falling-crossing-index <event number>
```

RMON events use SNMP and syslogs to send notifications. Therefore, an SNMP trap host must be configured for trap event notification to work properly.

RMON uses a syslog host to send syslog messages. Therefore, an existing syslog host must be configured for event log notification to work properly. Each log event generates a syslog of type RMON that corresponds to the event.

For example, to configure the RMON event parameters.

```
RS8296(config)# rmon event 110 type log
RS8296(config)# rmon event 110 description "SYSLOG_this_alarm"
RS8296(config)# rmon event 110 owner "log icmpInEchos alarm"
```

This configuration creates an RMON event that sends a syslog message each time it is triggered by an alarm.

Chapter 41. sFlow

The G8296 supports sFlow technology for monitoring traffic in data networks. The switch includes an embedded sFlow agent which can be configured to provide continuous monitoring information of IPv4 traffic to a central sFlow analyzer.

The switch is responsible only for forwarding sFlow information. A separate sFlow analyzer is required elsewhere on the network to interpret sFlow data.

Note: Lenovo Network Operating System 8.2 does not support IPv6 for sFlow.

sFlow Statistical Counters

The G8296 can be configured to send network statistics to an sFlow analyzer at regular intervals. For each port, a polling interval of 5 to 60 seconds can be configured, or 0 (the default) to disable this feature.

When polling is enabled, at the end of each configured polling interval, the G8296 reports general port statistics and port Ethernet statistics.

sFlow Network Sampling

In addition to statistical counters, the G8296 can be configured to collect periodic samples of the traffic data received on each port. For each sample, 128 bytes are copied, UDP-encapsulated, and sent to the configured sFlow analyzer.

For each port, the sFlow sampling rate can be configured to occur once each 256 to 65536 packets, or 0 to disable (the default). A sampling rate of 256 means that one sample will be taken for approximately every 256 packets received on the port. The sampling rate is statistical, however. It is possible to have slightly more or fewer samples sent to the analyzer for any specific group of packets (especially under low traffic conditions). The actual sample rate becomes most accurate over time, and under higher traffic flow.

sFlow sampling has the following restrictions:

- **Sample Rate**—The fastest sFlow sample rate is 1 out of every 256 packets.
- **ACLs**—sFlow sampling is performed before ACLs are processed. For ports configured both with sFlow sampling and one or more ACLs, sampling will occur regardless of the action of the ACL.
- **Port Mirroring**—sFlow sampling will not occur on mirrored traffic. If sFlow sampling is enabled on a port that is configured as a port monitor, the mirrored traffic will not be sampled.
- **Egress traffic**—sFlow sampling will not occur on egress traffic.

Note: Although sFlow sampling is not generally a CPU-intensive operation, configuring fast sampling rates (such as once every 256 packets) on ports under heavy traffic loads can cause switch CPU utilization to reach maximum. Use larger rate values for ports that experience heavy traffic.

sFlow Example Configuration

1. Specify the location of the sFlow analyzer (the server and optional port to which the sFlow information will be sent):

```
RS8296(config)# sflow server <IPv4 address>(sFlow server address)
RS8296(config)# sflow port <service port> (Set the optional service port)
RS8296(config)# sflow enable (Enable sFlow features)
```

By default, the switch uses established sFlow service port 6343.

To disable sFlow features across all ports, use the `no sflow enable` command.

2. On a per-port basis, define the statistics polling rate:

```
RS8296(config)# interface port <port>
RS8296(config-if)# sflow polling <polling rate>(Statistics polling rate)
```

Specify a polling rate between 5 and 60 seconds, or 0 to disable. By default, polling is 0 (disabled) for each port.

3. On a per-port basis, define the data sampling rate:

```
RS8296(config-if)# sflow sampling <sampling rate>(Data sampling rate)
```

Specify a sampling rate between 256 and 65536 packets, or 0 to disable. By default, the sampling rate is 0 (disabled) for each port.

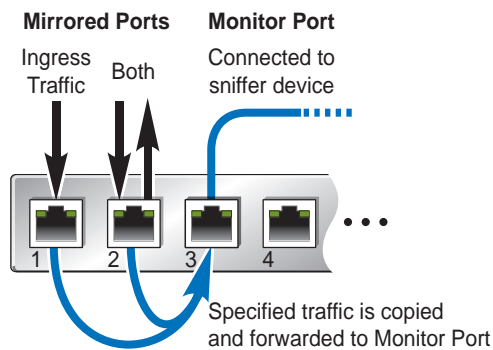
4. Save the configuration.

Chapter 42. Port Mirroring

The Lenovo Network Operating System port mirroring feature allows you to mirror (copy) the packets of a target port, and forward them to a monitoring port. Port mirroring functions for all layer 2 and layer 3 traffic on a port. This feature can be used as a troubleshooting tool or to enhance the security of your network. For example, an IDS server or other traffic sniffer device or analyzer can be connected to the monitoring port to detect intruders attacking the network.

The G8296 supports a “many to one” mirroring model. As shown in [Figure 54](#), selected traffic for ports 1 and 2 is being monitored by port 3. In the example, both ingress traffic and egress traffic on port 2 are copied and forwarded to the monitor. However, port 1 mirroring is configured so that only ingress traffic is copied and forwarded to the monitor. A device attached to port 3 can analyze the resulting mirrored traffic.

Figure 54. Mirroring Ports



The G8296 supports four monitor ports. Each monitor port can receive mirrored traffic from any number of target ports.

Networking OS does not support “one to many” or “many to many” mirroring models where traffic from a specific port traffic is copied to multiple monitor ports. For example, port 1 traffic cannot be monitored by both port 3 and 4 at the same time, nor can port 2 ingress traffic be monitored by a different port than its egress traffic.

Ingress and egress traffic is duplicated and sent to the monitor port after processing.

Configuring Port Mirroring

The following procedure may be used to configure port mirroring for the example shown in [Figure 54 on page 577](#):

1. Specify the monitoring port, the mirroring port(s), and the port-mirror direction.

```
RS8296(config)# port-mirroring monitor-port 3 mirroring-port 1 in
RS8296(config)# port-mirroring monitor-port 3 mirroring-port 2 both
```

2. Enable port mirroring.

```
RS8296(config)# port-mirroring enable
```

3. View the current configuration.

```
RS8296# show port-mirroring

Port Monitoring : Enabled

Monitoring Ports      Mirrored Ports
3                     1, in
                      2, both
```

Part 9: Appendices

- Glossary
- Getting help and technical assistance
- Notices

Appendix A. Glossary

CNA	Converged Network Adapter. A device used for I/O consolidation such as that in Converged Enhanced Ethernet (CEE) environments implementing Fibre Channel over Ethernet (FCoE). The CNA performs the duties of both a Network Interface Card (NIC) for Local Area Networks (LANs) and a Host Bus Adapter (HBA) for Storage Area Networks (SANs).
DIP	The destination IP address of a frame.
Dport	The destination port (application socket: for example, http-80/https-443/DNS-53)
HBA	Host Bus Adapter. An adapter or card that interfaces with device drivers in the host operating system and the storage target in a Storage Area Network (SAN). It is equivalent to a Network Interface Controller (NIC) from a Local Area Network (LAN).
NAT	Network Address Translation. Any time an IP address is changed from one source IP or destination IP address to another address, network address translation can be said to have taken place. In general, half NAT is when the destination IP or source IP address is changed from one address to another. Full NAT is when both addresses are changed from one address to another. No NAT is when neither source nor destination IP addresses are translated.
Preemption	In VRRP, preemption will cause a Virtual Router that has a lower priority to go into backup if a peer Virtual Router starts advertising with a higher priority.
Priority	In VRRP, the value given to a Virtual Router to determine its ranking with its peer(s). Minimum value is 1 and maximum value is 254. Default is 100. A higher number will win out for master designation.
Proto (Protocol)	The protocol of a frame. Can be any value represented by a 8-bit value in the IP header adherent to the IP specification (for example, TCP, UDP, OSPF, ICMP, and so on.)
SIP	The source IP address of a frame.
SPort	The source port (application socket: for example, HTTP-80/HTTPS-443/DNS-53).
Tracking	<p>In VRRP, a method to increase the priority of a virtual router and thus master designation (with preemption enabled). Tracking can be very valuable in an active/active configuration.</p> <p>You can track the following:</p> <ul style="list-style-type: none">● Active IP interfaces on the Web switch (increments priority by 2 for each)● Active ports on the same VLAN (increments priority by 2 for each)● Number of virtual routers in master mode on the switch
VIR	Virtual Interface Router. A VRRP address is an IP interface address shared between two or more virtual routers.
Virtual Router	A shared address between two devices utilizing VRRP, as defined in RFC 2338. One virtual router is associated with an IP interface. This is one of the IP interfaces that the switch is assigned. All IP interfaces on the G8296s must be in a VLAN. If there is more than one VLAN defined on the Web switch, then the VRRP broadcasts will only be sent out on the VLAN of which the associated IP interface is a member.

VRID	<p>Virtual Router Identifier. In VRRP, a numeric ID is used by each virtual router to create its MAC address and identify its peer for which it is sharing this VRRP address. The VRRP MAC address as defined in the RFC is 00-00-5E-00-01-<VRID>.</p> <p>If you have a VRRP address that two switches are sharing, then the VRID number needs to be identical on both switches so each virtual router on each switch knows with whom to share.</p>
VRRP	<p>Virtual Router Redundancy Protocol. A protocol that acts very similarly to Cisco's proprietary HSRP address sharing protocol. The reason for both of these protocols is so devices have a next hop or default gateway that is always available. Two or more devices sharing an IP interface are either advertising or listening for advertisements. These advertisements are sent via a broadcast message to an address such as 224.0.0.18.</p> <p>With VRRP, one switch is considered the master and the other the backup. The master is always advertising via the broadcasts. The backup switch is always listening for the broadcasts. If the master stops advertising, the backup will take over ownership of the VRRP IP and MAC addresses as defined by the specification. The switch announces this change in ownership to the devices around it by way of a Gratuitous ARP, and advertisements. If the backup switch didn't do the Gratuitous ARP the Layer 2 devices attached to the switch would not know that the MAC address had moved in the network. For a more detailed description, refer to RFC 2338.</p>

Appendix B. Getting help and technical assistance

If you need help, service, or technical assistance or just want more information about Lenovo products, you will find a wide variety of sources available from Lenovo to assist you.

Use this information to obtain additional information about Lenovo and Lenovo products, and determine what to do if you experience a problem with your Lenovo system or optional device.

Note: This section includes references to IBM web sites and information about obtaining service. IBM is Lenovo's preferred service provider for the System x, Flex System, and NeXtScale System products.

Before you call, make sure that you have taken these steps to try to solve the problem yourself.

If you believe that you require warranty service for your Lenovo product, the service technicians will be able to assist you more efficiently if you prepare before you call.

- Check all cables to make sure that they are connected.
- Check the power switches to make sure that the system and any optional devices are turned on.
- Check for updated software, firmware, and operating-system device drivers for your Lenovo product. The Lenovo Warranty terms and conditions state that you, the owner of the Lenovo product, are responsible for maintaining and updating all software and firmware for the product (unless it is covered by an additional maintenance contract). Your service technician will request that you upgrade your software and firmware if the problem has a documented solution within a software upgrade.
- If you have installed new hardware or software in your environment, check the [IBM ServerProven website](#) to make sure that the hardware and software is supported by your product.
- Go to the [IBM Support portal](#) to check for information to help you solve the problem.
- Gather the following information to provide to the service technician. This data will help the service technician quickly provide a solution to your problem and ensure that you receive the level of service for which you might have contracted.
 - Hardware and Software Maintenance agreement contract numbers, if applicable
 - Machine type number (if applicable—Lenovo 4-digit machine identifier)
 - Model number
 - Serial number
 - Current system UEFI and firmware levels
 - Other pertinent information such as error messages and logs

- Start the process of determining a solution to your problem by making the pertinent information available to the service technicians. The IBM service technicians can start working on your solution as soon as you have completed and submitted an Electronic Service Request.

You can solve many problems without outside assistance by following the troubleshooting procedures that Lenovo provides in the online help or in the Lenovo product documentation. The Lenovo product documentation also describes the diagnostic tests that you can perform. The documentation for most systems, operating systems, and programs contains troubleshooting procedures and explanations of error messages and error codes. If you suspect a software problem, see the documentation for the operating system or program.

Appendix C. Notices

Lenovo may not offer the products, services, or features discussed in this document in all countries. Consult your local Lenovo representative for information on the products and services currently available in your area.

Any reference to a Lenovo product, program, or service is not intended to state or imply that only that Lenovo product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any Lenovo intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any other product, program, or service.

Lenovo may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

Lenovo (United States), Inc.
1009 Think Place - Building One
Morrisville, NC 27560
U.S.A.

Attention: Lenovo Director of Licensing

LENOVO PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some jurisdictions do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. Lenovo may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

The products described in this document are not intended for use in implantation or other life support applications where malfunction may result in injury or death to persons. The information contained in this document does not affect or change Lenovo product specifications or warranties.

Nothing in this document shall operate as an express or implied license or indemnity under the intellectual property rights of Lenovo or third parties. All information contained in this document was obtained in specific environments and is presented as an illustration. The result obtained in other operating environments may vary.

Lenovo may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Any references in this publication to non-Lenovo Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this Lenovo product, and use of those Web sites is at your own risk.

Any performance data contained herein was determined in a controlled environment. Therefore, the result obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Trademarks

Lenovo, the Lenovo logo, Flex System, System x, NeXtScale System, and X-Architecture are trademarks of Lenovo in the United States, other countries, or both.

Intel and Intel Xeon are trademarks of Intel Corporation in the United States, other countries, or both.

Internet Explorer, Microsoft, and Windows are trademarks of the Microsoft group of companies.

Linux is a registered trademark of Linus Torvalds.

Other company, product, or service names may be trademarks or service marks of others.

Important Notes

Processor speed indicates the internal clock speed of the microprocessor; other factors also affect application performance.

CD or DVD drive speed is the variable read rate. Actual speeds vary and are often less than the possible maximum.

When referring to processor storage, real and virtual storage, or channel volume, KB stands for 1 024 bytes, MB stands for 1 048 576 bytes, and GB stands for 1 073 741 824 bytes.

When referring to hard disk drive capacity or communications volume, MB stands for 1 000 000 bytes, and GB stands for 1 000 000 000 bytes. Total user-accessible capacity can vary depending on operating environments.

Maximum internal hard disk drive capacities assume the replacement of any standard hard disk drives and population of all hard-disk-drive bays with the largest currently supported drives that are available from Lenovo.

Maximum memory might require replacement of the standard memory with an optional memory module.

Each solid-state memory cell has an intrinsic, finite number of write cycles that the cell can incur. Therefore, a solid-state device has a maximum number of write cycles that it can be subjected to, expressed as total bytes written (TBW). A device that has exceeded this limit might fail to respond to system-generated commands or might be incapable of being written to. Lenovo is not responsible for replacement of a device that has exceeded its maximum guaranteed number of program/erase cycles, as documented in the Official Published Specifications for the device.

Lenovo makes no representations or warranties with respect to non-Lenovo products. Support (if any) for the non-Lenovo products is provided by the third party, not Lenovo.

Some software might differ from its retail version (if available) and might not include user manuals or all program functionality.

Recycling Information

Lenovo encourages owners of information technology (IT) equipment to responsibly recycle their equipment when it is no longer needed. Lenovo offers a variety of programs and services to assist equipment owners in recycling their IT products. For information on recycling Lenovo products, go to:

<http://www.lenovo.com/recycling>

Particulate Contamination

Attention: Airborne particulates (including metal flakes or particles) and reactive gases acting alone or in combination with other environmental factors such as humidity or temperature might pose a risk to the device that is described in this document.

Risks that are posed by the presence of excessive particulate levels or concentrations of harmful gases include damage that might cause the device to malfunction or cease functioning altogether. This specification sets forth limits for particulates and gases that are intended to avoid such damage. The limits must not be viewed or used as definitive limits, because numerous other factors, such as temperature or moisture content of the air, can influence the impact of particulates or environmental corrosives and gaseous contaminant transfer. In the absence of specific limits that are set forth in this document, you must implement practices that maintain particulate and gas levels that are consistent with the protection of human health and safety. If Lenovo determines that the levels of particulates or gases in your environment have caused damage to the device, Lenovo may condition provision of repair or replacement of devices or parts on implementation of appropriate remedial measures to mitigate such environmental contamination. Implementation of such remedial measures is a customer responsibility..

Contaminant	Limits
Particulate	<ul style="list-style-type: none"> The room air must be continuously filtered with 40% atmospheric dust spot efficiency (MERV 9) according to ASHRAE Standard 52.2¹. Air that enters a data center must be filtered to 99.97% efficiency or greater, using high-efficiency particulate air (HEPA) filters that meet MIL-STD-282. The deliquescent relative humidity of the particulate contamination must be more than 60%². The room must be free of conductive contamination such as zinc whiskers.
Gaseous	<ul style="list-style-type: none"> Copper: Class G1 as per ANSI/ISA 71.04-1985³ Silver: Corrosion rate of less than 300 Å in 30 days

¹ ASHRAE 52.2-2008 - *Method of Testing General Ventilation Air-Cleaning Devices for Removal Efficiency by Particle Size*. Atlanta: American Society of Heating, Refrigerating and Air-Conditioning Engineers, Inc.

² The deliquescent relative humidity of particulate contamination is the relative humidity at which the dust absorbs enough water to become wet and promote ionic conduction.

³ ANSI/ISA-71.04-1985. *Environmental conditions for process measurement and control systems: Airborne contaminants*. Instrument Society of America, Research Triangle Park, North Carolina, U.S.A.

Telecommunication Regulatory Statement

This product may not be certified in your country for connection by any means whatsoever to interfaces of public telecommunications networks. Further certification may be required by law prior to making any such connection. Contact a Lenovo representative or reseller for any questions.

Electronic Emission Notices

When you attach a monitor to the equipment, you must use the designated monitor cable and any interference suppression devices that are supplied with the monitor.

Federal Communications Commission (FCC) Statement

Note: This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference, in which case the user will be required to correct the interference at his own expense.

Properly shielded and grounded cables and connectors must be used to meet FCC emission limits. Lenovo is not responsible for any radio or television interference caused by using other than recommended cables and connectors or by unauthorized changes or modifications to this equipment. Unauthorized changes or modifications could void the user's authority to operate the equipment.

This device complies with Part 15 of the FCC Rules. Operation is subject to the following two conditions: (1) this device may not cause harmful interference, and (2) this device must accept any interference received, including interference that might cause undesired operation.

Industry Canada Class A Emission Compliance Statement

This Class A digital apparatus complies with Canadian ICES-003.

Avis de Conformité à la Réglementation d'Industrie Canada

Cet appareil numérique de la classe A est conforme à la norme NMB-003 du Canada.

Australia and New Zealand Class A Statement

Attention: This is a Class A product. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

European Union EMC Directive Conformance Statement

This product is in conformity with the protection requirements of EU Council Directive 2004/108/EC on the approximation of the laws of the Member States relating to electromagnetic compatibility. Lenovo cannot accept responsibility for any failure to satisfy the protection requirements resulting from a non-recommended modification of the product, including the installation of option cards from other manufacturers.

This product has been tested and found to comply with the limits for Class A Information Technology Equipment according to European Standard EN 55022. The limits for Class A equipment were derived for commercial and industrial environments to provide reasonable protection against interference with licensed communication equipment.

Lenovo, Einsteinova 21, 851 01 Bratislava, Slovakia

Germany Class A Statement

Zulassungsbescheinigung laut dem Deutschen Gesetz über die elektromagnetische Verträglichkeit von Betriebsmitteln, EMVG vom 20. Juli 2007 (früher Gesetz über die elektromagnetische Verträglichkeit von Geräten), bzw. der EMV EG Richtlinie 2004/108/EC (früher 89/336/EWG), für Geräte der Klasse A.

Dieses Gerät ist berechtigt, in Übereinstimmung mit dem Deutschen EMVG das EG-Konformitätszeichen - CE - zu führen. Verantwortlich für die Konformitätserklärung nach Paragraph 5 des EMVG ist die Lenovo (Deutschland) GmbH, Gropiusplatz 10, D-70563 Stuttgart.

Informationen in Hinsicht EMVG Paragraph 4 Abs. (1) 4:

Das Gerät erfüllt die Schutzanforderungen nach EN 55024 und EN 55022 Klasse A.

Nach der EN 55022: "Dies ist eine Einrichtung der Klasse A. Diese Einrichtung kann im Wohnbereich Funkstörungen verursachen; in diesem Fall kann vom Betreiber verlangt werden, angemessene Maßnahmen durchzuführen und dafür aufzukommen."

Nach dem EMVG: Dieses Produkt entspricht den Schutzanforderungen der EU-Richtlinie 2004/108/EG (früher 89/336/EWG) zur Angleichung der Rechtsvorschriften über die elektromagnetische Verträglichkeit in den EU-Mitgliedsstaaten und hält die Grenzwerte der EN 55022 Klasse A ein.

Um dieses sicherzustellen, sind die Geräte wie in den Handbüchern beschrieben zu installieren und zu betreiben. Des Weiteren dürfen auch nur von der Lenovo empfohlene Kabel angeschlossen werden. Lenovo übernimmt keine Verantwortung für die Einhaltung der Schutzanforderungen, wenn das Produkt ohne Zustimmung der Lenovo verändert bzw. wenn Erweiterungskomponenten von Fremdherstellern ohne Empfehlung der Lenovo gesteckt/eingebaut werden.

Deutschland:

Einhaltung des Gesetzes über die elektromagnetische Verträglichkeit von Betriebsmitteln

Dieses Produkt entspricht dem "Gesetz über die elektromagnetische Verträglichkeit von Betriebsmitteln" EMVG (früher "Gesetz über die elektromagnetische Verträglichkeit von Geräten"). Dies ist die Umsetzung der EU-Richtlinie 2004/108/EG (früher 89/336/EWG) in der Bundesrepublik Deutschland.

Zulassungsbescheinigung laut dem Deutschen Gesetz über die elektromagnetische Verträglichkeit von Betriebsmitteln, EMVG vom 20. Juli 2007 (früher Gesetz über die elektromagnetische Verträglichkeit von Geräten), bzw. der EMV EG Richtlinie 2004/108/EC (früher 89/336/EWG), für Geräte der Klasse A.

Dieses Gerät ist berechtigt, in Übereinstimmung mit dem Deutschen EMVG das EG-Konformitätszeichen - CE - zu führen. Verantwortlich für die Konformitätserklärung nach Paragraph 5 des EMVG ist die Lenovo (Deutschland) GmbH, Gropiusplatz 10, D-70563 Stuttgart.

Informationen in Hinsicht EMVG Paragraph 4 Abs. (1) 4:

Das Gerät erfüllt die Schutzanforderungen nach EN 55024 und EN 55022 Klasse A.

Nach der EN 55022: "Dies ist eine Einrichtung der Klasse A. Diese Einrichtung kann im Wohnbereich Funkstörungen verursachen; in diesem Fall kann vom Betreiber verlangt werden, angemessene Maßnahmen durchzuführen und dafür aufzukommen."

Nach dem EMVG: "Geräte dürfen an Orten, für die sie nicht ausreichend entstört sind, nur mit besonderer Genehmigung des Bundesministers für Post und Telekommunikation oder des Bundesamtes für Post und Telekommunikation betrieben werden. Die Genehmigung wird erteilt, wenn keine elektromagnetischen Störungen zu erwarten sind." (Auszug aus dem EMVG, Paragraph 3, Abs. 4). Dieses Genehmigungsverfahren ist nach Paragraph 9 EMVG in Verbindung mit der entsprechenden Kostenverordnung (Amtsblatt 14/93) kostenpflichtig.

Anmerkung: Um die Einhaltung des EMVG sicherzustellen sind die Geräte, wie in den Handbüchern angegeben, zu installieren und zu betreiben.

Japan VCCI Class A Statement

この装置は、クラス A 情報技術装置です。この装置を家庭環境で使用すると電波妨害を引き起こすことがあります。この場合には使用者が適切な対策を講ずるよう要求されることがあります。 VCCI-A

This is a Class A product based on the standard of the Voluntary Control Council for Interference (VCCI). If this equipment is used in a domestic environment, radio interference may occur, in which case the user may be required to take corrective actions.

Japan Electronics and Information Technology Industries Association (JEITA) Statement

高調波ガイドライン適合品

Japan Electronics and Information Technology Industries Association (JEITA)
Confirmed Harmonics Guidelines (products less than or equal to 20 A per phase)

高調波ガイドライン準用品

Japan Electronics and Information Technology Industries Association (JEITA)
Confirmed Harmonics Guidelines with Modifications (products greater than 20 A per phase).

Korea Communications Commission (KCC) Statement

이 기기는 업무용(A급)으로 전자파적합기기로
서 판매자 또는 사용자는 이 점을 주의하시기
바라며, 가정외의 지역에서 사용하는 것을 목
적으로 합니다.

This is electromagnetic wave compatibility equipment for business (Type A).
Sellers and users need to pay attention to it. This is for any areas other than home.

Russia Electromagnetic Interference (EMI) Class A statement

ВНИМАНИЕ! Настоящее изделие относится к классу А.
В жилых помещениях оно может создавать радиопомехи, для
снижения которых необходимы дополнительные меры

People's Republic of China Class A electronic emission statement

中华人民共和国“A类”警告声明

声明

此为A级产品，在生活环境中，该产品可能会造成无线电干扰。在这种情况下，可能需要用户对其干扰采取切实可行的措施。

Taiwan Class A compliance statement

警告使用者：
這是甲類的資訊產品，在居住的環境中使用時，可能會造成射頻干擾，在這種情況下，使用者會被要求採取某些適當的對策。

Index

Symbols

[] 25

Numerics

40GbE ports 132
802.1p QoS 270
802.1Q VLAN tagging 122, 286
802.1Qaz ETS 286
802.1Qbb PFC 283
802.1Qbg. *See* EVB
802.3x flow control 283

A

Access Control List (ACL) 191
Access Control Lists. *See* ACLs.
accessing the switch
 Browser-based Interface 30, 34
 LDAP authentication 92
 RADIUS authentication 84
 security 75, 83
 TACACS+ 88
ACL metering 192
ACLs 103, 191
 FCoE 273
 FIP snooping 267, 268, 272
 Policy-based routing 341
active-active redundancy 503
administrator account 40, 86
advertise flag (DCBX) 293
aggregating routes 434
 example 441
AH 372
anycast address, IPv6 360
application ports 105
assistance, getting 579, 583
Australia Class A statement 592
authenticating, in OSPF 455
Authentication Header (AH) 372
autoconfiguration
 IPv6 362
 link 54
auto-negotiation
 setup 54
autonomous systems (AS) 448

B

bandwidth allocation 270, 288
BBI 30
 See Browser-Based Interface 449
Bootstrap Router, PIM 480

Border Gateway Protocol (BGP) 423
 attributes 436
 failover configuration 439
 route aggregation 434
 route maps 430
 selecting route paths 438
Bridge Protocol Data Unit (BPDU) 148
broadcast domains 119
broadcast storm control 115
Browser-Based Interface 30, 449
BSR, PIM 480

C

Canada Class A electronic emission statement 592
CEE 265, 269
 802.1p QoS 270
 bandwidth allocation 270
 DCBX 266, 269, 292
 ETS 265, 270, 286
 FCoE 268, 269
 LLDP 269
 on/off 269
 PFC 265, 271, 283
 priority groups 287
China Class A electronic emission statement 597
Cisco EtherChannel 135, 137
CIST 162
Class A electronic emission notice 592
Class of Service queue/COS queue 200
CNA 268
command conventions 25
Command Line Interface 449
Command-Line Interface (CLI) 49
Community VLAN 129
component, PIM 476
configuration rules
 CEE 269
 FCoE 268
 Trunking 135
configuring
 BGP failover 439
 DCBX 294
 ETS 289
 FIP snooping 275
 IP routing 328
 OSPF 459
 PFC 284
 port trunking 136
 spanning tree groups 158, 164
contamination, particulate and gaseous 590
Converged Enhanced Ethernet. *See* CEE.
Converged Network Adapter. *See* CNA.

D

Data Center Bridging Capability Exchange. *See* DCBX.
date
 setup 52
DCBX 266, 269, 292
default gateway 327
 configuration example 330
default password 40, 86
default route
 OSPF 453
Dense Mode, PIM 475, 476, 484
Designated Router, PIM 474, 479
Differentiated Services Code Point (DSCP) 193
digital certificate 373
 generating 375
 importing 374
downloading software 66
DR, PIM 474, 479
DSCP 193

E

EAPoL 96
ECMP route hashing 337
ECP 297
Edge Control Protocol. *See* ECP
Edge Virtual Bridging. *See* EVB
electronic emission Class A notice 592
Encapsulating Security Payload (ESP) 372
End user access control
 configuring 80
Enhanced Transmission Selection. *See* ETS.
ENodes 268, 272
ESP 372
EtherChannel 134
 as used with port trunking 135, 137
Ethernet Nodes (FCoE). *See* ENodes.
ETS 265, 270, 286
 bandwidth allocation 270, 288
 configuring 289
 DCBX 294
 PGID 270, 287
 priority groups 287
 priority values 288
European Union EMC Directive conformance statement 592
EVB 297
Extensible Authentication Protocol over LAN 96
external routing 424, 448

F

factory default configuration 42, 50
failover 493
 overview 503
FC-BB-5 267
FCC Class A notice 592
FCC, Class A 592

FCF 265, 268, 272, 277
 detection mode 273
FCoE 265, 267
 CEE 268, 269
 CNA 268
 ENodes 268
 FCF 268
 FIP snooping 265, 267, 268, 272
 FLOGI 273
 point-to-point links 267
 requirements 268
 SAN 267, 269
 topology 267
 VLANs 274
FCoE Forwarder. *See* FCF.
FCoE Initialization Protocol snooping. *See* FIP snooping.
Fibre Channel over Ethernet. *See* FCoE.
Final Steps 61
FIP snooping 265, 267, 268, 272
 ACL rules 273
 ENode mode 273
 FCF mode 273
 timeout 273
first-time configuration 42, 49 to ??
FLOGI 273
flow control 283
 setup 54
frame size 120
frame tagging. *See* VLANs, tagging.

G

gaseous contamination 590
gateway. *See* default gateway.
Germany Class A statement 593
getting help 583

H

help
 sources of 583
help, getting 583
high-availability 499
Host routes
 OSPF 457
Hot Links 491
HP-OpenView 37, 527

I

IBM DirectorSNMP
 IBM Director 37, 527
ICMP 104

- IEEE standards
 - 802.1D 146
 - 802.1p 199
 - 802.1Q 122
 - 802.1Qaz 286
 - 802.1Qbb 283
 - 802.1Qbg 297
 - 802.1s 162
 - 802.1x 96
 - 802.3x 283
- IGMP 104, 387
 - PIM 482
 - Querier 391, 418
- IGMP Relay 404
- IGMPv3 392
- IKEv2 372
 - digital certificate 373, 374, 375
 - presared key 373, 375
- IKEv2 proposal 374
- image
 - downloading 66
- INCITS T11.3 267
- incoming route maps 431
- internal routing 424, 448
- Internet Group Management Protocol (IGMP) 387
- Internet Key Exchange Version 2 (IKEv2) 372
- Internet Protocol Security
 - See also IPsec 371
- IP address 57
 - IP interface 57
 - routing example 328
- IP configuration via setup 57
- IP interfaces 57
 - example configuration 328, 330
- IP routing 57
 - cross-subnet example 326
 - default gateway configuration 330
 - IP interface configuration 328, 330
 - IP subnets 326
 - NAT mapping 332
 - subnet configuration example 327
 - switch-based topology 327
- IP subnet mask 57
- IP subnets 327
 - routing 326, 327
 - VLANs 119
- IPsec 371
 - key policy 375
 - maximum traffic load 373
- IPv6 addressing 357, 359
- ISL Trunking 134
- Isolated VLAN 129

J

- Japan Class A electronic emission statement 594
- Japan Electronics and Information Technology Industries Association statement 595

- JEITA statement 595
- jumbo frames 120

K

- Korea Class A electronic emission statement 595

L

- LACP 138
- Layer 2 Failover 493
- LDAP
 - authentication 92
- Link Aggregation Control Protocol 138
- Link Layer Discovery Protocol 515
- LLDP 269, 292, 515
- logical segment. *See* IP subnets.
- lossless Ethernet 267, 269
- LSAs 446

M

- manual style conventions 25
- Maximum Transmission Unit 120
- meter 107
- meter (ACL) 192
- mirroring ports 577
- modes, PIM 475
- monitoring ports 577
- MSTP Multiple Spanning Tree Protocol (MSTP) 162
- MTU 120
- multi-links between switches
 - using port trunking 131
- multiple spanning tree groups 153
- Multiple Spanning Tree Protocol 162

N

- Neighbor Discovery, IPv6 364
- network component, PIM 476
- Network Load Balancing, *See* NLB,
- network management 30, 37, 527
- New Zealand Class A statement 592
- NLB 303
- notes, important 588
- notices 585

O

OSPF

- area types 444
 - authentication 455
 - configuration examples 460
 - default route 453
 - external routes 458
 - filtering criteria 104
 - host routes 457
 - link state database 446
 - neighbors 446
 - overview 444
 - redistributing routes 434
 - route maps 430, 432
 - route summarization 452
 - router ID 454
 - virtual link 454
- outgoing route maps 431

P

- packet size 120
- particulate contamination 590
- password
 - administrator account 40, 86
 - default 40, 86
 - user account 40, 86
- passwords 40
- payload size 120
- PBR. *See Policy-Based Routing*
- People's Republic of China Class A electronic emission statement 597
- Per Hop Behavior (PHB)PHB 194
- PFC 265, 271, 283
 - DCBX 293
- PGID 270, 287
- PIM 473 to 485
 - Bootstrap Router (BSR) 480
 - component 476
 - Dense Mode 475, 476, 484
 - Designated Router (DR) 474, 479
 - examples 483 to 485
 - IGMP 482
 - modes 475, 476
 - overview 474
 - Rendezvous Point (RP) 474, 479
 - Sparse Mode 474, 475, 476
- PIM-DM 475, 476, 484
- PIM-SM 474, 475, 476
- Policy-Based Routing 341
 - Health Check 343
- port flow control. *See flow control.*
- port mirroring 577
- port modes 132
- Port Trunking 135

- port trunking
 - configuration example 136
 - description 137
 - EtherChannel 134
- ports
 - configuration 54
 - for services 105
 - monitoring 577
 - physical. *See switch ports.*
- preshared key 373
 - enabling 375
- priority groups 287
- priority value (802.1p) 271, 286
- Priority-based Flow Control. *See PFC.*
- Private VLANs 129
 - Community VLAN 129
 - Isolated VLAN 129
- promiscuous port 129
- Protocol Independent Multicast (see PIM) 473 to 485
- protocol types 104
- PVID (port VLAN ID) 121

Q

- QoS 189, 545
- QSFP+ 132
- Quality of Service 189, 545
- Querier (IGMP) 391, 418

R

- RADIUS
 - authentication 84
 - port 1812 and 1645 105
 - port 1813 105
 - SSH/SCP 79
- Rapid Spanning Tree Protocol (RSTP) 160
- Rapid Spanning Tree Protocol (RSTP)RSTP 160
- receive flow control 54
- redistributing routes 434, 441
- redundancy
 - active-active 503
- re-mark 107, 192
- Rendezvous Point, PIM 474, 479
- restarting switch setup 51
- RIP (Routing Information Protocol)
 - advertisements 382
 - distance vector protocol 382
 - hop count 382
 - TCP/IP route information 22, 381
 - version 1 382
- RMON alarms 572
- RMON events 573
- RMON History 571
- RMON statistics 570
- route aggregation 434, 441

- route maps 430
 - configuring 432
 - incoming and outgoing 431
- route paths in BGP 438
- Routed Ports 349
- Router ID
 - OSPF 454
- routers 326, 330
 - border 448
 - peer 448
 - port trunking 134
 - switch-based routing topology 327
- routes, advertising 448
- routing 424
 - internal and external 448
- Routing Information Protocol. *See* RIP
- RP candidate, PIM 474, 479
- RSA keys 79
- RSTP 160
- Russia Class A electronic emission statement 596
- rx flow control 54

S

- SA 372
- SAN 267, 269
- security
 - LDAP authentication 92
 - port mirroring 577
 - RADIUS authentication 84
 - TACACS+ 88
 - VLANs 119
- security association (SA) 372
- segmentation. *See* IP subnets.
- segments. *See* IP subnets.
- server ports 242
- service and support
 - before you call 583
- service ports 105
- setup facility 42, 49
 - IP configuration 57
 - IP subnet mask 57
 - port auto-negotiation mode 54
 - port configuration 54
 - port flow control 54
 - restarting 51
 - Spanning-Tree Protocol 53
 - starting 50
 - stopping 51
 - system date 52
 - system time 52
 - VLAN name 56
 - VLAN tagging 54
 - VLANs 56
- SNMP 30, 37, 449, 527
 - HP-OpenView 37, 527
- SNMP Agent 527

- software
 - image 65
- Source-Specific MulticastSSM 392
- Spanning-Tree Protocol
 - multiple instances 153
 - setup (on/off) 53
- Sparse Mode, PIM 474, 475, 476
- SSH/SCP
 - configuring 76
 - RSA host and server keys 79
- starting switch setup 50
- Static ARP 303
- stopping switch setup 51
- Storage Area Network. *See* SAN.
- subnet mask 57
- subnets 57
- summarizing routes 452
- switch failover 503
- switch ports
 - VLAN membership 121

T

- TACACS+ 88
- tagging. *See* VLANs tagging.
- Taiwan Class A electronic emission statement 598
- TCP 104
- technical assistance 583
- technical terms
 - port VLAN identifier (PVID) 122
 - tagged frame 122
 - tagged member 122
 - untagged frame 122
 - untagged member 122
 - VLAN identifier (VID) 122
- Telnet support
 - optional setup for Telnet support 62
- text conventions 25
- time
 - setup 52
- trademarks 587
- transmit flow control 54
- Trunking
 - configuration rules 135
- tx flow control 54
- typographic conventions 25

U

- UDP 104
- United States FCC Class A notice 592
- upgrade, switch software 65
- uplink ports 242
- USB drive 68
- user account 40, 86

V

- VDP 297
- vDS. *See virtual Distributed Switch*
- VEB 297
- VEPA 297
- virtual Distributed Switch 250
- Virtual Ethernet Bridging. *See VEB*
- Virtual Ethernet Port Aggregator. *See VEPA*
- virtual interface router (VIR) 500
- virtual link, OSPF 454
- Virtual Local Area Networks. *See VLANs.*
- virtual router group 503
- virtual router ID numbering 505
- Virtual Station Interface. *See VSI*
- VLAN tagging
 - setup 54
- VLANs 57
 - broadcast domains 119
 - default PVID 121
 - example showing multiple VLANs 126
 - FCoE 274
 - ID numbers 120
 - interface 57
 - IP interface configuration 330
 - multiple spanning trees 147
 - multiple VLANs 122
 - name setup 56
 - port members 121
 - PVID 121
 - routing 328
 - security 119
 - setup 56
 - Spanning-Tree Protocol 147
 - tagging 54, 122
 - topologies 126
- VRRP (Virtual Router Redundancy Protocol)
 - active-active redundancy 503
 - overview 500
 - virtual interface router 500
 - virtual router ID numbering 505
 - vrid 500
- VSI 297
- VSI Database. *See VSIDB*
- VSI Discovery and Configuration Protocol. *See VDP*
- VSIDB 298

W

- willing flag (DCBX) 293