HITACHI
Inspire the Next

Application Guide

Networking OS 7.8 for 1/10Gb LAN Switch Module

FASTFIND LINKS

Product Version

Getting Help

Contents

Hitachi, Ltd., reserves the right to make changes to this document at any time without notice and assumes no responsibility for its use. This document contains the most current information available at the time of publication. When new or revised information becomes available, this entire document will be updated and distributed to all registered users.

Some of the features described in this document might not be currently available. Refer to the most recent product announcement for information about feature and product availability, or contact your reseller.

Contents

Preface v

Intended Audience vi
Product Version vi
Release Notes vi
Referenced Documents vi
Document Conventions vii
Convention for storage capacity values viii
Getting Help...viii

Getting Started 1-1

Securing the Switch 2-1

Switch Basics.. 3-1

Advanced Switching Features 4-1

IP Routing 5-1

High Availability Fundamentals 6-1

Network-Management... 7-1

Monitoring.. 8-1

Glossary Glossary-1

Preface

This document describes how to use the Hitachi BladeSymphony 1/10Gb LAN Switch Module. The Networking OS 7.8 Application Guide describes how to configure and use the Networking OS 7.8 software on Hitachi BladeSymphony 1/10Gb LAN Switch Module (referred to as 1/10Gb LAN Switch Module throughout this document).

This preface includes the following information:

- □ Intended Audience
- □ Product Version
- □ Release Notes
- □ Referenced Documents
- □ Document Conventions
- □ Convention for storage capacity values
- □ Getting Help

Intended Audience

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.

Product Version

This document revision applies to 1/10Gb LAN Switch Module version Networking OS 7.8.

Release Notes

Read the release notes before installing and using this product. They may contain requirements or restrictions that are not fully described in this document or updates or corrections to this document.

Referenced Documents

1/10Gb LAN Switch Module documents:

- Networking OS Command Reference
- Networking OS Browser-Based Interface Quick Guide

Document Conventions

This document uses the following typographic conventions:

Convention	Description
Regular text bold	In text: keyboard key, parameter name, property name, hardware labels, hardware button, hardware switch. In a procedure: user interface item
Italic	Variable, emphasis, reference to document title, called-out term
`Screen text`	Command name and option, drive name, file name, folder name, directory name, code, file content, system and application output, user input
< > (angled brackets)	Variable (used when italic is not enough to identify variable).
[ ] (square bracket)	Optional values
{ } braces	Required or expected value
| vertical bar	Choice between two or more options or arguments
_(underline)	Default value, for example, [a | b]

This document uses the following icons to draw attention to information:

Icon	Meaning	Description
⚠	WARNING	This indicates the presence of a potential risk that might cause death or severe injury.
⚠	CAUTION	This indicates the presence of a potential risk that might cause relatively mild or moderate injury.
NOTICE	NOTICE	This indicates the presence of a potential risk that might cause severe damage to the equipment and/or damage to surrounding properties.
Note	Note	This indicates notes not directly related to injury or severe damage to equipment.
Tip	Tip	This indicates advice on how to make the best use of the equipment.

Convention for storage capacity values

Physical storage capacity values (for example, disk drive capacity) are calculated based on the following values:

Physical capacity unit	Value
1 kilobyte (KB)	1,000 (10^3) bytes
1 megabyte (MB)	1,000 KB or $1,000^2$ bytes
1 gigabyte (GB)	1,000 MB or $1,000^3$ bytes
1 terabyte (TB)	1,000 GB or $1,000^4$ bytes
1 petabyte (PB)	1,000 TB or $1,000^5$ bytes
1 exabyte (EB)	1,000 PB or $1,000^6$ bytes

Logical storage capacity values (for example, logical device capacity) are calculated based on the following values:

Logical capacity unit	Value
1 block	512 bytes
1 KB	1,024 (2^{10}) bytes
1 MB	1,024 KB or $1,024^2$ bytes
1 GB	1,024 MB or $1,024^3$ bytes
1 TB	1,024 GB or $1,024^4$ bytes
1 PB	1,024 TB or $1,024^5$ bytes
1 EB	1,024 PB or $1,024^6$ bytes

Getting Help

If you need technical support, please contact Hitachi Solution Support Center or your reseller.

Thank you!

1

Getting Started

This material is intended to help those new to Networking OS products with the basics of switch management.

- □ Switch Administration
- □ Initial Setup
- □ Service Location Protocol
- □ System License Keys

Switch Administration

Your 1/10Gb LAN Switch Module 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 Networking OS switching software included in 1/10Gb LAN Switch Module 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

The switch software 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:
The Hitachi BladeSymphony Management Module tools for general chassis management

- 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.

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.

Management Module

1/10Gb LAN Switch Module is an integral subsystem within the overall Hitachi BladeSymphony. The Hitachi BladeSymphony chassis also includes a Management Module as the central element for overall chassis management and control. Using the tools available through the Management Module, the administrator can configure many of 1/10Gb LAN Switch Module features and can also access other 1/10Gb LAN Switch Module administration interfaces.

For more information, see “Using the Management Module” on page 1-4.

Command Line Interface

The Command Line Interface (CLI) provides a simple, direct method for switch administration. Using a basic terminal, you can issue commands that allow you to view detailed information and statistics about the switch, and to perform any necessary configuration and switch software maintenance.

You can establish a connection to the CLI in any of the following ways:

- Serial connection via the serial port on 1/10Gb LAN Switch Module (this option is always avail- able)

- Telnet connection over the network

- SSH connection over the network

Browser-Based Interface

The Browser-based Interface (BBI) provides access to the common configuration, management and operation features of 1/10Gb LAN Switch Module through your Web browser.

For more information, refer to the Networking OS 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 1-10), 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.

Using the Management Module

1/10Gb LAN Switch Module is an integral subsystem within the overall Hitachi BladeSymphony. The Hitachi BladeSymphony chassis includes a Management Module as the central element for overall chassis management and control.

1/10Gb LAN Switch Module uses port 66 (MGT1) to communicate with the Management Module(s). Even when 1/10Gb LAN Switch Module is in a factory default configuration, you can use the 1Gb Ethernet port on each Management Module to configure and manage 1/10Gb LAN Switch Module.

Factory-Default vs. Management Module-Assigned IP Addresses

Each 1/10Gb LAN Switch Module must be assigned its own Internet Protocol version 4 (IPv4) address, which is used for communication with an SNMP network manager or other transmission control protocol/Internet Protocol (TCP/IP) applications (for example, BOOTP or TFTP). The factory-default IPv4 address is 0.0.0.0. The Management Module assigns an IPv4 address of 0.0.0.0, as shown in the following table:

Table 2. 1/10Gb LAN Switch Module IPv4 addresses, by switch-module bay numbers

Bay Number	**Factory-Default IPv4 Address Assigned** **by Management Module**
Bay 1	0.0.0.0
Bay 2	0.0.0.0

Note: 1/10Gb LAN Switch Modules installed in Bay 1 and Bay 2 connect to server blades NICs, respectively.

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 disabled. Use the following commands (available on the console only) to enable or disable Telnet access:

```
Router(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 1-11.

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 1/10Gb LAN Switch Module 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 listed below.

- 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, RADIUS, TACACS+ The following SSH clients have been tested:

- OpenSSH_5.1p1 Debian-3ubuntu1

- SecureCRT 5.0 (Van Dyke Technologies, Inc.)

- Putty beta 0.60

Note: The Networking OS implementation of SSH supports version 2.0 and supports SSH client version 2.0.

Using SSH to Access the Switch

By default, the SSH feature is enabled. For information about enabling and using SSH for switch access, see "Secure Shell and Secure Copy" on page 2-3.

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 1-11.

Using a Web Browser

The switch provides a Browser-Based Interface (BBI) for accessing the common configuration, management and operation features of 1/10Gb LAN Switch Module through your Web browser.

You can 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*>).

When you first access the switch, you must enter the default username and password: USERID; PASSW0RD (with a zero). You are required to change the password after first login.

Configuring HTTP Access to the BBI

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

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

```
Router(config)# access http enable    (Enable HTTP access)

      -or-

Router(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:

```
Router(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 enabled on the switch. To disable or re-enable BBI access via HTTPS, use the following command:

```
Router(config)# no access https enable      (Disable HTTPS access)

      -or-

Router(config)# access https enable   (Enable HTTPS access)
```

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

The default HTTPS web server port to access the BBI is port 443. However, you can change the default Web server port with the following command:

```
Router(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.

```
Router(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 that it is retained beyond reboot or power cycles, use the following command:

```
Router(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 *Networking OS BBI Quick Guide.*

BBI 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 for virtual machine (VM) support.

For information on using the BBI, refer to the *Networking OS BBI Quick Guide*.

Using Simple Network Management Protocol

Networking OS provides Simple Network Management Protocol (SNMP) version 1, version 2, and version 3 support for access through any network management software.

To access the SNMP agent on 1/10Gb LAN Switch Module, the read and write community strings on the SNMP manager should be configured to match those on the switch.

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

```
Router(config)# snmp-server read-community <1-32 characters>

-and-

Router(config)# snmp-server write-community <1-32 characters>
```

The SNMP manager should 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:

```
Router(config)# snmp-server trap-source <trap source IP interface>

Router(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:

```
>> # snmp trap-source loopback <1-5>
```

For more information on SNMP usage and configuration, see "Simple Network Management Protocol" on page 7-12.

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 obtained automatically via BOOTP or DHCP relay as discussed below.

1/10Gb LAN Switch Module 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 four BOOTP/DHCP servers. In addition to the four global BOOTP/DHCP servers, up to four 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 1/10Gb LAN Switch Module 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 1/10Gb LAN Switch Module.

Switch Login Levels

To enable better switch management and user accountability, three levels or *classes* of user access have been implemented on 1/10Gb LAN Switch Module. 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 1/10Gb LAN Switch Module. 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 1/10Gb LAN Switch Module. 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 1/10Gb LAN Switch Module. 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. For more information, see “Changing the Switch Passwords” on page 2-2.

Table 3. 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 1/10Gb LAN Switch Module, 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:

```
Router(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.

Secure FTP

Networking OS supports Secure FTP (SFTP) to the switch. SFTP uses Secure Shell (SSH) to transfer files. SFTP encrypts both commands and data, and prevents passwords and sensitive information from being transmitted openly over the network.

All file transfer commands include SFTP support along with FTP and TFTP support. SFTP is available through the menu-based CLI, CLI, BBI, and SNMP.

The following examples illustrate SFTP support for CLI commands:

```
Router# copy sftp {image1|image2|boot-image} [mgt-port|data-port]

(Copy software image from SFTP server to the switch)

Router# copy sftp {ca-cert|host-cert|host-key} [mgt-port|data-port]

(Copy HTTPS certificate or host key from SFTP server to the switch)
```

Boot Strict Mode

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

1/10Gb LAN Switch Module 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.

By default, HTTP, Telnet, and SNMPv1 and SNMPv2 are disabled on 1/10Gb LAN Switch Module. Before enabling strict mode, ensure the following:

- The software version on all connected switches is Networking OS 7.8.

- 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

(Router(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 4. 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
HTTPS	TLS 1.2 only See "Acceptable Cipher Suites" on page 1-16;	TLS 1.0, 1.1, 1.2 See "Acceptable Cipher Suites" on page 1-16;
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. How- ever, 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 dis- abled. 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	SNMPv1, SNMPv2, SNMPv3 DES/MD5, AES-128-CFB-128/SHA1

Table 4. 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-HELLMAN-GROUP-EXCHANGE-SHA256 DIFFIE-HELLMAN-GROUP-EXCHANGE-SHA1	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 3DES-CBC	AES128-CTR AES128-CBC RIJNDAEL128-CBC BLOWFISH-CBC 3DES-CBC ARCFOUR256 ARCFOUR128 ARCFOUR
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 1/10Gb LAN Switch Module is in compatibility mode:

Table 5. 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	RC4	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	RC4	SHA1	SSL_RSA_WITH_RC4_128_SHA
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 1/10Gb LAN Switch Module is in strict mode:

Table 6. 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:

```
Router(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 7-13.

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.

Limitations

In Networking OS 7.8, 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.

- 1/10Gb LAN Switch Module 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.

Initial Setup

To help with the initial process of configuring your switch, the Networking OS 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: `Router(config)# setup`

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 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 will appear as shown here.

```
Enter login username:

Enter login password:
```

2. Enter **USERID** as the default administrator and **PASSW0RD** (with a zero) as the default password.

3. Enter the following command at the prompt:

```
Router(config)# setup
```

Stopping and Restarting Setup Manually

Stopping Setups

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 administrator prompt:

```
Router(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 *Networking OS 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 [2012]:
```

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, 2012.
```

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, 2012.
```

8. Turn BOOTP on or off at the prompt:

```
BootP Option:

Current BOOTP: disabled

Enter new BOOTP [d/e]:
```

9. 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 tagging for ports:

```
Port Config:

Will you configure VLANs and VLAN tagging 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 1-23.

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 tagging for the port.

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

```
Port VLAN tagging config (tagged port can be a member of multiple
VLANs)

Current VLAN tag support: disabled

Enter new VLAN tag support [d/e]:
```

Enter d to disable VLAN 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 (INTA1-B14, EXT1-24):
```

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 1-24.

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 1-24.

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, 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 STG index [1-128](802.1d)/[1](RSTP)/[0-32](MSTP):
```

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 5-12.

IP Interfaces

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

Up to 128 IP interfaces can be configured on 1/10Gb LAN Switch Module. 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: (1-128)
```

2. 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 1-26. 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>.

4. 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.

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

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

6. The system prompts you to configure another interface:

```
Enter interface number: (1-128)
```

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.

Default Gateways

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

```
IP default gateways:

Enter default gateway number: (1-3, 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 5-1.

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 1/10Gb LAN Switch Module, 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 as shown in "Changing the Switch Passwords" on page 2-2.

Optional Setup for Telnet Support

Note: This step is optional. Perform this procedure only if you are planning on connecting to 1/10Gb LAN Switch Module through a remote Telnet connection.

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

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

Service Location Protocol

Service Location Protocol (SLP) allows the switch to provide dynamic directory services that helps 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.

Note: SLP is not supported on the internal management port (MGT).

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 CLI commands to configure SLP for the switch:

Table 7. SLP CLI 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
`ip slp active-da-discovery start-wait-time` *<1-10>* Configures the wait time before starting Active DA Discovery, in seconds. The default value is 3 seconds. **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 counters`

Displays SLP statistics. **Command mode:** All
`clear ip slp counters` Clears all Directory Agents learned by the switch. **Command mode:** Global configuration

System License Keys

License keys determine the number of available ports on 1/10Gb LAN Switch Module. Each switch comes with basic license that provides the use of a limited number of physical ports. On top of the basic license, optional upgrade licenses can be installed to expand the number of available ports.

The upgrade licenses can be purchased from Hitachi. Please contact your reseller.

Flexible Port Mapping

Flexible Port Mapping allows administrators to manually enable or disable specific switch ports within the limitations of the installed licenses' bandwidth.

For instance, the Hitachi BladeSymphony may include two compute nodes and a single SFP+ uplink, while the current license has the INTA1 – INTA14 and EXT1 – EXT10 Ethernet ports enabled by default.

To make best use of the available resources, the administrator decides to activate internal ports INTB1 and INTB2 to provide redundant connections for the two compute nodes and to enable the high speed SFP+ EXT21 port for the uplink.

The total bandwidth required for this operation amounts to 12 Gbps (2 Gbps for the two additional 1 Gbps internal ports and 10 Gbps for the additional external SFP+ port). The administrator decides to allocate this bandwidth by deactivating 6 internal and 6 external 1 Gbps ports.

To implement the above scenario, follow these steps:

1. Deactivate the ports required to clear the 12 Gbps required bandwidth:

```
Router(config)# no boot port-map INTA9
Router(config)# no boot port-map INTA10
Router(config)# no boot port-map INTA11
Router(config)# no boot port-map INTA12
Router(config)# no boot port-map INTA13
Router(config)# no boot port-map INTA14
Router(config)# no boot port-map EXT5
Router(config)# no boot port-map EXT6
Router(config)# no boot port-map EXT7
Router(config)# no boot port-map EXT8
Router(config)# no boot port-map EXT9
Router(config)# no boot port-map EXT10
```

2. Activate the required ports:

```
Router(config)# boot port-map INTB1
Router(config)# boot port-map INTB2
Router(config)# boot port-map EXT21
```

3. A reboot is required for the changes to take effect.

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

Flexible Port Mapping is disabled if all available licenses are installed (all physical ports are available).

Removing a license key reverts the port mapping to the default settings for the remaining licensing level. To manually revert the port mapping to the default settings use the following command:

```
Router(config)# default boot port-map
```

2

Securing the Switch

This chapter discusses different methods of securing local and remote administration on 1/10Gb LAN Switch Module.

- □ Securing Administration
- □ Authentication & Authorization Protocols
- □ 802.1X Port-Based Network Access Control
- □ Access Control Lists

Securing Administration

This chapter discusses different methods of securing local and remote administration on 1/10Gb LAN Switch Module:

- "Changing the Switch Passwords" on page 2-2
- "Secure Shell and Secure Copy" on page 2-3
- "End User Access Control" on page 2-8
- "Protected Mode" on page 2-11

Changing the Switch Passwords

It is recommended that you change the administrator and user passwords after initial configuration and as regularly as required under your network security policies.

To change the administrator password, you must login using the administrator password.

Note: If you forget your administrator password, call your technical support representative for help using the password fix-up mode.

You can also change the default user names, if desired. The user name length can be up to 64 characters.

Changing the Default Administrator Password

The administrator has complete access to all menus, information, and configuration commands, including the ability to change both the user and administrator passwords.

The default administrator account is USERID. The default password for the administrator account is PASSW0RD (with a zero). To change the administrator password, use the following procedure:

1. Connect to the switch and log in as the administrator.

2. Use the following command to change the administrator password:

```
Router(config)# access user administrator-password <password>
```

Secure Shell and Secure Copy

Because using Telnet does not provide a secure connection for managing 1/10Gb LAN Switch Module, Secure Shell (SSH) and Secure Copy (SCP) features have been included for 1/10Gb LAN Switch Module 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 1/10Gb LAN Switch Module 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 1/10Gb LAN Switch Module, 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

The Networking OS implementation of SSH supports both versions 1.5 and 2.0 and supports SSH clients version 1.5 - 2.x. The following SSH clients have been tested:

- SSH 1.2.23 and SSH 1.2.27 for Linux (freeware)
- SecureCRT 3.0.2 and SecureCRT 3.0.3 for Windows® NT (Van Dyke Technologies, Inc.)
- F-Secure SSH 1.1 for Windows® (Data Fellows)
- Putty SSH
- Cygwin OpenSSH
- Mac X OpenSSH
- Solaris 8 OpenSSH
- AxeSSH SSHPro
- SSH Communications Vandyke SSH A
- F-Secure

Configuring SSH/SCP Features on the Switch

SSH and SCP are disabled by default. To change the setting, using the following procedures.

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

To Enable or Disable the SSH Feature

Begin a Telnet session from the console port and enter the following commands:

```
Router(config)# ssh enable       (Turn SSH on)

Router(config)# no ssh enable    (Turn SSH off)
```

To Enable or Disable SCP

Enter the following command to enable or disable SCP:

```
Router(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):

```
Router(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 common client commands.

To Log In to the Switch from the Client

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, in order 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,DES
● User Authentication:	Local password authentication, RADIUS

Generating RSA Host Key for SSH Access

To support the SSH server feature, an RSA host key is required. The host key is 2048 bits and is used to identify 1/10Gb LAN Switch Module.

When the SSH server is first enabled and applied, the switch automatically generates the RSA host key and stores it in FLASH memory.

To configure RSA host key, first connect to 1/10Gb LAN Switch Module through the console port (commands are not available via external Telnet connection), and enter the following command to generate it manually.

```
Router(config)# ssh generate-host-key (Generates the 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

- A maximum of 20 user IDs are supported on the switch.
- Networking OS supports end user support for Console, Telnet, BBI, and SSHv1/v2 access to the switch.
- If RADIUS authentication is used, the user password on the Radius server will override the user password on 1/10Gb LAN Switch Module. Also note that the password change command modifies only the user switch 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 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 1/10Gb LAN Switch Module. 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:

```
Router(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 (last time) to change the password. A warning provides advance notice for users to change the password.

User Access Control Menu

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 in the User ID menu.

```
Router(config)# access user 1 name <1-8 characters>
Router(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 8 on page 2-15.

To change the user’s level, enter the class of service `cos` command:

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

Validating a User’s Configuration

```
Router# 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.

```
Router(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 2-8). Then use the following command:

```
Router(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:

```
Router(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:

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

Listing Current Users

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

```
Router# show access user


Usernames:

user     - Enabled – offline

oper     - Disabled - offline

admin    - Always Enabled - online 1 session


Current User ID table:

1:name USERID     , ena, cos admin    , password valid, offline

2:name jane       , ena, cos user     , password valid, online

3:name john       , ena, cos user     , password valid, online
```

Logging In to 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 Class of Service established for the end user account.

Protected Mode

Protected Mode settings allow the switch administrator to block the Management Module from making configuration changes that affect switch operation. The switch retains control over those functions.

The following Management Module functions are disabled when Protected Mode is turned on:

- External Ports: Enabled/Disabled
- External management over all ports: Enabled/Disabled
- Restore Factory Defaults
- New Static IP Configuration

In this release, configuration of the functions listed above are restricted to the local switch when you turn Protected Mode on. In future releases, individual control over each function may be added.

Note: Before you turn Protected Mode on, make sure that external management (Telnet) access to one of the switch's IP interfaces is enabled.

Use the following command to turn Protected Mode on:

```
Router(config)# protected-mode enable
```

If you lose access to the switch through the external ports, use the console port to connect directly to the switch, and configure an IP interface with Telnet access.

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 2-12
- "TACACS+ Authentication" on page 2-16
- "LDAP Authentication and Authorization" on page 2-21

Note: Networking OS 7.8 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

1/10Gb LAN Switch Module—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

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 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 1/10Gb LAN Switch Module.

1. Turn RADIUS authentication on, then configure the Primary and Secondary RADIUS servers.

```
Router(config)# radius-server primary-host 10.10.1.1

Router(config)# radius-server secondary-host 10.10.1.2
```

2. Configure the RADIUS secret.

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

Router(config)# radius-server secondary-host 10.10.1.2 key <1-32
character secret>

Router(config)# radius-server enable
```

Statement 21:

CAUTION

If you configure the RADIUS secret using any method other than through

the console port, the secret may be transmitted over the network as clear text.

3. If desired, you may change the default UDP port number used to listen to RADIUS.

The well-known port for RADIUS is 1645.

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

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

```
Router(config)# radius-server retransmit 3

Router(config)# radius-server timeout 5
```

RADIUS Authentication Features in Networking OS

Networking 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 a RADIUS secret password of up to 32 characters.

- 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:

```
Router# 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 within 1-10 seconds. The switch automatically retries connecting to the RADIUS server 1-3 times before it declares the server down.
- Supports user-configurable RADIUS application port. The default is UDP port 1645.UDP port 1812, based on RFC 2138, 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 8 can be defined in the RADIUS server dictionary file.

Table 8. User Access Levels

User Account	Description and Tasks Performed	Password
User	The User has no direct responsibility for switch management. He/she can view all switch status information and statistics but cannot make any configuration changes to the switch.	`user`
Operator	In addition to User capabilities, the Operator has limited switch management access, including the ability to make temporary, operational configuration changes to some switch features, and to reset switch ports (other than management ports).	`oper`
Administrator (`USERID`)	The super-user Administrator has complete access to all menus, information, and configuration commands on the switch, including the ability to change both the user and administrator passwords.	`PASSW0RD`

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 1/10Gb LAN Switch Module 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 Networking OS user privileges levels:

Table 9. Networking OS-proprietary Attributes for RADIUS

User Name/Access	User-Service-Type	Value
User	*Vendor-supplied*	255
Operator	*Vendor-supplied*	252
Administrator (`USERID`)	*Vendor-supplied*	6

TACACS+ Authentication

Networking OS supports authentication, authorization, and accounting with networks using the Cisco Systems TACACS+ protocol. 1/10Gb LAN Switch Module 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 1/10Gb LAN Switch Module either through a data or management 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 2-17.

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. Networking 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 Networking OS management access levels is shown in Table 10. The authorization levels listed in this table must be defined on the TACACS+ server.

Table 10. Default TACACS+ Authorization Levels

Networking OS User Access Level	**TACACS+ Level**
`user`	0
`oper`	3
`admin (USERID)`	6

Alternate mapping between TACACS+ authorization levels and Networking OS management access levels is shown in Table 11. Use the following command to use the alternate TACACS+ authorization levels:

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

Table 11. Alternate TACACS+ Authorization Levels

Networking OS User Access Level	**TACACS+ Level**
`user`	0-1
`oper`	6-8
`admin (USERID)`	14-15

You can customize the mapping between TACACS+ privilege levels and 1/10Gb LAN Switch Module management access levels. Use the following command to manually map each TACACS+ privilege level (0-15) to a corresponding 1/10Gb LAN Switch Module management access level: `Router(config)# tacacs-server user-mapping`

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 *backdoor* access via Telnet (`Router(config)# tacacs-server backdoor`). The default value for Telnet access is disabled. The administrator also can enable *secure backdoor (*`Router(config)# tacacs-server secure-backdoor`*)*, to allow access if both the primary and the secondary TACACS+ servers fail to respond.

Note: To obtain the TACACS+ backdoor password for your switch, contact your Service and Support line.

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.

1/10Gb LAN Switch Module supports the following TACACS+ accounting attributes:

- protocol (console/telnet/ssh/http)
- 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 **Quit** button on the browser is clicked.

Command Authorization and Logging

When TACACS+ Command Authorization is enabled

(`Router(config)# tacacs-server command-authorization`), Networking OS configuration commands are sent to the TACACS+ server for authorization. When TACACS+ Command Logging is enabled

(`Router(config)# tacacs-server command-logging`), Networking OS configuration commands are logged on the TACACS+ server.

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

```
authorization request, cmd=cfgtree, cmd-arg=/cfg/l3/if
accounting request, cmd=/cfg/l3/if, cmd-arg=1
authorization request, cmd=cfgtree, cmd-arg=/cfg/l3/if/ena
accounting request, cmd=/cfg/l3/if/ena
authorization request, cmd=cfgtree, cmd-arg=/cfg/l3/if/addr
accounting request, cmd=/cfg/l3/if/addr, cmd-arg=10.90.90.91
authorization request, cmd=apply
accounting request, cmd=apply
```

The following rules apply to TACACS+ command authorization and logging:

- Only commands from a Console, Telnet, or SSH connection are sent for autho- rization and logging. SNMP, BBI, or file-copy commands (for example, TFTP or sync) are not sent.

- Only leaf-level commands are sent for authorization and logging. For example,

```
Router(config)# is not sent, but

Router(config)# tacacs-server command-logging is sent.
```

- The full path of each command is sent for authorization and logging. For example:
`Router(config)# tacacs-server command-logging`

- Command arguments are not sent for authorization.

- Only executed commands are logged.

- Invalid commands are checked by Networking OS, and are not sent for authorization or logging.

- Authorization is performed on each leaf-level command separately. If the user issues multiple commands at once, each command is sent separately as a full path.

- Only the following global commands are sent for authorization and logging:

```
diff
ping
revert
telnet
traceroute
```

TACACS+ Password Change

Networking OS supports TACACS+ password change. When enabled, users can change their passwords after successful TACACS+ authorization. Use the following command to enable or disable this feature:

```
Router(config)# [no] tacacs-server password-change
```

Use the following commands to change the password for the primary and secondary TACACS+ servers:

```
Router(config)# tacacs-server chpassp      (Change primary TACACS+ password)

Router(config)# tacacs-server chpasss      (Change secondary TACACS+ password)
```

Configuring TACACS+ Authentication on the Switch

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

```
Router(config)# tacacs-server primary-host 10.10.1.1
                                       (Enter primary server IPv4 address)
Router(config)# tacacs-server primary-host mgt-port
Router(config)# tacacs-server secondary-host 10.10.1.1
                                       (Enter secondary server IPv4 address)
Router(config)# tacacs-server secondary-host data-port
Router(config)# tacacs-server enable
```

2. Configure the TACACS+ secret and second secret.

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

Statement 21:

CAUTION
If you configure the TACACS+ secret using any method other than a direct console connection, the secret may be transmitted over the network as clear text.

3. If desired, you may change the default TCP port number used to listen to TACACS+. The well-known port for TACACS+ is 49.

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

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

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

5. Configure custom privilege-level mapping (optional).

```
Router(config)# tacacs-server user-mapping 2 user

Router(config)# tacacs-server user-mapping 3 user

Router(config)# tacacs-server user-mapping 4 user

Router(config)# tacacs-server user-mapping 5 oper
```

LDAP Authentication and Authorization

Networking 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

1/10Gb LAN Switch Module user groups and user accounts must reside within the same domain. On the LDAP server, configure the domain to include 1/10Gb LAN Switch Module 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 1/10Gb LAN Switch Module, as follows:

  - admin (USERID)
  - oper
  - user

Configuring LDAP Authentication on the Switch

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

```
Router(config)# ldap-server primary-host 10.10.1.1 (Enter primary server IPv4
address)

Router(config)# ldap-server primary-host 10.10.1.2 (Enter secondary server
IPv4 address)
```

2. Configure the domain name.

```
Router(config)# ldap-server domain <ou=people,dc=my-domain,dc=com>
```

3. If desired, you may change the default TCP port number used to listen to LDAP. The well-known port for LDAP is 389.

```
Router(config)# ldap-server port <1-65000>
```

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

```
Router(config)# ldap-server retransmit 3   (server retries)
Router(config)# ldap-server timeout 10     (Enter the timeout period in
seconds)
```

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 1/10Gb LAN Switch Module that connect to blade servers.

The following topics are discussed in this section:

- "Extensible Authentication Protocol over LAN" on page 2-22
- "EAPoL Authentication Process" on page 2-24
- "EAPoL Port States" on page 2-25
- "Guest VLAN" on page 2-26
- "Supported RADIUS Attributes" on page 2-26
- "EAPoL Configuration Guidelines" on page 2-28

Extensible Authentication Protocol over LAN

Networking OS 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 Authenticator 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. 1/10Gb LAN Switch Module acts as an Authenticator.

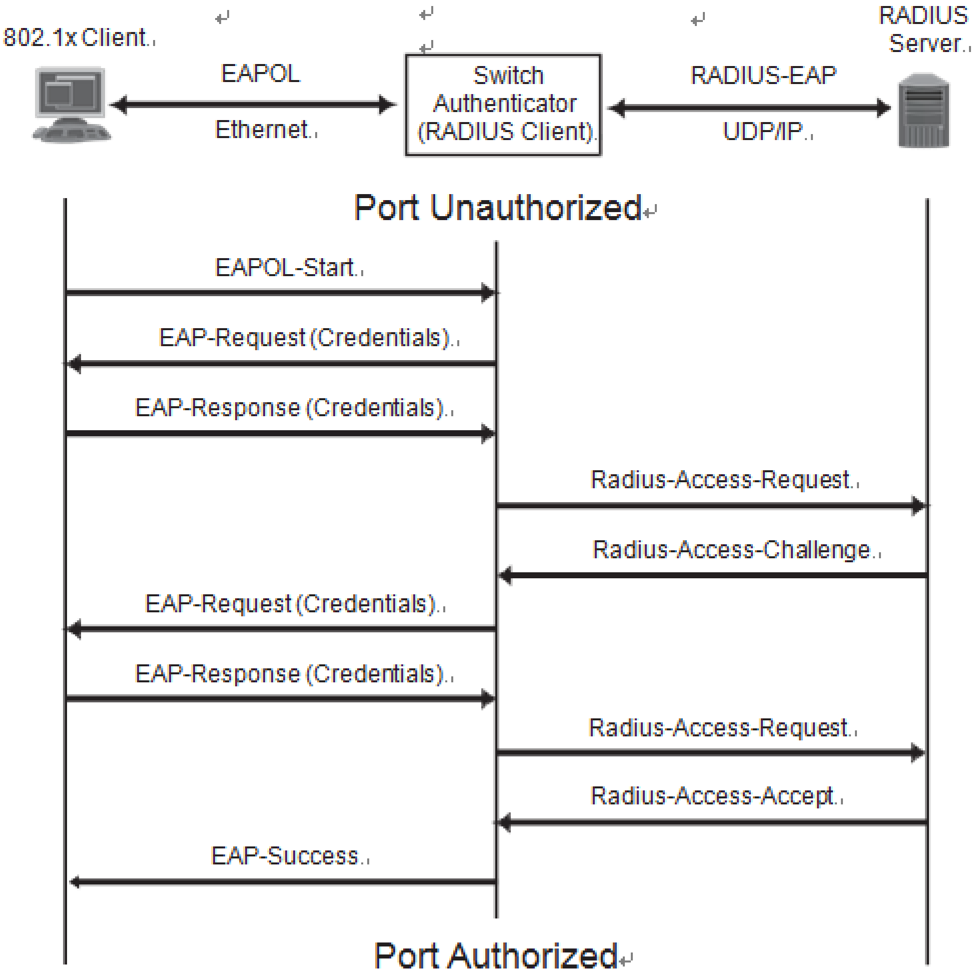
- Authentication Server
  The Authentication Server validates the credentials provided by the Supplicant to determine if the Authenticator should grant access to the network. The Authentication Server may be co-located with the Authenticator. 1/10Gb LAN Switch Module relies on external RADIUS servers 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 1/10Gb LAN Switch Module authenticator, while RADIUS-EAP messages are exchanged between 1/10Gb LAN Switch Moduleauthenticator and the RADIUS server.

Authentication is initiated by one of the following methods:

- 1/10Gb LAN Switch Module authenticator sends an EAP-Request/Identity packet to the client
- The client sends an EAPOL-Start frame to 1/10Gb LAN Switch Module authenticator, which responds with an EAP-Request/Identity frame.

The client confirms its identity by sending an EAP-Response/Identity frame to 1/10Gb LAN Switch Module 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 1/10Gb LAN Switch Module 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 1/10Gb LAN Switch Module 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, 1/10Gb LAN Switch Module 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.

Guest VLAN

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

```
Router(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 12 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 12. 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 (i.e. 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	Called-Station-ID	The MAC address of the supplicant encoded as an ASCII string in canonical format, such as 00034B436206.	1	0	0	0

Table 12. Support for RADIUS Attributes(continued)

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
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 1/10Gb LAN Switch Module is connected to another 1/10Gb LAN Switch Module, 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.

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 in order 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.

Networking OS 7.8 supports the following ACLs:

- IPv4 ACLs

Up to 640 ACLs are supported for networks that use IPv4 addressing. IPv4 ACLs are configured using the following CLI menu:

```
Router(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 CLI menu:

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

- Management ACLs

Up to 128 MACLs are supported. ACLs for the different types of management protocols (Telnet, HTTPS, etc.) provide greater granularity for securing management traffic.

Management ACLs are configured using the following CLI menu:

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

- VLAN Maps (VMaps)

Up to 128 VLAN Maps are supported for attaching filters to VLANs rather than ports. See “VLAN Maps” on page 2-35 for details.

```
Router(config)# access-control vmap <vmap 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.

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

- Ethernet header options (for regular ACLs and VMaps only)
    - Source MAC address
    - Destination MAC address
    - VLAN number and mask
    - Ethernet type (ARP, IPv4, MPLS, RARP, etc.)
    - Ethernet Priority (the IEEE 802.1p Priority)
- IPv4 header options (for regular 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 13:

Table 13. Well-Known Protocol Types

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

- TCP/UDP header options (for all ACLs)
  - TCP/UDP application source port as shown in Table 14.

Table 14. 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	Accouting
70	gopher	162	snmptrap		hsrp

  - TCP/UDP application destination port and mask as shown in Table 14.
  - TCP/UDP flag value as shown in Table 15.

Table 15. Well-Known TCP flag values

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

- Packet format (for regular ACLs and VMaps only)
  - Ethernet format (eth2, SNAP, LLC)
  - Ethernet tagging format
  - IP format (IPv4)
- Egress port packets (for all ACLs)

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. 1/10Gb LAN Switch Module 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, or in groups.

To assign an individual ACL to a port, use the following IP interface commands:

```
Router(config)# interface port <port>
Router(config-if)# access-control list <IPv4 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.

To create and assign ACLs in groups, see "ACL Groups" on page 2-31.

ACL Order of Precedence

When multiple ACLs are assigned to a port, they are evaluated in numeric sequence, based on the ACL number. Lower-numbered ACLs take precedence over higher-numbered ACLs. For example, ACL 1 (if assigned to the port) is evaluated first and has top priority.

If multiple ACLs match the port traffic, only the action of the one with the lowest ACL number is applied. The others are ignored.

The ACL number is the sole factor in determining ACL order of precedence. The order in which ACLs are applied to a port does not affect the order of precedence, nor does the ACL Group number (see "ACL Groups" on page 2-31), the order in which an ACL is assigned to an ACL Group, or the order in which the ACL Group is assigned to a port.

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

ACL Groups

To assist in organizing multiple ACLs and assigning them to ports, you can place ACLs into ACL Groups, thereby defining complex traffic profiles. ACLs and ACL Groups can then be assigned on a per-port basis. Any specific ACL can be assigned to multiple ACL Groups, and any ACL or ACL Group can be assigned to multiple ports. If, as part of multiple ACL Groups, a specific ACL is assigned to a port multiple times, only one instance is used. The redundant entries are ignored.

- **Individual ACLs**

1/10Gb LAN Switch Module supports up to 256 ACLs. Each ACL defines one filter rule for matching traffic criteria. Each filter rule can also include an action (permit or deny the packet). For example:

ACL 1:
VLAN = 1
SIP = 10.10.10.1 (255.255.255.0)
Action = permit

- **Access Contorol List Groups**

An Access Control List Group (ACL Group) is a collection of ACLs. For example:

ACL Group 1

ACL 1:
VLAN = 1
SIP = 10.10.10.1 (255.255.255.0)
Action = permit

ACL 2:
VLAN = 1
SIP = 10.10.10.2 (255.255.255.0)
Action = deny

ACL 3:
VLAN = 1
SIP = 10.10.10.3 (255.255.255.0)
Action = permit

ACL Groups organize ACLs into traffic profiles that can be more easily assigned to ports. 1/10Gb LAN Switch Module supports up to 256 ACL Groups.

Note: ACL Groups are used for convenience in assigning multiple ACLs to ports. ACL Groups have no effect on the order in which ACLs are applied (see "ACL Order of Precedence" on page 2-31). All ACLs assigned to the port (whether individually assigned or part of an ACL Group) are considered as individual ACLs for the purposes of determining their order of precedence.

Assigning ACL Groups to a Port

To assign an ACL Group to a port, use the following commands:

```
Router(config)# interface port <port number>
Router(config-if)# access-control group <ACL group number>
Router(config-if)# exit
```

ACL Metering and Re-Marking

You can define a profile for the aggregate traffic flowing through the switch 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 2-31).

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 (1000 bits per second in each 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-Making

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 should receive.
- Change the 802.1p priority of a packet.

ACL Port Mirroring

For regular ACLs and VMaps, packets that match an ACL on a specific port 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.

If the ACL or VMap has an action (permit, drop, etc.) assigned, it cannot be used to mirror packets for that ACL.

Use the following commands to add mirroring to an ACL:

- For regular ACLs:

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

  The ACL must be also assigned to it target ports as usual (see "Assigning Individual ACLs to a Port" on page 2-31, or "Assigning ACL Groups to a Port" on page 2-33).

- For VMaps (see "VLAN Maps" on page 2-35):

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

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:

```
Router(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 EXT1 is denied if it is destined for the host at IP address 100.10.1.1

1. Configure an Access Control List.

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

2. Add ACL 1 to port EXT1.

```
Router(config)# interface port EXT1
Router(config-if)# access-control list 1
Router(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 EXT2 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.

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

2. Add ACL 2 to port EXT2.

```
Router(config)# interface port EXT2
Router(config-if)# access-control list 2
Router(config-if)# exit
```

ACL Example 3

This configuration blocks traffic from a network that is destined for a specific egress port. All traffic that ingresses port EXT1 from the network 100.10.1.0/24 and is destined for port 3 is denied.

1. Configure an Access Control List.

```
Router(config)# access-control list 4 ipv4 source-ip-address 100.10.1.0
255.255.255.0
Router(config)# access-control list 4 egress-port 3
Router(config)# access-control list 4 action deny
```

2. Add ACL 4 to port EXT1

```
Router(config)# interface port EXT1
Router(config-if)# access-control list 4
Router(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 regular 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.

VMAPs are configured using the following CLI command path:

```
Router(config)# access-control vmap <VMAP ID (1-128)>

  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
```

1/10Gb LAN Switch Module 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 the VMAP are assigned to a specific VLAN or associated with a VM group VLAN).

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

- For a regular VLAN:

```
Router(config)# vlan <VLAN ID>
Router(config-vlan)# [no] vmap <VMap ID> [intports|extports]
```

- For a VM group (see “VM Group Types” on page 4-4):

```
Router(config)# [no] virt vmgroup <ID> vmap <VMap ID> [intports|extports]
```

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 intportsor extportsparameter is specified, the action to add or remove the vMAP is applies for either the internal downlink ports or external uplink ports only. 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.

VMap Example

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

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

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. Up to 128 MACLs can be configured. Following is an example MACL configuration based on a destination IP address and a TCP-UDP destination port:

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

Use the following command to view the MACL configuration:

```
Router(config)# show access-control macl 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
```

3

Switch Basics

This section discusses basic switching functions.

- □ VLANs
- □ Ports and Trunking
- □ Spanning Tree Protocols
- □ Quality of Service

VLANs

This chapter describes network design and topology considerations for using Virtual Local Area Networks (VLANs). VLANs are commonly 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 3-3
- "VLAN Tagging/Trunk Mode" on page 3-5
- "VLAN Topologies and Design Considerations" on page 3-8
- "Protocol-Based VLANs" on page 3-10
- "Private VLANs" on page 3-12

Note: Basic VLANs can be configured during initial switch configuration (see "Using the Setup Utility" in the *Networking OS 7.8 Command Reference*). More comprehensive VLAN configuration can be done from the Command Line Interface (see "VLAN Configuration" as well as "Port Configuration" in the *Networking OS 7.8 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.

1/10Gb LAN Switch Module automatically supports jumbo frames. This default cannot be manually configured or disabled.

1/10Gb LAN Switch Module supports jumbo frames with a Maximum Transmission Unit (MTU) of 9,216 bytes. Within each frame, 18 bytes are reserved for the Ethernet header and CRC trailer. The remaining space in the frame (up to 9,198 bytes) comprise the packet, which includes the payload of up to 9,000 bytes 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.

Note: Jumbo frames are not supported for traffic sent to switch management interfaces.

VLANs and Port VLAN ID Numbers

VLAN Numbers

Networking OS supports up to 1024 VLANs per switch. Even though the maximum number of VLANs supported at any given time is 1024, each can be identified with any number between 1 and 4094. VLAN 1 is the default VLAN for the external ports and the internal blade ports.

VLAN 4095 is reserved for use by the management network, which includes the management ports. This configuration allows Serial over LAN (SoL) management— a feature available on certain server blades. Management functions can also be assigned to other VLANs (using the following command:

```
Router(config)# vlan <x>
Router(config-vlan)# management
```

Use the following command to view VLAN information:

```
Router# show vlan

 VLAN   Name                              Status MGT     Ports
 ----   --------------------------------  ------ --- -------------------------
 1      Default VLAN                      ena     dis INTA1-EXT24
 4095   Mgmt VLAN                         ena     ena MGT1
Primary Secondary Type Ports
------- --------- ---------------  -------------------------------------------
```

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 type of blade chassis unit that you are using and the firmware versions and options that are installed.

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 CLI commands to view PVIDs:

- Port information:

```
Router# show interface information
(or)
Router# show interface trunk

Alias   Port Tag RMON Lrn Fld PVID    DESCRIPTION    VLAN(s)
             Trk              NVLAN
------- ---- --- ---- --- --- ------ ------------ ------------------------------
INTA     1   n    d   e    e   1      INTA1         1
INTA2    2   n    d   e    e   1      INTA2         1
INTA3    3   n    d   e    e   1      INTA3         1
INTA4    4   n    d   e    e   1      INTA4         1
INTA5    5   y    d   e    e   1      INTA5         1
INTA6    6   n    d   e    e   1      INTA6         1
INTA7    7   n    d   e    e   1      INTA7         1
INTA8    8   n    d   e    e   1      INTA8         1
INTA9    9   n    d   e    e   1      INTA9         1
INTA10   10  n    d   e    e   1      INTA10        1
INTA11   11  n    d   e    e   1      INTA11        1
INTA12   12  n    d   e    e   1      INTA12        1
INTA13   13  n    d   e    e   1      INTA13        1
INTA14   14  y    d   e    e   1      INTA14        1
INTB1    15  n    d   e    e   1      INTB1         1
INTB2    16  n    d   e    e   1      INTB2         1
INTB3    17  n    d   e    e   1      INTB3         1
...
INTB13   27  n    d   e    e   1      INTB13        1
INTB14   28  n    d   e    e   1      INTB14        1
EXT1     29  n    d   e    e   1      EXT1          1
EXT2     30  n    d   e    e   1      EXT2          1
EXT3     31  n    d   e    e   1      EXT3          1
EXT4     32  n    d   e    e   1      EXT4          1
EXT5     33  n    d   e    e   1      EXT5          1
EXT6     34  n    d   e    e   1      EXT6          1
...
EXT18    46  n    d   e    e   1      EXT18         1
EXT19    47  n    d   e    e   1      EXT19         1
EXT20    48  n    d   e    e   1      EXT20         1
EXT21    49  n    d   e    e   1      EXT21         1
EXT22    50  n    d   e    e   1      EXT22         1
EXT23    51  n    d   e    e   1      EXT23         1
EXT24    52  y    d   e    e   1      EXT24         1
MGT1     53  y    d   e    e   4095   MGT1          4095
* = PVID/Native-VLAN is tagged.
# = PVID is ingress tagged.
Trk = Trunk mode
NVLAN = Native-VLAN
```

Note: The sample output that appears in this document might differ slightly from that displayed by your system. Output varies based on the type of blade chassis unit that you are using and the firmware versions and options that are installed.

- Port Configuration:

```
Access Mode Port

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

For Trunk Mode Port


Router(config)# interface port <port number>
Router(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* enabled (see "VLAN Tagging/Trunk Mode" on page 3-5).

VLAN Tagging/Trunk Mode

Networking OS 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.

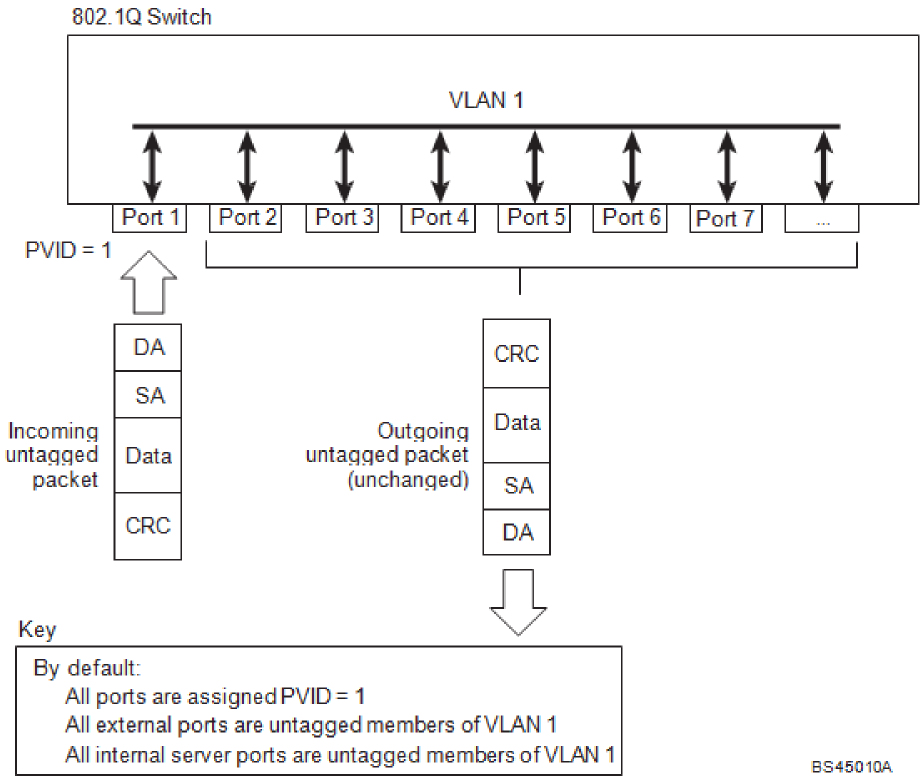
Important terms used with the 802.1Q tagging feature are:

- 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, then the frame is dropped at the ingress port.

Figure 2. Default VLAN settings


802.1Q Switch
VLAN 1
Port 1
Port 2
Port 3
Port 4
Port 5
Port 6
Port 7
...
PVID = 1
DA
SA
Data
CRC
Incoming untagged packet
CRC
Data
SA
DA
Outgoing untagged packet (unchanged)
Key
By default:
All ports are assigned PVID = 1
All external ports are untagged members of VLAN 1
All internal server ports are untagged members of VLAN 1
BS45010A

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

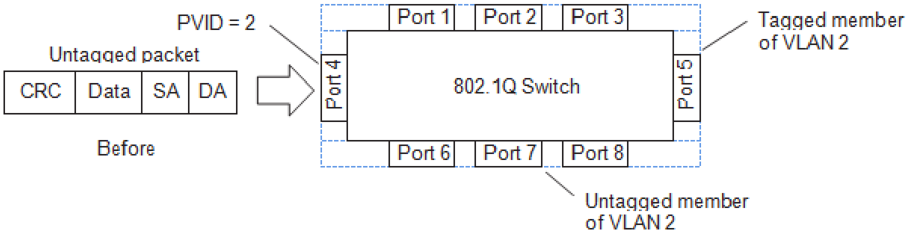
When a VLAN is configured, ports are added as members of the VLAN, and the ports are defined as either *tagged* or *untagged* (see Figure 3 through Figure 6).

The default configuration settings for 1/10Gb LAN Switch Modules have 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.

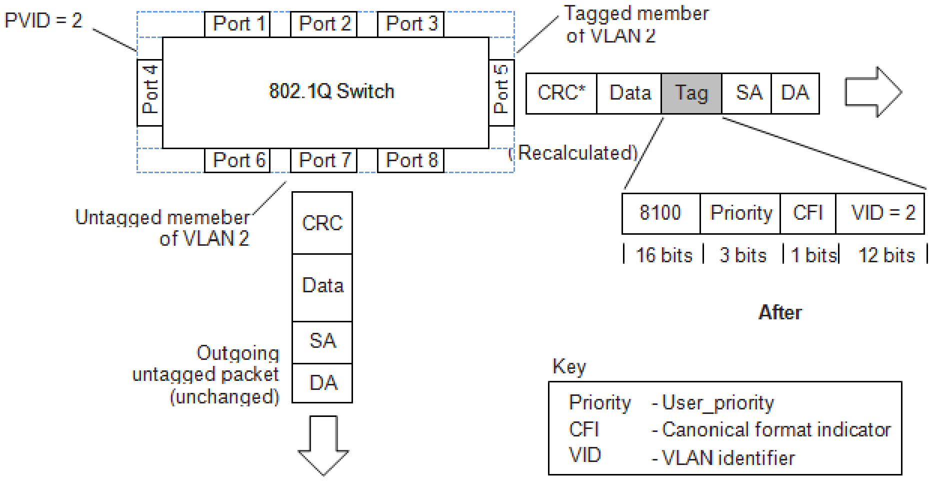
Note: The port assignments in the following figures are general examples and are not meant to match any specific 1/10Gb LAN Switch Module.

Figure 3. Port-based VLAN assignment



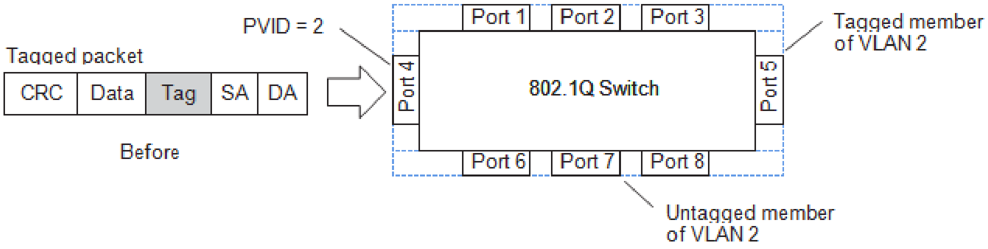
As shown in Figure 4, the untagged packet is marked (tagged) as it leaves the switch through port 5, which is configured as a tagged member of VLAN 2. The untagged packet remains unchanged as it leaves the switch through port 7, which is configured as an untagged member of VLAN 2.

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



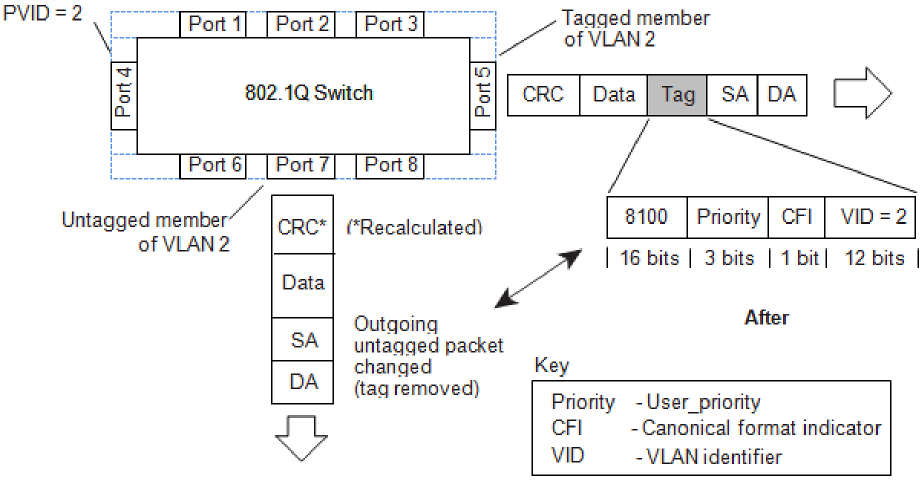
In Figure 5, tagged incoming packets are assigned directly to VLAN 2 because of the tag assignment in the packet. Port 5 is configured as a tagged member of VLAN 2, and port 7 is configured as an untagged member of VLAN 2.

Figure 5. 802.1Q tag assignment



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)



Note: Set the configuration to factory default

(`Router(config)# boot configuration-block factory`) to reset all non-management ports to VLAN 1.

VLAN Topologies and Design Considerations

- By default, the Networking OS software is configured so that tagging is disabled on all external ports and on all internal ports.
- By default, the Networking OS software is configured so that all internal ports are members of VLAN 1.
- By default, the Networking OS software is configured so that the management port is a member of the default management VLAN 4095.
- Multiple management VLANs can be configured on the switch, in addition to the default VLAN 4095, using the following commands:

```
Router(config)# vlan <x>
Router(config-vlan)# management
```

- 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.

VLAN Configuration Rules

VLANs operate according to specific configuration rules. When creating VLANs, consider the following rules that determine how the configured VLAN reacts in any network topology:

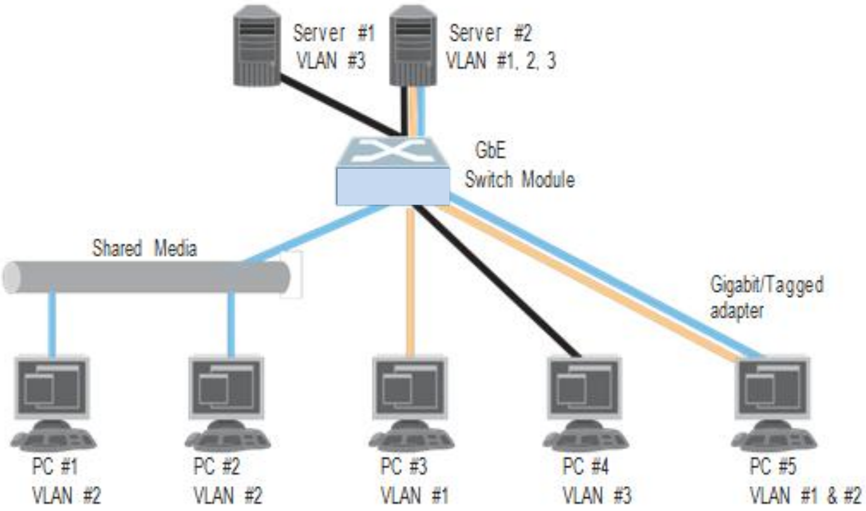
- All ports involved in trunking and port mirroring must have the same VLAN con- figuration. If a port is on a trunk with a mirroring port, the VLAN configuration cannot be changed. For more information trunk groups, see “Configuring a Static Port Trunk” on page 3-18.

- All ports that are involved in port mirroring must have memberships in the same VLANs. If a port is configured for port mirroring, the port's VLAN membership cannot be changed. For more information on configuring port mirroring, see "Port Mirroring" on page 8-10.

- Management VLANs must contain the management port, and can include one or more internal ports (INT*x*). External ports (EXT*x*) cannot be members of any management VLAN.

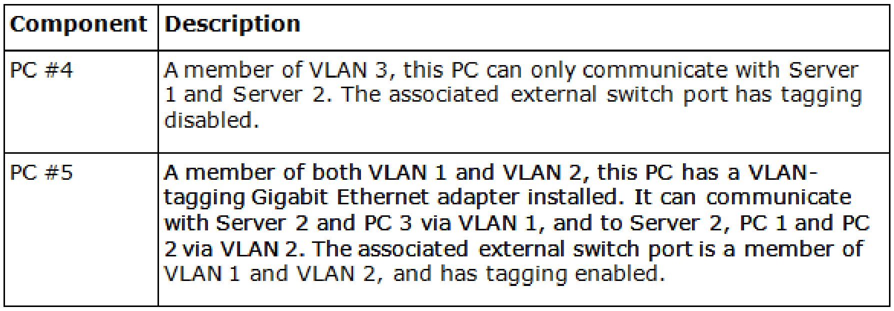
Example: Multiple VLANs with Tagging Adapters

Figure 7. Multiple VLANs with VLAN-Tagged Gigabit Adapters



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

Component	Description
Switch	This switch is configured for three VLANs that represent three different IP subnets. Two servers and five clients are attached to the switch.
Server #1	This server is a member of VLAN 3 and has presence in only one IP subnet. The associated internal switch port is only a member of VLAN 3, so tagging is disabled.
Server #2	This high-use server needs to be accessed from all VLANs and IP subnets. The server has a VLAN-tagging adapter installed with VLAN tagging turned on. The adapter is attached to one of the internal switch ports, that is a member of VLANs 1, 2, and 3, and has tagging enabled. Because of the VLAN tagging capabilities of both the adapter and the switch, the server is able to communicate on all three IP subnets in this network. Broadcast separation between all three VLANs and subnets, however, is maintained.
PCs #1 and #2	These PCs are attached to a shared media hub that is then connected to the switch. They belong to VLAN 2 and are logically in the same IP subnet as Server 2 and PC 5. The associated external switch port has tagging disabled.
PC #3	A member of VLAN 1, this PC can only communicate with Server 2 and PC 5. The associated external switch port has tagging disabled.



Component	Description
PC #4	A member of VLAN 3, this PC can only communicate with Server 1 and Server 2. The associated external switch port has tagging disabled.
PC #5	A member of both VLAN 1 and VLAN 2, this PC has a VLAN-tagging Gigabit Ethernet adapter installed. It can communicate with Server 2 and PC 3 via VLAN 1, and to Server 2, PC 1 and PC 2 via VLAN 2. The associated external switch port is a member of VLAN 1 and VLAN 2, and has tagging enabled.

Note: VLAN tagging is required only on ports that are connected to other 1/10Gb LAN Switch Modules or on ports that connect to tag-capable end-stations, such as servers with VLAN-tagging adapters.

Protocol-Based VLANs

Protocol-based VLANs (PVLANs) allow you to segment network traffic according to the network protocols in use. Traffic for supported network protocols can be confined to a particular port-based VLAN. You can give different priority levels to traffic generated by different network protocols.

With PVLAN, the switch classifies incoming packets by Ethernet protocol of the packets, not by the configuration of the ingress port. When an untagged or priority-tagged frame arrives at an ingress port, the protocol information carried in the frame is used to determine a VLAN to which the frame belongs. If a frame’s protocol is not recognized as a pre-defined PVLAN type, the ingress port’s PVID is assigned to the frame. When a tagged frame arrives, the VLAN ID in the frame’s tag is used.

Each VLAN can contain up to eight different PVLANs. You can configure separate PVLANs on different VLANs, with each PVLAN segmenting traffic for the same protocol type. For example, you can configure PVLAN 1 on VLAN 2 to segment IPv4 traffic, and PVLAN 8 on VLAN 100 to segment IPv4 traffic.

To define a PVLAN on a VLAN, configure a PVLAN number (1-8) and specify the frame type and the Ethernet type of the PVLAN protocol. You must assign at least one port to the PVLAN before it can function. Define the PVLAN frame type and Ethernet type as follows:

- Frame type—consists of one of the following values:
  - Ether2 (Ethernet II)
  - SNAP (Subnetwork Access Protocol)
  - LLC (Logical Link Control)
- Ethernet type—consists of a 4-digit (16 bit) hex value that defines the Ethernet type. You can use common Ethernet protocol values, or define your own values. Following are examples of common Ethernet protocol values:
  - IPv4 = 0800
  - IPv6 = 86dd
  - ARP = 0806

Port-Based vs. Protocol-Based VLANs

Each VLAN supports both port-based and protocol-based association, as follows:

- The default VLAN configuration is port-based. All data ports are members of VLAN 1, with no PVLAN association.
- When you add ports to a PVLAN, the ports become members of both the port-based VLAN and the PVLAN. For example, if you add port EXT1 to PVLAN 1 on VLAN 2, the port also becomes a member of VLAN 2.
- When you delete a PVLAN, it’s member ports remain members of the port-based VLAN. For example, if you delete PVLAN 1 from VLAN 2, port EXT1 remains a member of VLAN 2.
- When you delete a port from a VLAN, the port is deleted from all corresponding PVLANs.

PVLAN Priority Levels

You can assign each PVLAN a priority value of 0-7, used for Quality of Service (QoS). PVLAN priority takes precedence over a port’s configured priority level. If no priority level is configured for the PVLAN (priority = 0), each port’s priority is used (if configured).

All member ports of a PVLAN have the same PVLAN priority level.

PVLAN Tagging

When PVLAN tagging is enabled, the switch tags frames that match the PVLAN protocol. For more information about tagging, see “VLAN Tagging/Trunk Mode” on page 3-5.

Untagged ports must have PVLAN tagging disabled. Tagged ports can have PVLAN tagging either enabled or disabled.

PVLAN tagging has higher precedence than port-based tagging. If a port is tag enabled, and the port is a member of a PVLAN, the PVLAN tags egress frames that match the PVLAN protocol.

Use the tag-pvlan command (`vlan <x> protocol-vlan <x> tag-pvlan <x>`) to define the complete list of tag-enabled ports in the PVLAN. Note that all ports not included in the PVLAN tag list will have PVLAN tagging disabled.

PVLAN Configuration Guidelines

Consider the following guidelines when you configure protocol-based VLANs:

- Each port can support up to 16 VLAN protocols.
- 1/10Gb LAN Switch Module can support up to 16 protocols simultaneously.
- Each PVLAN must have at least one port assigned before it can be activated.
- The same port within a port-based VLAN can belong to multiple PVLANs.
- An untagged port can be a member of multiple PVLANs.
- A port cannot be a member of different VLANs with the same protocol association.

Configuring PVLAN

Follow this procedure to configure a Protocol-based VLAN (PVLAN).

1. Configure VLAN tagging/trunk mode for ports.

```
Router(config)# interface port 1,2
Router(config-if)# switchport mode trunk
Router(config-if)# exit
```

2. Create a VLAN and define the protocol type(s) supported by the VLAN.

```
Router(config)# vlan 2
Router(config-vlan)# protocol-vlan 1 frame-type ether2 0800
```

3. Configure the priority value for the protocol.

```
Router(config-vlan)# protocol-vlan 1 priority 2
```

4. Add member ports for this PVLAN.

```
Router(config-vlan)# protocol-vlan 1 member 1,2
```

Note: If VLAN tagging is turned on and the port being added to the VLAN has a different default VLAN (PVID/Native VLAN), you will be asked to confirm changing the PVID to the current VLAN.

5. Enable the PVLAN.

```
Router(config-vlan)# protocol-vlan 1 enable
Router(config-vlan)# exit
```

6. Verify PVLAN operation.

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 secondary VLAN, as follows:

- Primary VLAN—carries unidirectional traffic downstream from promiscuous ports. Each Private VLAN 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 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 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 an external 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). Each promiscuous port can belong to only one Private VLAN.
- 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:

- Management VLANs cannot be Private VLANs. Management ports cannot be members of a Private VLAN.
- 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.

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

2. Configure a promiscuous port for VLAN 700.

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

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

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

4. Map secondary VLANs to primary VLAN.

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

5. Configure host ports for secondary VLANs.

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

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

6. Verify the configuration.

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

Primary Secondary     Type          Ports
------- --------- --------------- ---------------------------------
700      701           solated       1 2
700      702           community     1 3
```

Ports and Trunking

Trunk groups can provide super-bandwidth, multi-link connections between 1/10Gb LAN Switch Module 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 Port Modes" on page 3-15
- "Trunking Overview" on page 3-16
- "Static Trunks" on page 3-16
- "Configurable Trunk Hash Algorithm" on page 3-19
- "Link Aggregation Control Protocol" on page 3-20

Configuring Port Modes

The switch allows you to set the port mode. Select the port mode that fits your network configuration.

The following port modes are available:

- **Base** port mode:
  - Fourteen 1Gb internal ports (1 port x 14 blade servers)
  - Ten 1Gb external ports
- **Key1** port mode:
  - Twenty-Eight 1Gb internal ports (2 ports x 14 blade servers)
  - Twenty 1Gb external ports
- **Key2** port mode:
  - Twenty-Eight 1Gb internal ports (2 ports x 14 blade servers)
  - Ten 1Gb external ports
  - Four 10Gb SFP+ external ports

Base port mode is the default. You may upgrade the port mode using **Key1**, **Key2**, or both. To upgrade the port mode, you must obtain a software license key.

Use the following command to enter the software license key to upgrade the port mode:

```
Router(config)# software-key enakey
```

After you enter the license key, you must reset the switch (/boot/reset) for the change to take affect. Use the following command to verify the port configuration:

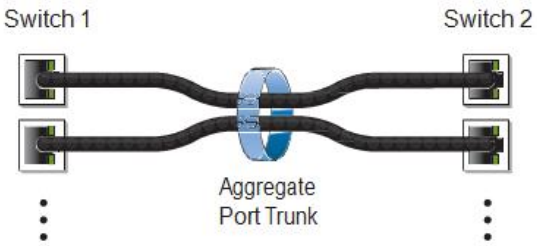
```
Router(config)# show interface information

Alias   Port Tag RMON Lrn Fld PVID       NAME       VLAN(s)
------- ---- --- ---- --- --- ----- -------------- -----------------------
INTA1    1    n   d    e   e   1      INTA1          1
INTA2    2    n   d    e   e   1      INTA2          1
INTA3    3    n   d    e   e   1      INTA3          1
INTA4    4    n   d    e   e   15     INTA4          15
INTA5    5    n   d    e   e   1      INTA5          1
INTA6    6    n   d    e   e   1      INTA6          1
INTA7    7    n   d    e   e   1      INTA7          1
INTA8    8    n   d    e   e   1      INTA8          1
INTA9    9    n   d    e   e   1      INTA9          1
INTA10   10   n   d    e   e   1      INTA10         1
INTA11   11   n   d    e   e   1      INTA11         1
INTA12   12   n   d    e   e   1      INTA12         1
INTA13   13   n   d    e   e   1      INTA13         1
INTA14   14   n   d    e   e   1      INTA14         1
INTB1    15   n   d    e   e   1      INTB1          1
INTB2    16   n   d    e   e   1      INTB2          1
...
EXT18    46   n   d    e   e   1      EXT18          1
EXT19    47   n   d    e   e   1      EXT19          1
EXT20    48   n   d    e   e   1      EXT20          1
MGT1     53   y   d    e   e   4095   MGT1           4095

* = PVID is tagged.
```

Trunking Overview

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

Two trunk types are available: static trunk groups (portchannel), and dynamic LACP trunk groups. Up to 52 trunks of each type are supported, either static, LACP or mixed, depending of the number and type of available ports. Each trunk can include up to 32 member ports .

Figure 8. Port Trunk Group



Trunk groups are also useful for connecting 1/10Gb LAN Switch Module 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

Before Configurating Static Trunks

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:

- Read the configuration rules provided in the section, “Static Trunk Group Configuration Rules” on page 3-17.”
- Determine which switch ports are to become *trunk members* (the specific ports making up the trunk).
- Ensure that the chosen switch ports are set to enabled.
- Ensure all member ports in a trunk have the same VLAN configuration.

- Consider how the existing Spanning Tree will react to the new trunk configuration. See “Spanning Tree Protocols” on page 3-22 for configuration guidelines.
- Consider how existing VLANs will be affected by the addition of a trunk.

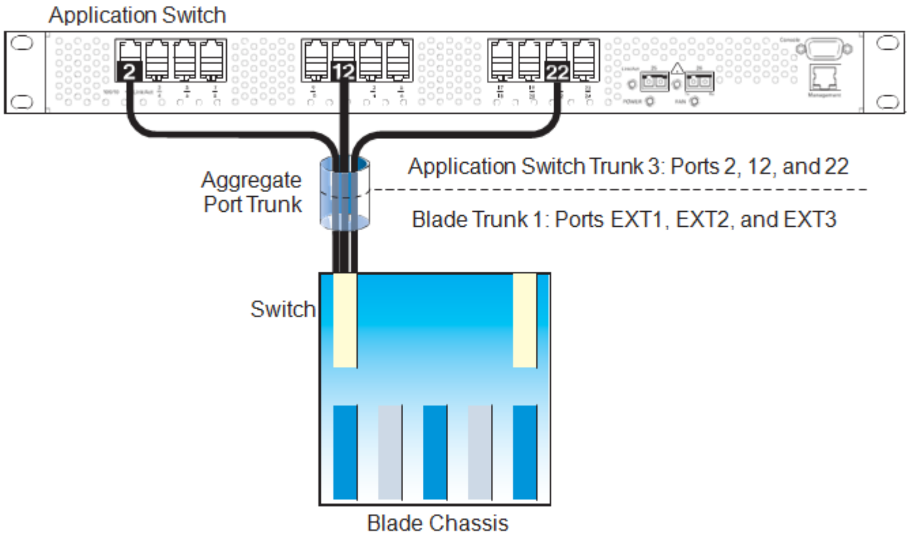
Static Trunk Group Configuration Rules

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 network entity (a single device, or multiple devices acting in a stack) and lead to one destination entity. For example, you cannot combine links from two different servers into one trunk group.
- Any physical switch port can belong to only one trunk group.
- Depending on port availability, the switch supports up to 8 ports in each trunk group.
- Internal ports (INT*x*) and external ports (EXT*x*) cannot become members of the same trunk group.
- Trunking from third-party devices must comply with Cisco® EtherChannel® technology.
- All trunk member ports must be assigned to the same VLAN configuration before the trunk can be enabled.
- If you change the VLAN settings of any trunk member, you cannot apply the change until you change the VLAN settings of all trunk members.
- When an active port is configured in a trunk, the port becomes a *trunk member* when you enable the trunk. The Spanning Tree parameters for the port then change to reflect the new trunk settings.
- All trunk members must be in the same Spanning Tree Group (STG) and can belong to only one Spanning Tree Group (STG). However if all ports are *tagged*, then all trunk ports can belong to multiple STGs.
- If you change the Spanning Tree participation of any trunk member to enabled or disabled, the Spanning Tree participation of all members of that trunk should be changed similarly.
- When a trunk is enabled, the trunk Spanning Tree participation setting takes precedence over that of any trunk member.
- 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.
- All ports in static trunks must have the same link configuration (speed, duplex, flow control).

Configurating a Static Port Trunk

In the example below, three ports are trunked between two switches. Figure 9. Port Trunk Group Configuration Example


Application Switch
2
12
22
Aggregate Port Trunk
Application Switch Trunk 3: Ports 2, 12, and 22
Blade Trunk 1: Ports EXT1, EXT2, and EXT3
Switch
Blade Chassis

Prior to configuring each switch in the above example, you must connect to the appropriate switch's Command Line Interface (CLI) as the administrator.

Note: For details about accessing and using any of the menu commands described in this example, see the Networking OS *Command Reference*.

1. Connect the switch ports that will be members in the trunk group.
2. Configure the trunk using these steps on 1/10Gb LAN Switch Module:

   a. Define a trunk group.

```
Router(config)# portchannel 1 port ext1,ext2,ext3   (Add port s to trunk group 1)
Router(config)# portchannel 1 enable         (Enable trunk group 1)
```

   b. Verify the configuration.

```
Router(config)# show portchannel information
```

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

3. Repeat the process on the other switch.

```
Router(config)# portchannel 3 port 2,12,22
Router(config)# portchannel 3 enable
```

Trunk group 1 (on 1/10Gb LAN Switch Module) is now connected to trunk group 3 on the Application Switch.

Note: In this example, 1/10Gb LAN Switch Module and an application switch 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 should 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.

```
Router# show portchannel information   (View trunking information)
```

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.

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.

1/10Gb LAN Switch Module supports the following hashing options, which can be used in any combination:

- Source MAC address (smac)

```
Router(config)# portchannel hash source-mac-address
```

- Destination MAC address (dmac)

```
Router(config)# portchannel hash destination-mac-address
```

- Both source and destination MAC address (enabled by default)

```
Router(config)# portchannel hash source-destination-mac
```

- For Layer 3 IPv4/IPv6 traffic, one of the following are permitted:
  - Source IP address (sip)

```
Router(config)# portchannel hash source-ip-address
```

- Destination IP address (dip)

```
Router(config)# portchannel hash destination-ip-address
```

- Both source and destination IP address (enabled by default)Ingress port

```
Router(config)# portchannel hash source-destination-ip
```

number (disabled by default)

```
Router(config)# portchannel hash ingress
```

- Layer 4 port information (disabled by default)

```
Router(config)# portchannel hash L4port
```

When enabled, Layer 4 port information (TCP, UPD, etc.) is added to the hash if available. The L4port option is ignored when Layer 4 information is not included in the packet (such as for Layer 2 packets), or when the useL2 option is enabled.

Note: For MPLS packets, Layer 4 port information is excluded from the hash calculation. Instead, other IP fields are used, along with the first two MPLS labels.

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.

IEEE 802.3ad 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 or links of the dynamic trunk group.

1/10Gb LAN Switch Module supports up to 52 LACP trunks, each with up to 8 ports.
Note: LACP implementation in Networking OS 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 *system ID* and the port’s *admin key*, as follows:

- **System ID**: an integer value based on the 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 1/10Gb LAN Switch Module 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 (1/10Gb LAN Switch Module) and a Partner (another switch), as shown in Table 16.

Table 16. Actor vs. Partner LACP configuration

Actor Switch	Partner Switch 1
Port 38 (admin key = 100)	Port 1 (admin key = 50)
Port 39 (admin key = 100)	Port 2 (admin key = 50)

In the configuration shown in Table 16, Actor switch ports 38 and 39 aggregate to form an LACP trunk group with Partner switch ports 1 and 2. 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.

Each port in 1/10Gb LAN Switch Module 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.

If an LACP group member port is connected to a port that is in LACP off mode, the LACP port will not be able to converge. The link remains up, but the port is set to discarding state.

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.

```
Router # show lacp information
```

Static trunks are listed as trunks 1 though 52. Dynamic trunks are listed as 53 through 104.

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 1/10Gb LAN Switch Module uses only the most efficient network path.

This chapter covers the following topics:

- "Spanning Tree Protocol Modes" on page 3-22
- "Global STP Control" on page 3-23
- "PVSRT Mode" on page 3-23
- "Rapid Spanning Tree Protocol" on page 3-35
- "Multiple Spanning Tree Protocol" on page 3-36
- "Port Type and Link Type" on page 3-39

Spanning Tree Protocol Modes

Networking OS 7.8 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 in order 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 3-35 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 1/10Gb LAN Switch Module. See "PVSRT Mode" on page 3-23 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 3-36 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 or re-enabled using the following commands:

```
Router(config)# spanning-tree mode disable    (Globally disable Spanning Tree)
```

Spanning Tree can be re-enabled by specifying the STP mode:

```
Router(config)# spanning-tree mode {pvrst|rstp|mst}
```

PVSRT Mode

Note: Per-VLAN Rapid Spanning Tree (PVRST) is enabled by default on 1/10Gb LAN Switch Module.

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.

Networking 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, with each STG acting 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 17.

Table 17. Ports, Trunk Groups, and VLANs

Switch Element	**Belogs To**
Port	Trunk group, or one or more VLANs
Trunk Group	One or more VLANs
VLAN (non-default)	• 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 3-39) may bypass the `discarding`and `learning`states, and enter directly into the forwardingstate.

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 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, 1/10Gb LAN Switch Module 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:

```
Router(config)# spanning-tree stp <x> 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 commands to configure the port priority:

```
Router(config)# interface port <x>
Router(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.

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:

```
Router(config)# interface port <port number>
Router(config-if)# spanning-tree guard root
```

The default state is none (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:

```
Router(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:

```
Router(config-if)# spanning-tree guard loop none
```

To enable loop guard at the port level, enter the following command:

```
Router(config)# interface port <port number>
Router(config-if)# spanning-tree guard loop
```

The default state is “none”, i.e. 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 cost of a port also depends on whether the port operates at full-duplex (lower cost) or half-duplex (higher cost). For example, if a 100-Mbps (Fast Ethernet) link has a "cost" of 10 in half-duplex mode, it will have a cost of 5 in full-duplex mode. 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:

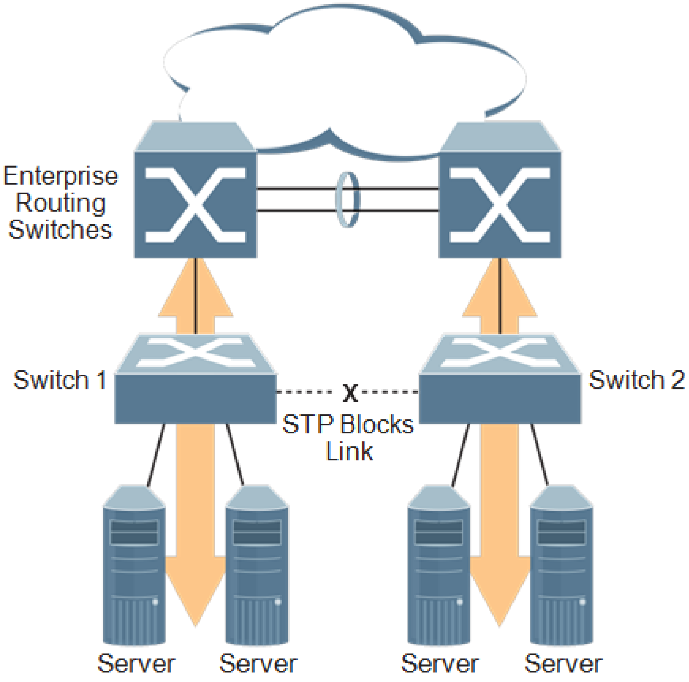
```
Router(config)# interface port <port number>
Router(config-if)# spanning-tree stp <STG> path-cost <path cost value>
Router(config-if)# exit
```

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 switches (via either external ports or internal Inter-Switch Links).

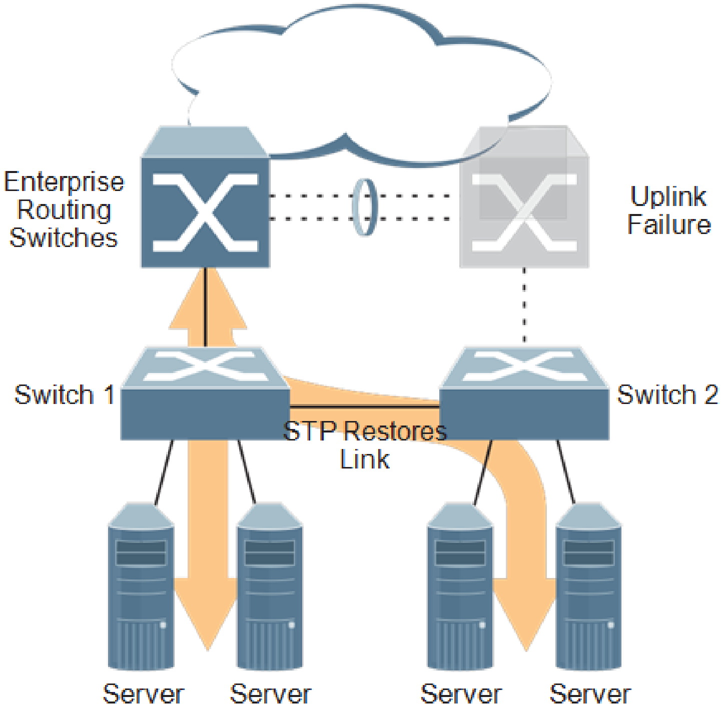
Figure 10. Spanning Tree Blocking a Switch-to-Switch Link



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 blade switches, and not one of 1/10Gb LAN Switch Module uplinks or the Enterprise switch trunk.

During operation, if one 1/10Gb LAN Switch Module experiences an uplink failure, STP will activate the switch-to-switch link so that server traffic on the affected 1/10Gb LAN Switch Module may pass through to the active uplink on the other 1/10Gb LAN Switch Module, as shown in Figure 11.

Figure 11. Spanning Tree Restoring the Switch-to-Switch Link



In this example, port 10 on each switch is used for the switch-to-switch link. To ensure that 1/10Gb LAN Switch Module 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 switch.

```
Router(config)# interface port 10
Router(config-if)# spanning-tree stp 1 path-cost 60000
Router(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 1/10Gb LAN Switch

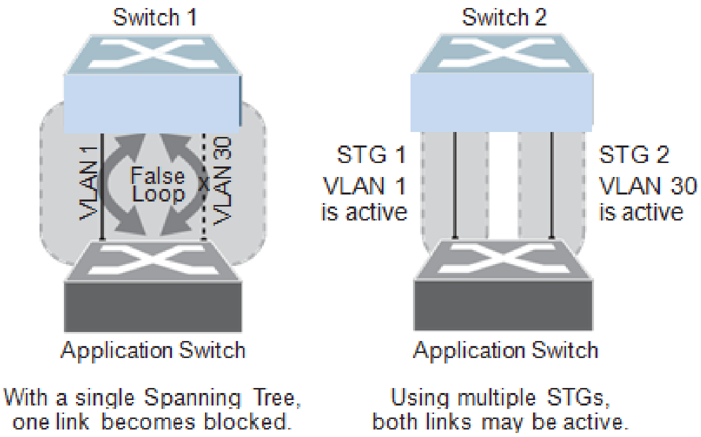
Modules using multiple STGs, configure each path with a different VLAN and then assign each VLAN to a separate STG. Since each STG is independent, they each send their own IEEE 802.1Q tagged Bridge Protocol Data Units (BPDUs).

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 2-128 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 1/10Gb LAN Switch Module 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 1/10Gb LAN Switch Module, 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.

VLAN 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.

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 3-30).

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 command:

```
Router(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 3-30 for information about creating VLANs and assigning ports to them.

2. Assign the VLAN to an STG using one of the following methods:
   - From the global configuration mode:

```
Router(config)# spanning-tree stp <STG number>  vlan <VLAN>
```

   - Or from within the VLAN configuration mode:

```
Router(config)# vlan <VLAN number>
Router(config-vlan)# stg <STG number>
Router(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

- 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 3-30. 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 Ports

- Tagged ports can belong to more than one STG, but untagged ports can belong to only one STG.
- 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.

Adding and Removing 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 that 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 not automatically add the port to STG 1. Instead, the switch will prompt you to decide whether to change the PVID from 3 to 1:

  ```
  "Port 5 is an UNTAGGED/Access Mode port and its current PVID/Native VLAN is 3.
  Confirm changing PVID/Native VLAn from 3 to 1 [y/n]:" y
  ```

- 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 2.
- 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.

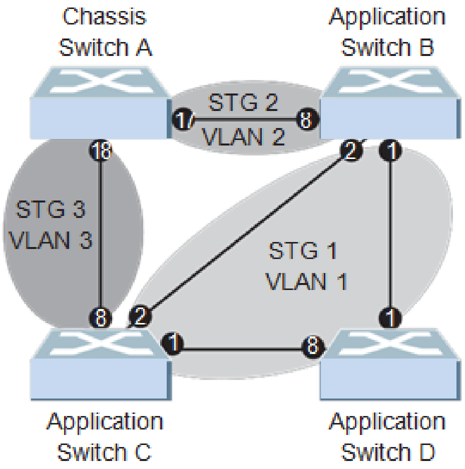
The relationship between port, trunk groups, VLANs, and Spanning Trees is shown in Table 17 on page 3-23.

Switch-Centric Configuration

PVRST is switch-centric: STGs are enforced only on the switch where they are configured. 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, though VLAN 2 is shared by the Switch A and Switch B, 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 3-32 is as follows:

- VLAN 1 Participation

Assuming Switch B to be the root bridge, Switch B transmits the BPDU for VLAN 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 3-32.

Because VASA is enabled by default, each new VLAN is automatically assigned its own STG. However, for this configuration example, some VLANs are explicitly reassigned to other STGs.

1. Set the Spanning Tree mode on each switch to PVRST.

```
Router(config)# spanning-tree mode pvrst
```

Note: PVRST is the default mode on 1/10Gb LAN Switch Module. 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:
3. Enable VLAN 2 and VLAN 3.

```
Router(config)# vlan 2
Router(config-vlan)# exit
Router(config)# vlan 3
Router(config-vlan)# exit

If VASA is disabled, enter the following commands:
Router(config)# spanning-tree stp 2 vlan 2
Router(config)# spanning-tree stp 3 vlan 3
```

Add port 17 to VLAN 2, port 18 to VLAN 3.

```
Router(config)# interface port 17
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk allowed vlan add 2
Router(config-if)# exit

Router(config)# interface port 18
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk allowed vlan add 3
Router(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.

4.   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.

```
Router(config)# vlan 2
Router(config-vlan)# stg 2
Router(config-vlan)# exit
Router(config)# interface port 8
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk allowed vlan add 2
Router(config-if)# exit

If VASA is disabled, enter the following command: Router(config)#
spanning-tree stp 2 vlan 2
```

VLAN 2 is automatically removed from STG 1. By default VLAN 1 remains in STG 1.

5.   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.

```
Router(config)# vlan 3
Router(config-vlan)# stg 3
Router(config-vlan)# exit
Router(config)# interface port 8
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk allowed vlan add 3
Router(config-if)# exit

If VASA is disabled, enter the following command:
Router(config)# spanning-tree stp 3 vlan 3
```

VLAN 3 is automatically removed from STG 1. By default VLAN 1 remains in STG 1.

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.

Note: In RSTP mode, Spanning Tree for the management ports is turned off by default.

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* ports ("Port Type and Link Type" on page 3-39) 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.

```
Router(config)# spanning-tree mode rstp
```

3. Configure RSTP parameters.

```
Router(config)# spanning-tree stp 1 bridge priority 8192
Router(config)# spanning-tree stp 1 bridge hello-time 5
Router(config)# spanning-tree stp 1 bridge forward-delay 20
Router(config)# spanning-tree stp 1 bridge maximum-age 30
Router(config)# no spanning-tree stp 1 enable
```

4. Configure port parameters:

```
Router(config)# interface port 3
Router(config-if)# spanning-tree stp 1 priority 240
Router(config-if)# spanning-tree stp 1 path-cost 500
Router(config-if)# no spanning-tree stp 1 enable
Router(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, 1/10Gb LAN Switch Module supports up to 32 instances of Spanning Tree, corresponding to STGs 1-32, with each STG acting as an independent, simultaneous instance of STP.

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 ports (“Port Type and Link Type” on page 3-39) 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) 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, Edge port enable/disable, and Link Type. 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 automatically moves management VLAN 4095 to the CIST. When MSTP is turned off, the switch moves VLAN 4095 from the CIST to Spanning Tree Group 128.
- When you enable MSTP, you must configure the Region Name. A default version number of 0 is configured automatically.
- Each bridge in the region must have the same name, version number, and VLAN mapping.

MSTP Configuration Examples

Example 1

This section provides steps to configure MSTP on 1/10Gb LAN Switch Module.

1. Configure port and VLAN membership on the switch.
2. Configure Multiple Spanning Tree region parameters and set the mode to MSTP.

```
Router(config)# spanning-tree mst configuration     (Enter MST configuration mode)
Router(config-mst)# name <name>      (Define the Region name)
Router(config-mst)# revision <0-65535>
Router(config-mst)# exit
Router(config)# spanning-tree mode mst       (Set mode to Multiple Spanning Trees)
```

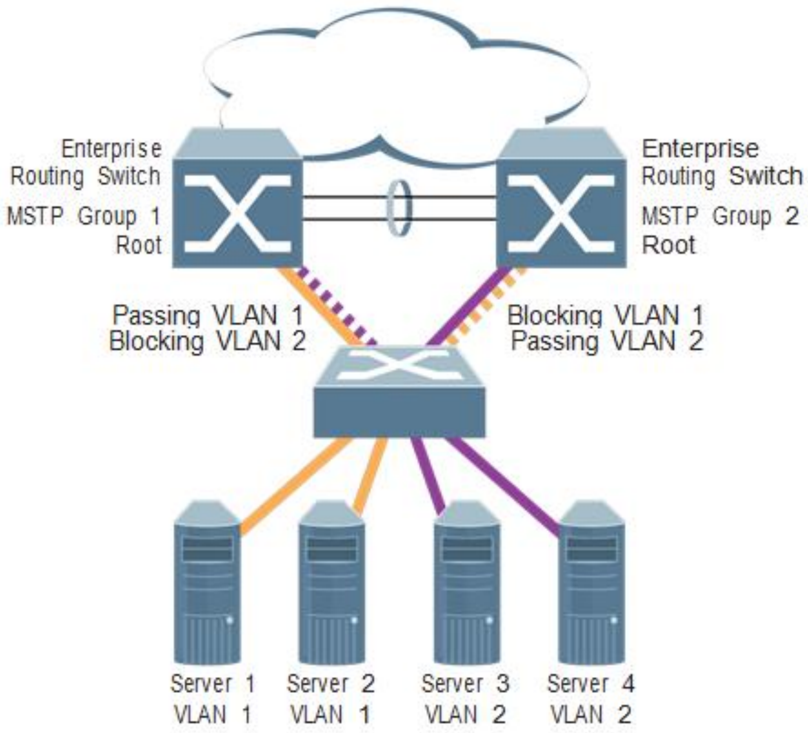
3. Map VLANs to MSTP instances:

```
Router(config)# spanning-tree mst configuration     (Enter MST configuration mode)
Router(config-mst)# instance <instance ID> vlan <vlan number or range>
```

MSTP Configuration Example 2

This configuration shows how to configure MSTP Groups on the switch, as shown in Figure 14.

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.

```
Router(config)# interface port 19,20
Router(config-if)# switchport mode trunk
Router(config-if)# exit
```

2. Configure MSTP: Spanning Tree mode, region name, and version.

```
Router(config)# spanning-tree mst configuration
Router(config-mst)# name MyRegion (Define the Region name)
Router(config-mst)# revision 100  (Define the Revision level)
Router(config-mst)# exit
Router(config)# spanning-tree mode mst  (Set mode to Multiple Spanning
Trees)
```

3. Map VLANs to MSTP instances:

```
Router(config)# spanning-tree mst configuration
Router(config-mst)# instance 1 vlan 1
Router(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.

```
Router(config)# interface port 1,2,19,20
Router(config-if)# switchport trunk allowed vlan add 1
Router(config-if)# exit
```

5. Configure port membership and define the STGs for VLAN 2. Add server ports 3, 4, and 5 to VLAN 2. Assign VLAN 2 to STG 2.

```
Router(config)# interface port 3,4,5
Router(config-if)# switchport access vlan 2
Router(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. Internal ports (INTx) should be configured as edge ports.

Edge ports send BPDUs to upstream STP devices like normal STP ports, but should 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:

```
Router(config)# interface port <port>
Router(config-if)# [no] spanning-tree portfast
Router(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:

```
Router(config)# interface port <port>
Router(config-if)# [no] spanning-tree link-type <type>
Router(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.

Quality of Service

Quality of Service (QoS) 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 QoS level.

The following topics are discussed in this section:

- "QoS Overview" on page 3-41
- "Using ACL Filters" on page 3-42
- "Using DSCP Values to Provide QoS" on page 3-44
- "Using 802.1p Priorities to Provide QoS" on page 3-48
- "Queuing and Scheduling" on page 3-49
- "Packet Drop Logging" on page 3-50

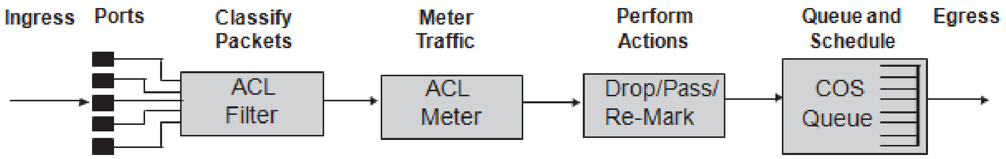
QoS Overview

QoS helps you allocate guaranteed bandwidth to 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 those that cannot tolerate delay, assigning that 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 15 on page 3-41 shows the basic QoS model used by 1/10Gb LAN Switch Module.

Figure 15. QoS Model



1/10Gb LAN Switch Module uses the Differentiated Services (DiffServ) architecture to provide QoS functions. DiffServ is described in IETF RFC 2474 and RFC 2475.

With DiffServ, you can establish policies for directing 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.

1/10Gb LAN Switch Modul 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 1/10Gb LAN Switch Module 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.

The basic 1/10Gb LAN Switch Module QoS model works as follows:

- Classify traffic:
  - Read DSCP
  - Read 802.1p Priority
  - Match ACL filter parameters
- Meter traffic:
  - Define bandwidth and burst parameters
  - Select actions to perform on in-profile and out-of-profile traffic
- Perform actions:
  - Drop packets
  - Pass 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 available COS queues
  - Schedule transmission based on the COS queue weight

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 conditions that packets must match for inclusion in a particular service class, and also the actions that are performed for matching traffic.

1/10Gb LAN Switch Module 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
- IPv6—source IP address/prefix, destination address/prefix, next header, flow label, traffic class
- TCP/UPD—source port, destination port, TCP flag
- Packet format—Ethernet format, tagging format, IPv4, IPv6
- Egress port

For ACL details, see “Access Control Lists” on page 2-28.

Summary of ACL Actions

Actions determine how the traffic is treated. 1/10Gb LAN Switch Module QoS 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

ACL Metering and Re-Marking

You can define a profile for the aggregate traffic flowing through 1/10Gb LAN Switch Module by configuring a QoS meter (if desired) and assigning ACL Groups to ports. When you add ACL Groups 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.

Note: Metering is not supported for IPv6 ACLs. All traffic matching an IPv6 ACL is considered in-profile for re-marking purposes.

Using meters, you set a Committed Rate in Kbps (1000 bits per second in each 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 traffic should receive.
- Change the 802.1p priority of a packet.

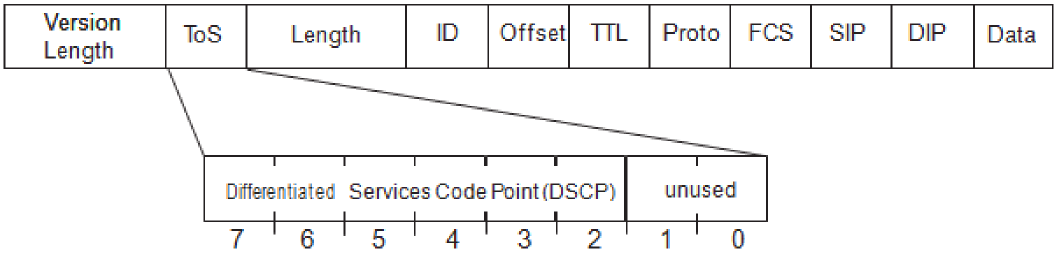
Using DSCP Values to Provide QoS

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.

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 16. Layer 3 IPv4 Packet



1/10Gb LAN Switch Module 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 an 802.1p priority

Once the DSCP value is marked, 1/10Gb LAN Switch Module can use it to direct traffic prioritization.

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 Router 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 below. 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)

- Assured Assured Class Selector (CS)—This PHB has eight priority classes, with CS7 repre- senting the highest priority, and CS0 representing the lowest priority, as shown below. 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 18 shows the default service levels provided by 1/10Gb LAN Switch Module, listed from highest to lowest importance:

Table 18. 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

DSCP Re-Marking Overview

1/10Gb LAN Switch Module can re-mark the DSCP value of ingress packets to a new value, and set the 802.1p priority value, based on the DSCP value. You can view the settings by using the following command:

```
Router(config)# show qos dscp
Current DSCP Remarking Configuration: OFF
DSCP      New DSCP  New 802.1p Prio
--------  --------  ----------------
0         0         0
1         1         0
...
51        51        0
52        52        0
53        53        0
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:

```
Router(config)# qos dscp re-marking
```

Then you must enable DSCP re-marking on any port that you wish to perform this function.

Note: If an ACL meter is configured for DSCP re-marking, the meter function takes precedence over QoS re-marking.

DSCP Re-Marking Configuration Example

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.

```
Router(config)#  qos dscp re-marking
Router(config)#  qos dscp dscp-mapping <DSCP value (0-63)> <new value>
Router(config)#  qos dscp dot1p-mapping <DSCP value (0-63)> <802.1p value>
```

2. Enable DSCP re-marking on a port.

```
Router(config)# interface port 1
Router(config-if)#  qos dscp re-marking
Router(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.

```
Router(config)# access-control list 2 tcp-udp source-port 5060 0xffff
Router(config)# access-control list 2 meter committed-rate 10000000
Router(config)# access-control list 2 meter enable
Router(config)# access-control list 2 re-mark in-profile dscp 56
Router(config)# access-control list 2 re-mark dot1p 7
Router(config)# access-control list 2 action permit
```

2. Create an ACL to set a low priority to all other traffic.

```
Router(config)# access-control list 3 action set-priority 1
Router(config)# access-control list 3 action permit
```

3. Apply the ACLs to a port and enable DSCP marking.

```
Router(config)# interface port 5
Router(config-if)# access-control list 2
Router(config-if)# access-control list 3 ethernet source-mac-address
00:00:00:00:00:00 00:00:00:00:00:00
Router(config-if)# dscp-marking
Router(config-if)# exit
```

4. Enable DSCP re-marking globally.

```
Router(config)# qos dscp re-marking
```

5. Assign the DSCP re-mark value.

```
Router(config)# qos dscp dscp-mapping 40 9
Router(config)# qos dscp dscp-mapping 46 9
```

6. Assign strict priority to VoIP COS queue.

```
Router(config)# qos transmit-queue weight-cos 7 0
```

7. Map priority value to COS queue for non-VoIP traffic.

```
Router(config)# qos transmit-queue mapping 1 1
```

8. Assign weight to the non-VoIP COS queue.

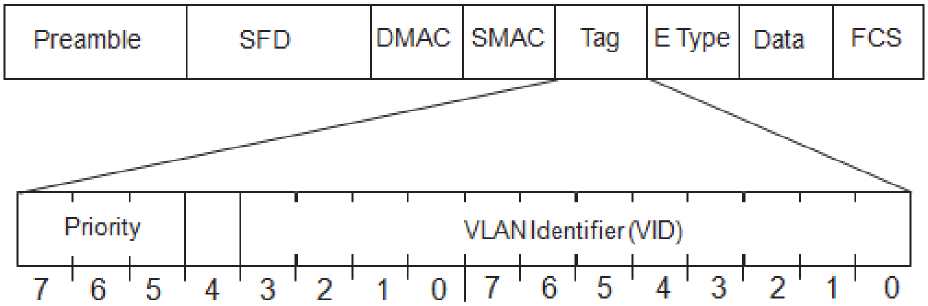
```
Router(config)# qos transmit-queue weight-cos 1 2
```

Using 802.1p Priorities to Provide QoS

802.1p Overview
Networking OS 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 that should be given to packets during forwarding. Packets with a numerically higher (non-zero) priority are given forwarding preference over packets with lower priority bit 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. 1/10Gb LAN Switch Module can filter packets based on the 802.1p values, and it can assign or overwrite the 802.1p value in the packet.

Figure 17. Layer 2 802.1q/802.1p VLAN Tagged Packet



Ingress packets receive a priority value, as follows:

- **Tagged packets**—1/10Gb LAN Switch Module reads the 802.1p priority in the VLAN tag.
- Untagged packets—1/10Gb LAN Switch Module tags the packet and assigns an 802.1p priority, based on the port's default priority.

Egress packets are placed in a COS queue based on the priority value, and scheduled for transmission based on the scheduling weight of the COS queue.

To configure a port's default 802.1p priority value, use the following commands.

```
Router(config)# interface port 1
Router(config-if)# dot1p <802.1p value (0-7)>
Router(config-if)# exit
```

See "Queuing and Scheduling" on page 3-48 for details on scheduling weights.

Queuing and Scheduling

1/10Gb LAN Switch Module can be configured to have either 2 or 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:

```
Router(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.

```
Router(config)# qos transmit-queue weight-cos <COSq number> <COSq weight (0-15)>
```

The scheduling weight can be set from 0 to 15. Weight values from 1 to 15 set the queue to use weighted round-robin (WRR) scheduling, which distributes larger numbers of packets to queues with the highest weight values. For distribution purposes, each packet is counted the same, regardless of the packet’s size.

A scheduling weight of 0 (zero) indicates strict priority. Traffic in strict priority queue has precedence over other all queues. If more than one queue is assigned a weight of 0, the strict queue with highest queue number will be served first. Once all traffic in strict queues is delivered, any remaining bandwidth will be allocated to the WRR queues, divided according to their weight values.

Note: Use caution when assigning strict scheduling to queues. Heavy traffic in queues assigned with a weight of 0 can starve lower priority queues.

For a scheduling method that uses a weighted deficit round-robin (WDRR) algorithm, distributing packets with an awareness of packet size, see “Enhanced Transmission Selection”.

Packet Drop Logging

Packet drop logging allows you to monitor network deficiencies by generating syslog messages for packet drops in the CPU queues. By default, the switch will generate such messages once every 30 minutes, specifying the type of traffic, queue data rate and queue number on which the drops occurred, such as:

```
Apr 19 11:27:35 172.31.37.200 NOTICE Protocol control discards: ARP Broadcast
packets are received at rate higher than 200pps, hence are discarded on queue 5.
```

To enable or disable packet drop logging, use the following commands:

```
Router(config)# [no] logging pdrop enable
```

You can adjust the logging interval between 0 and 30 minutes using the following command:

```
Router(config)# logging pdrop interval <0-30>
```

Setting the logging interval to 0 will log packet drops immediately (with up to 1 second delay), and will ignore further drops on the same queue during the next 2 minutes.

Setting the logging interval to a greater value (1 – 30 minutes), regularly displays packet drop information at the designated time intervals. Once the packet drops stop, or if new packet drops are encountered only within 2 minutes after a syslog message, the switch does not display any more messages.

4

Advanced Switching Features

This section discusses the Advanced Switching Features.

□ Virtualization

□ VMready

□ Static Multicast ARP

□ Switch Partition

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 Networking OS 7.8 on 1/10Gb LAN Switch Module:

- 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 3-2.

- 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 3-14.

- 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 4-3.

Networking OS virtualization features provide a highly-flexible framework for allocating and managing switch resources.

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 Networking OS 7.8 VMready feature supports up to2048 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.

1/10Gb LAN Switch Module with VMready also detects the migration of VEs across different hypervisors. As VEs move, 1/10Gb LAN Switch Module 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 1/10Gb LAN Switch Module 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 7.8 supports up to 2048 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).

VM Group Types

VEs, as well as internal ports, external 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.

Networking OS 7.8 supports up to 32 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 man- agement server (see “Assigning a vCenter” on page 4-12).

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 1/10Gb LAN Switch Module, may require manual configuration when using local VM groups).

Configuring a Local VM Group

Local VM groups are configured in the VM Group command path:

```
Router(config)# virt vmgroup <VM group number>
```

se the following CLI configuration commands to assign group properties and membership :

```
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)
```

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. Note that VM groups and vNICs (see “Virtual NICs” on page 4-6) are not supported simultaneously on the same port.
- `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 4-6).
- `stg`: The group may be assigned to a Spanning-Tree group for broadcast loop control (see‟Spanning Tree Protocols” on page 3-22).
- `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 validate 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 2-35)
- `vm`: Add VMs.
  VMs and other VEs are primarily specified by MAC address. They can also be specified by UUID or by the index number as shown in various VMready information output (see “VMready Information Displays” on page 4-17).

Distributed VM Groups

Distributed VM groups allow configuration profiles to be synchronized between 1/10Gb LAN Switch Module 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).

1/10Gb LAN Switch Module 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 4-12 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:

```
Router(config)# virt vmprofile <profile name>
```

Next, the profile must be edited and configured using the following configuration commands:

```
Router(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).

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 4-16.

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 4-6 for details.

Once a VM profile is available, a distributed VM group may be initialized using the following configuration command:

```
Router(config)# virt vmgroup <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.

```
Router(config)# no virt vmgroup <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 4-4) 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 4-4), 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 vlanoption (see page 4-6) 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 internal 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:

```
HITACHI_<VM profile name>
(or)
HITACHI_<VM profile name>_<index number> (for vDS profiles)
```

Using the VM Group command path (`Router(config)# virt vmgroup <x>  vm`) to add 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 to be sure the expected changes are properly applied.

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 Defaultport 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

1/10Gb LAN Switch Module 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:

```
Router(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:

```
Router(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 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 used for dropping traffic:

```
Router(config)# virt vmcheck acls max <1-640>
```

Use the following command to set the action to be performed if the switch is unable to validate the VM MAC address:

```
Router(config)# virt vmcheck action advanced {log|link|acl}
```

Following are the other VMcheck commands:

Table 19. VMcheck Commands

Command	Description
`Router(config)# virt vmware hello {ena| 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
`Router(config)# no virt vmware hello {enable|hport <port number>}`	Disable hello messages/remove port
`Router(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
`Router# no virt vmcheck acl [mac-address [<port number>]|port]`	Delete ACL(s): all ACLs/an ACL by MAC address ((optional) and port number) /all ACLs installed on a port

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:

```
Router# virt vmware dvswitch add <datacenter name> <dvSwitch name> [<dvSwitch-version>]
```

Prerequisites

Before adding a vDS on 1/10Gb LAN Switch Module, 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:
  - vDS: `Home> Inventory > Networking`
  - vDS Hosts: `Home > Inventory > Networking > vDS > Hosts`

  Note: These changes will not be displayed in the running configuration on 1/10Gb LAN Switch Module.

Migrating to vDS

You can migrate VMs to the vDS using vCenter. The migration may also be accomplished using the operational commands on 1/10Gb LAN Switch Module available in the following CLI menus:

For VMware vDS operations:

```
Router# 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:

```
Router# 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

1/10Gb LAN Switch Module 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. Networking OS 7.8 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 privi- leges:
    - Network
    - Host Network > Configuration
    - Virtual Machine > Modify Device Settings

Once vCenter requirements are met, the following configuration command can be used on 1/10Gb LAN Switch Module to associate the vCenter with the switch:

```
Router(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 to the switch to ignores 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, 1/10Gb LAN Switch Module 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 CLI privileged EXEC commands:

```
Router# virt vmware scan      (Scan the vCenter)
            -or-
Router# show virt vm -v -r    (Scan vCenter and display result)
```

Deleting the vCenter

To detach the vCenter from the switch, use the following configuration command:

```
Router(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 1/10Gb LAN Switch Module 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, 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 CLI privileged EXEC command:

```
Router# 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. If the virtual switch name is omitted, the administrator will be prompted to select one from a list or to enter a new virtual switch name.

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, it will be created with default properties, but with no uplink connection to a physical NIC (the administrator must assign uplinks using VMware management tools.

VMware Operational Commands

1/10Gb LAN Switch Module may be used as a central point of configuration for VMware virtual switches and port groups using the CLI privileged EXEC commands:

```
Router# 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:

```
Router(config)# [no] virt vmgroup <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 2-28). 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.

VMAPs are configured using the following CLI configuration command path:

```
Router(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
 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
```

Networking OS 7.8 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).

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

- For regular VLANs, use config-vlan mode:

```
Router(config)# vlan <VLAN ID>
Router(config-vlan)# [no] vmap <VMAP ID> [intports| extports]
```

- For a VM group, use the global configuration mode:

```
Router(config)# [no] virt vmgroup <ID> vmap <VMAP ID> [intports|extports]
```

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 intports or extports parameter can be specified to apply the action (to add or remove the VMAP) for either the internal ports or external ports only. 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 to or 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:

```
Router(config)# virt vmpolicy vmbwidth <VM MAC>|<index>|<UUID>| <IPv4 address>|<name>  ?
txrate <committed rate> <burst>  [<ACL number>]    (Set the VM to switch transmit rate)
rxrate <committed rate> <burst>  [<ACL number>]    (Set the VM to switch receive rate)
bwctrl  (Enable bandwidth control)
```

Bandwidth allocation can be defined either 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 Control (`rxrate`) 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 in that direction (TX or RX) 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 4-6) 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 1/10Gb LAN Switch Module. 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 1/10Gb LAN Switch Module.

VM Policy Bandwidth Control is configured per VE, and can be set independently for transmit and receive traffic. Bandwidth policies are enforced by 1/10Gb LAN Switch Module. VE traffic that exceeds configured levels is dropped by the switch upon ingress (for txrate) or before egress (for rxrate). Setting txrateuses ACL resources on the switch.

Bandwidth shaping and bandwidth policies can be used separately or in concert.

VMready Information Displays

1/10Gb LAN Switch Module 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 4-12), 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 4-13.

Local VE Information

A concise list of local VEs and pre-provisioned VEs is available with the following CLI privileged EXEC command:

```
Router# show virt vm

IP Address       VMAC Address          Index Port    VM Group (Profile) Check status
---------------- ------------------    ----- ------- ------------------ ------------
 *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 CLI privileged EXEC command:

```
Router# show virt vm -v

Index  MAC Address,           Name (VM or Host),  Port, Group Vswitch,
       IP Address             @Host (VMs only)    VLAN        Port Group
----- ------------            ------------------  ----- ----- ----------
0      00:50:56:9c:21:2f      atom                   4        vSwitch0
       172.16.46.15           @172.16.46.10          500      Eng_A

+1     00:50:56:72:ec:86      172.16.46.50           3        vSwitch0
       172.16.46.51                                  0        VMkernel

*2     00:50:56:4f:f2:85      172.16.46.10           4        vSwitch0
       172.16.46.10                                  0        Mgmt

+3     00:50:56:7c:1c:ca      172.16.46.10           4        vSwitch0
       172.16.46.11                                  0        VMkernel

*4     00:50:56:4e:62:f5      172.16.46.50           3        vSwitch0
       172.16.46.50                                  0        Mgmt

5      00:50:56:9c:00:c8      quark                  4        vSwitch0
       172.16.46.25           @172.16.46.10          0        Corp

6      00:50:56:9c:29:29      particle               3        vSwitch0
       172.16.46.35           @172.16.46.50          0        VM Network

7      00:50:56:9c:47:fd      nucleus                3        vSwitch0
       172.16.46.45           @172.16.46.50          0        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 4-20).

vCenter Hypervisor Hosts

If a vCenter is available, the following CLI privileged EXEC command displays the name and UUID of all VMware hosts, providing an essential overview of the data center:

```
Router# show virt vmware hostsUUID      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:

```
Router# show virt vmware showhost {<UUID>|<IPv4 address>|<host
name>}Vswitches available on the host:
vSwitch0
Port Groups and their Vswitches on the host:
             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          Default
VLAN ID             0
...
```

vCenter VEs

If a vCenter is available, the following CLI privileged EXEC command displays a list of all known VEs:

```
Router# show virt vmware vmsUUID        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 CLI privileged EXEC command displays detailed information about a specific VE:

```
Router# 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             Default
VLAN ID                0
```

VMready Configuration Example

This example has the following characteristics:

- A VMware vCenter is fully installed and configured prior to VMready configura- tion and includes a “bladevm” administration account and a valid SSL certifi- cate.
- 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 internal switch ports INT1A and INT2A, and one static trunk (previously configured) that includes external ports EXT2 and EXT2.

1. Enable the VMready feature.

```
Router(config)# virt enable
```

2. Specify the VMware vCenter IPv4 address.

```
Router(config)# virt vmware vmware vcspec 172.16.100.1 bladevm
```

When prompted, enter the user password that the switch must use for access to the vCenter.

3. Create the VM profile.

```
Router(config)# virt vmprofile Finance
Router(config)# virt vmprofile edit Finance vlan 30
Router(config)# virt vmprofile edit Finance shaping 1000 2000 3000
```

4. Define the VM group.

```
Router(config)# virt vmgroup 1 profile Finance
Router(config)# virt vmgroup 1 vm arctic
Router(config)# virt vmgroup 1 vm monster
Router(config)# virt vmgroup 1 vm sierra
Router(config)# virt vmgroup 1 vm 00:50:56:4f:f2:00
Router(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 and .

5. If necessary, enable VLAN tagging for the VM group:

```
Router(config)# virt vmgroup 1 tag
```

Note: If the VM group contains ports which also exist in other VM groups, tagging should be enabled in both VM groups. In this example configuration, no ports exist in more than VM group.

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.

Networking OS supports a maximum of 20 static multicast ARP entries.
Note: If you use the ACL profile or IPMC-OPT profile, an ACL entry is consumed for each Static Multicast ARP entry that you configure. Hence, you can configure a maximum of 640 ACL and multicast MAC entries together.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:

```
Router(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:

```
Router(config)# ip arp <destination unicast IP address>  <destination
                    multicast MACaddress>  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.

```
Router(config)# mac-address-table multicast 03:bf:0A:0A:0A:2A 42
54,56
```

2. Configure the static multicast ARP entry:

```
Router(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:

```
Router(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:

```
Router(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.

Switch Partition

Switch Partition (SPAR) enables consolidation of multiple network partitions within an embedded switch. SPARs divide the data plane of a physical switch into independent switching domains. Switch partitions are isolated from each other. Traffic originating in one SPAR stays local to that SPAR. Within a partitioned switch, traffic from one SPAR is never delivered to another SPAR. Traffic from one SPAR can, however, be delivered to another SPAR by traversing an upstream link and switch.

Each individual SPAR requires exactly one uplink, which can be a port, a port channel, or an LACP group. Limiting SPAR connectivity to one external uplink prevents the creation of loops.

SPAR operates as a Layer 2 broadcast network. Hosts on the same VLAN, attached to a SPAR, can communicate with each other and with the upstream switch. Hosts on the same VLAN, but attached to different SPARs, communicate via the upstream switch.

SPAR Processing Modes

SPAR operates in two processing modes. The default mode is pass-through domain.

- Local Domain: In local-domain processing mode, VLAN classification and assignment is based on the user-defined VLAN.
- Pass-through Domain: In pass-through domain processing mode, VLAN classification and assignment is based on the outer tag, which contains the unique domain VLAN ID of the SPAR. The inner tag with the user-defined VLAN remains unchanged.

Local Domain Processing

Each SPAR on a switch has a unique VLAN ID, which separates data between SPARs. If multiple networks share the uplink, the upstream switch port must be configured as a 802.1Q trunk port so it can process multiple VLAN traffic from a SPAR. The SPAR domain uses a single uplink port or LAG shared among all the VLANs. For link redundancy or greater bandwidth, the uplinks can be grouped as static or LACP LAG.

If a VLAN is defined on multiple SPARs, the egress port mask is used to prevent communication between the SPARs in the same local domain VLAN. Since port membership of each SPAR is unique, the egress port mask ensures that different SPAR ports in the same local domain VLAN do not communicate with each other.

In local domain processing, all SPAR ports must have the following settings:

- Tagging/Trunk mode must be enabled.
- Ingress VLAN tagging is disabled on all SPAR ports.
- PVID/Native VLAN is based on any VLAN defined in SPAR.

Pass-Through Domain Processing

Pass-through domain processing is the default operating mode for SPAR when performing L2 switching based on an outer tag.

In pass-through processing mode, each SPAR is identified by its unique VLAN domain ID. Packets are classified based on the SPAR domain ID (outer tag). SPAR ports must be configured in tunnel mode.

SPAR provides single or multiple VLAN connectivity through a single uplink port or LAG (static or LACP). VLAN definition within the SPAR domain is not required.

Pass-through domain operates in Q-In-Q mode. Inside SPAR, different user-defined VLAN traffic is classified into single S-VLAN (service VLAN) associated with the SPAR.

Although the uplink can be shared by multiple networks using the pass-through domain, SPAR will not be server-VLAN aware. Hence, multiple VLAN traffic will be mixed together in a single broadcast domain, that is, broadcast traffic on different VLANs from the upstream network will reach all servers attached to the SPAR pass-through domain. The servers drop the packets if they do not belong to the desired VLAN. The pass-through implementation uses ingress VLAN tagging, that is, tagpvid-ingress is enabled on all SPAR ports.

In pass-through domain processing mode, all SPAR ports must have the following settings:

- PVID/Native VLAN tagging is disabled.
- Ingress VLAN tagging is enabled on all SPAR ports.
- PVID/Native VLAN is based on the SPAR DVLAN.

Unsupported Features

The following features are not supported when SPAR is configured:

- 802.1x
- Edge Virtual Bridging
- Hotlinks
- IGMP
- Layer 3 Configuration
- Management VLAN
- Private VLAN
- Protocol VLAN
- sFlow
- Stacking
- STP, RSTP, MRSTP, PVST
- UFP
- vLAG
- VMAP
- VMready
- VNIC

Limitations

The following limitations apply:

- UFP and SPAR cannot be configured together.
- Trunks must first be configured for SPAR before they can be used. Static or Link Aggregation Control Protocol (LACP) trunks created on the global switch cannot reference any SPAR ports. Use the commands in the following menus to define trunks in the SPAR context:

```
Router(config)# spar <num>
Router(config-spar)# uplink ?

       adminkey      Set lacp trunk for uplink
       port          Set external port for uplink
       PortChannel   Set portchannel for uplink
```

```
Router(config)# portchannel ?

       <1-64> PortChannel group
       <65-128>  LACP PortChannel group
       thash  Port Channel hash configuration
```

- ACLs defined on the global switch can be used for SPAR ports. However, the following restrictions apply:
  – An ACL cannot be shared across SPAR ports if:
  • An exit port (`Router(config)# access-control list <number> egress-port port <number>`) is used as a filtering criteria and the exit port does not belong to the same SPAR as the port on which the ACL is applied.
  • A monitor port is used as a filtering criteria, and the monitor port does not belong to the same SPAR as the mirrored port and is not defined on the global switch.
  – The above ACL restrictions apply to all ACLs defined in an ACL group.
- Port mirroring can be configured on SPAR ports with the following restrictions:
  – The monitor port must belong to the same SPAR as the mirrored port, or must be defined on the global switch.
- Layer 2 failover features can be configured on SPAR ports. However, the Layer 2 failover Auto Monitor (AMON) option is not supported. Only the Layer 2 failover Manual Monitor (MMON) option can be used when all ports defined within the trigger belong to the same SPAR.

SPAR VLAN Management

SPAR VLANs use the same 1024 VLAN space available for other applications/features on the switch. The VLAN ID can be in the range of 2 - 4094. VLAN 1 and the management VLAN 4095 are reserved for the global switch context.

A VLAN assigned to a SPAR cannot be used for any other switch application. Similarly, VLAN used by any other switch application cannot be assigned to a SPAR.

SPAR member ports cannot be members of any other VLAN.

Example Configurations

The following are examples of SPAR pass through and local domain configurations.

Pass Through Configuration

This example describes configuration of SPAR 1 in pass-through mode with internal server ports INTA5 through INTA10, with a single port, EXT1.

1. Create SPAR 1.

```
Router(config)# spar 1
```

   Each SPAR is identified with a number that ranges 1 though 8.

2. Add a single uplink port to SPAR 1.

```
Router(config-spar)# uplink port EXT1
```

3. Set the mode of the SPAR to passthrough

```
Router(config-spar)# domain mode passthrough
```

4. Configure SPAR VLAN to 4081.

```
Router(config-spar)# domain default vlan 4081
```

5. Add ports INTA5 through INTA10 to SPAR 1.

```
Router(config-spar)# domain default member INTA5-INTA10
```

6. Enable SPAR 1.

```
Router(config-spar)# enable
```

Local Domain Configuration

This example demonstrates how to create a SPAR in local-domain mode consisting of internal server ports INTA11-INTA14 and a single uplink port, EXT 2.

1. Create SPAR 2.

```
Router(config)# spar 2
```

2. Add uplink port EXT 2 to SPAR 2.

```
Router(config-spar)# uplink port EXT2
```

3. Set the SPAR to local domain mode.

```
Router(config-spar)# domain mode local
```

4. Configure SPAR VLAN to 4082.

```
Router(config-spar)# domain default vlan 4082
```

5. Add server ports INTA11 through INTA14.

```
Router(config-spar)# domain default member INTA11-INTA14
```

6. Configure the VLANs for SPAR 2.

   Each SPAR has a set of local domains numbered 1 through 32, each of which identifies an allowed VLAN.

   The following steps create three local domains: VLAN, 10, 20, and 30

7. Create local domain 1, assign VLAN 10, and specify the SPAR ports that are members of the thatVLAN.

```
Router(config-spar)# domain local 1 vlan 10
Router(config-spar)# domain local 1 member INTA11-INTA14
Router(config-spar)# domain local 1 enable
```

8. Create local domain 2, assign VLAN 20, and specify the SPAR ports that are members of the that VLAN.

```
Router(config-spar)# domain local 2 vlan 20
Router(config-spar)# domain local 2 member INTA11-INTA14
Router(config-spar)# domain local 2 enable
```

9. Create local domain 3, assign VLAN 30, and specify the SPAR ports that are members of the that VLAN.

```
Router(config-spar)# domain local 3 vlan 30
Router(config-spar)# domain local 3 member INTA11-INTA14
Router(config-spar)# domain local 3 enable
```

10. Enable SPAR 2.

```
Router(config-spar)# enable
```

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 IP Routing
- Internet Protocol Version 6
- Using IPsec with IPv6
- Routing Information Protocol
- Internet Group Management Protocol
- Multicast Listener Discovery
- Border Gateway Protocol
- OSPF
- Protocol Independent Multicast

Basic IP Routing

This chapter provides configuration background and examples for using 1/10Gb LAN Switch Module to perform IP routing functions. The following topics are addressed in this chapter:

- "IP Routing Benefits" on page 5-2
- "Routing Between IP Subnets" on page 5-2
- "Subnet Routing Example" on page 5-3
- "Dynamic Host Configuration Protocol" on page 5-10

IP Routing Benefits

1/10Gb LAN Switch Module 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.1/10Gb LAN Switch Module is intelligent and fast enough to perform routing functions on par with wire-speed Layer 2 switching.

The combination of faster routing and switching in a single device provides another service—it allows you to build versatile topologies that account for legacy configurations.

Consider an example in which a corporate campus 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:

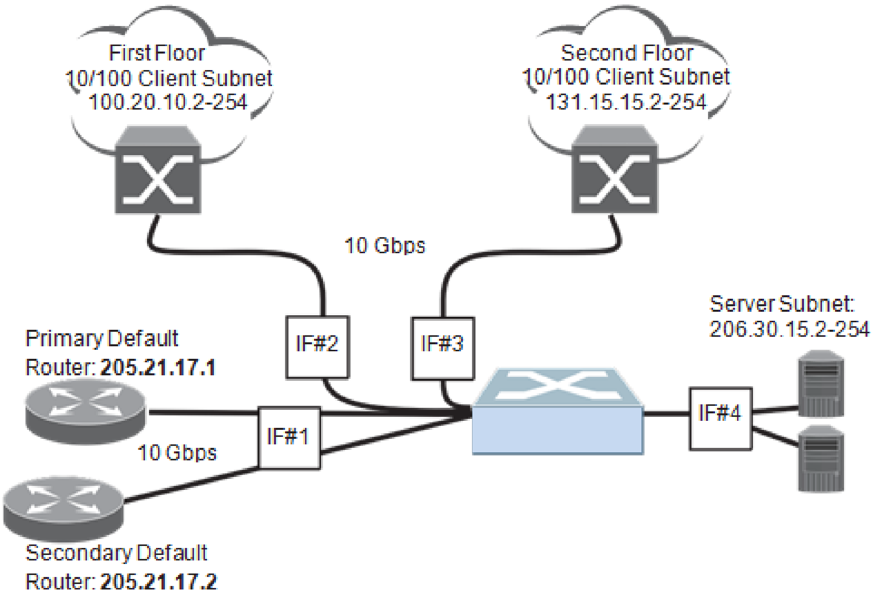
- Routers can be slower than switches. The cross-subnet side trip from the switch to the router and back again adds two hops for the data, slowing throughput considerably.
- Traffic to the router increases, increasing congestion.

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 1/10Gb LAN Switch Modules with built-in IP routing capabilities. Cross-subnet LAN traffic can now be routed within the switches with wire speed Layer 2 switching performance. This not only eases the load on the router but saves the network administrators from reconfiguring each and every end-station with new IP addresses.

Take a closer look at 1/10Gb LAN Switch Module in the following configuration example:

Figure 18. Switch-Based Routing Topology



1/10Gb LAN Switch Module 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 1/10Gb LAN Switch Module, 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.

Subnet Routing Example

Prior to configuring, you must be connected to the switch Command Line Interface (CLI) as the administrator.
Note: For details about accessing and using any of the menu commands described in this example, see the Networking OS Command Reference.

1. Assign an IP address (or document the existing one) for each router and client workstation.
In the example topology in Figure 18 on page 5-3, the following IP addresses are used:

Table 20. Subnet Routing Example: IP Address Assignments

Subnet	Devices	IP Addresses
1	Primary and Secondary Default Routers	205.21.17.1 and 205.21.17.2
2	First Floor Client Workstations	100.20.10.2-254
3	Second Floor Client Workstations	131.15.15.2-254
4	Common 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 21. Subnet Routing Example: IP Interface Assignments

Interface	Devices	IP Interface Address
IF 1	Primary and Secondary Default Routers	205.21.17.3
IF 2	First Floor Client Workstations	100.20.10.1
IF 3	Second Floor Client Workstations	131.15.15.1
IF 4	Common Servers	206.30.15.1

IP interfaces are configured using the following commands:

```
Router(config)# interface ip 1                  (Select IP interface 1)
Router(config-ip-if)# ip address 205.21.17.3 255.255.255.0 enable
Router(config-vlan)# exit
Router(config)# interface ip 2                 (Select IP interface 2)
Router(config-ip-if)# ip address 100.20.10.1 255.255.255.0 enable
Router(config-ip-if)# exit
Router(config)# interface ip 3                 (Select IP interface 3)
Router(config-ip-if)# ip address 131.15.15.1 255.255.255.0 enable
Router(config-ip-if)# exit
Router(config)# interface ip 4                 (Select IP interface 4)
Router(config-ip-if)# ip address 206.30.15.1 255.255.255.0 enable
Router(config-ip-if)# exit
```

3. Set each server and workstation’s default gateway to the appropriate switch IP interface (the one in the same subnet as the server or workstation).

4. Configure the default gateways to the routers’ addresses.
Configuring the default gateways allows the switch to send outbound traffic to the routers:

```
Router(config)# ip gateway 1 address 205.21.17.1 enable

Router(config)# ip gateway 2 address 205.21.17.2 enable
```

5. Verify the configuration.

```
Router(config)# show interface ip
```

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

Using VLANs to Segregate Broadcast Domains

In the previous example, devices that share a common IP network are all in the same broadcast domain. If you want to limit the broadcasts on your network, you could use VLANs to create distinct broadcast domains. For example, as shown in the following procedure, you could create one VLAN for the client trunks, one for the routers, and one for the servers.

In this example, you are adding to the previous configuration.

1. Determine which switch ports and IP interfaces belong to which VLANs.

The following table adds port and VLAN information:

Table 22. Subnet Routing Example: Optional VLAN Ports

VLAN	Devices	IP Interface	Switch Port	VLAN #
1	First Floor Client Workstations	2	EXT1	1
	Second Floor Client Workstations	3	EXT2	1
2	Primary Default Router	1	EXT3	2
	Secondary Default Router	1	EXT4	2
3	Common Servers 1	4	INT5A	3
	Common Servers 2	4	INT6A	3

2. Add the switch ports to their respective VLANs.

The VLANs shown in Table 22 are configured as follows:

```
Router(config)# vlan 1
Router(config-vlan)# exit
Router(config)# interface port ext1,ext2       (Add ports to VLAN 1)
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk allowed vlan add 1
Router(config-if)# exit

Router(config)# vlan 2
Router(config-vlan)# exit
Router(config)# interface port ext3,ext4       (Add ports to VLAN 2)
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk allowed vlan add 2
Router(config-if)# exit

Router(config)# vlan 3
Router(config-vlan)# exit
Router(config)# interface port inet5a,int6a    (Add ports to VLAN 3)
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk allowed vlan add 3
Router(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 current PVID is 1. Confirm changing PVID
from 1 to 2 [y/n]?
```

Enter `y` to set the default Port VLAN ID (PVID) for the port.

3. Add each IP interface to the appropriate VLAN.

Now that the ports are separated into three VLANs, the IP interface for each subnet must be placed in the appropriate VLAN. From Table 22, the settings are made as follows:

```
Router(config)# interface ip 1    (Select IP interface 1)
Router(config-ip-if)# vlan 2      (Add VLAN 2)
Router(config-vlan)# exit
Router(config)# interface ip 2    (Select IP interface 2)
Router(config-ip-if)# vlan 1      (Add VLAN 1)
Router(config-ip-if)# exit
Router(config)# interface ip 3    (Select IP interface 3)
Router(config-ip-if)# vlan 1      (Add VLAN 1)
Router(config-ip-if)# exit
Router(config)# interface ip 4    (Select IP interface 4)
Router(config-ip-if)# vlan 3      (Add VLAN 3)
Router(config-ip-if)# exit
```

4. Verify the configuration.

```
Router(config)# show vlan
Router(config)# show interface information
Router(config)# show interface ip
```

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

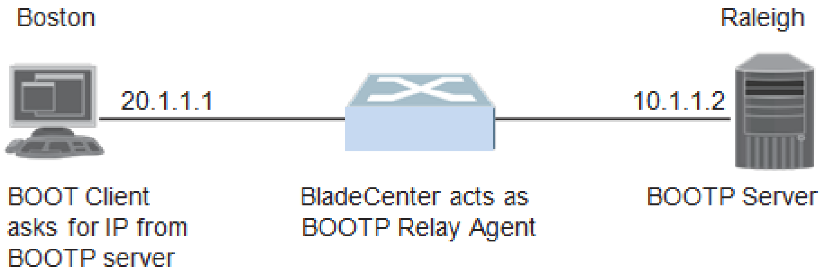
BOOTP Relay Agent

1/10Gb LAN Switch Module can function as a Bootstrap Protocol relay agent, enabling the switch to forward a client request for an IP address up to two BOOTP servers with IP addresses that have been configured on the switch.

When a switch receives a BOOTP request from a BOOTP client requesting an IP address, the switch acts as a proxy for the client. The request is then forwarded as a UDP Unicast MAC layer message to two BOOTP servers whose IP addresses are configured on the switch. The servers respond to the switch with a Unicast reply that contains the default gateway and IP address for the client. The switch then forwards this reply back to the client.

Figure 19 shows a basic BOOTP network example.

Figure 19. BOOTP Relay Agent Configuration


Boston
Raleigh
20.1.1.1
10.1.1.2
BOOT Client asks for IP from BOOTP server
BladeCenter acts as BOOTP Relay Agent
BOOTP Server

BOOTP Relay Agent Configuration

To enable 1/10Gb LAN Switch Module to be the BOOTP forwarder, you need to configure the BOOTP server IP addresses on the switch, and enable BOOTP relay on the interface(s) on which the BOOTP requests are received.

Generally, you should configure the command on the switch IP interface that is closest to the client, so that the BOOTP server knows from which IP subnet the newly allocated IP address should come.

Use the following commands to configure the switch as a BOOTP relay agent:

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

Use the following command to enable the Relay functionality on an IP interface:

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

Domain-Specific BOOTP Relay Agent Configuration

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

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

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.

DHCP is an extension of another network IP management protocol, Bootstrap Protocol (BOOTP), with an additional capability of being able to dynamically 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).

DHCP relay agent eliminates the need to have DHCP/BOOTP servers on every subnet. It allows the administrator to reduce the number of DHCP servers deployed on the network and to centralize them. Without the DHCP relay agent, there must be at least one DHCP server deployed at each subnet that has hosts needing to perform the DHCP request.

DHCP Relay Agent

DHCP is described in RFC 2131, and the DHCP relay agent supported on 1/10Gb LAN Switch Modules 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.

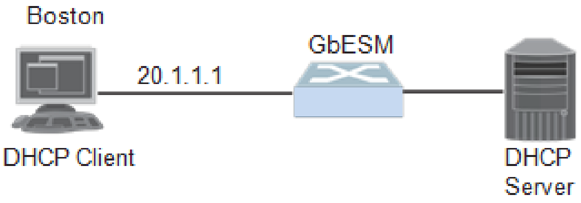
In the DHCP environment, 1/10Gb LAN Switch Module acts as a relay agent. The DHCP relay feature enables the switch to forward a client request for an IP address to two BOOTP servers with IP addresses that have been configured on the switch.

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 two 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.

DHCP Relay Agent Configuration

To enable 1/10Gb LAN Switch Module to be the BOOTP forwarder, you need to configure the DHCP/BOOTP server IP addresses on the switch. Generally, you should 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 should come.

The following figure shows a basic DHCP network example:
Figure 20. DHCP Relay Agent Configuration



In 1/10Gb LAN Switch Module 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 two servers provide failover redundancy. However, no health checking is supported.

Use the following commands to configure the switch as a DHCP relay agent:

```
Router(config)# ip bootp-relay server 1 <IP address>
Router(config)# ip bootp-relay server 2 <IP address>
Router(config)# ip bootp-relay server 3 <IP address>
Router(config)# ip bootp-relay server 4 <IP address>
Router(config)# ip bootp-relay server 5 <IP address>
Router(config)# ip bootp-relay enable
Router(config)# show ip bootp-relay
```

Additionally, DHCP Relay functionality can be assigned on a per interface basis. Use the following command to enable the Relay functionality:

```
Router(config)# interface ip <Interface number>
Router(config-ip-if)# relay
```

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 2461
- RFC 2462
- RFC 2474
- RFC 2526
- RFC 2711

- RFC 1981
- RFC 2404
- RFC 2410
- RFC 2451
- RFC 2460
- RFC 2461
- RFC 2462
- RFC 2474
- RFC 2526
- RFC 2711

- RFC 1981
- RFC 2404
- RFC 2410
- RFC 2451
- RFC 2460
- RFC 2461
- RFC 2462
- RFC 2474
- RFC 2526
- RFC 2711

- RFC 1981
- RFC 2404
- RFC 2410
- RFC 2451
- RFC 2460
- RFC 2461
- RFC 2462
- RFC 2474
- RFC 2526
- RFC 2711

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 Networking OS 7.8 features permit IP addresses to be configured using either IPv4 or IPv6 address formats. However, the following switch features support IPv4 only:

- Default switch management IP address
- Bootstrap Protocol (BOOTP) and DHCP
- RADIUS, TACACS+ and LDAP
- VMware Virtual Center (vCenter) for VMready
- Routing Information Protocol (RIP)
- Internet Group Management Protocol (IGMP)
- Border Gateway Protocol (BGP)
- 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 7.8 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.

Networking OS 7.8 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. By default, the switch automatically configures the IPv6 address of its management interface.

You can manually configure two IPv6 addresses for each interface, as follows:

- Initial IPv6 address is a global unicast or anycast address.

  ```
  Router(config)# interface ip <interface number>
  Router(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 .

  ```
  Router(config-ip-if)# ipv6 secaddr6 <IPv6 address>
  Router(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.

Interface 127 is reserved for IPv6 host support. This interface is included in management VLAN 4095. Use the following commands to configure the IPv6 gateway:

```
Router(config)# ip gateway6 1 address <IPv6 address>
Router(config)# ip gateway6 1 enable
```

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 various advertisements, flags, and interval settings is performed on a per-interface basis using the following command path:

```
Router(config)# interface ip <interface number>
Router(config-ip-if)# [no] ipv6 nd ?
Router(config-ip-if)# exit
```

To add or remove entries in the static neighbor cache, use the following command path:

```
Router(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:

```
Router(config)# interface ip <interface number>
Router(config-ip-if)# ip6host
Router(config-ip-if)# exit
```

By default, host mode is enabled on the management interface, and disabled on data interfaces.

1/10Gb LAN Switch Module supports up to 1156 IPv6 routes.

Supported Applications

The following applications have been enhanced to provide IPv6 support.

- **Ping**
  The pingcommand supports IPv6 addresses. Use the following format to ping an IPv6 address:

```
ping <host name>|<IPv6 address> [-n <tries (0-4294967295)>]
[-w <msec delay (0-4294967295)>] [-l <length (0/32-65500/2080)>]
[-s <IP source>] [-v <TOS (0-255)>] [-f] [-t]
```

  To ping a link-local address (begins with FE80), provide an interface index, as follows:

```
ping <IPv6 address>%<Interface index> [-n <tries (0-
4294967295)>][-w <msec delay (0-4294967295)>] [-l <length
(0/32-65500/2080)>][-s <IP source>] [-v <TOS (0-255)>] [-f]
[-t]
```

- **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:

  ```
  Router(config)# ip dns ipv6 request-version {ipv4|ipv6}
  ```

  If you set the request version to v4, 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 `v6`, 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.
- Interface 127 are reserved for IPv6 management.
- 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.

```
Router(config)# interface ip 2
Router(config-ip-if)# ip6host
Router(config-ip-if)# enable
Router(config-ip-if)# exit
```

2. Configure the IPv6 default gateway.

```
Router(config)# ip gateway6 1 address 2001:BA98:7654:BA98:FEDC:1234:ABCD:5412
Router(config)# ip gateway6 1 enable
```

3. Verify the interface address.

```
Router(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.

```
Router(config)# interface ip 3
Router(config-ip-if)# ipv6 address 2001:BA98:7654:BA98:FEDC:1234:ABCD:5214
Router(config-ip-if)# ipv6 prefixlen 64
Router(config-ip-if)# ipv6 seccaddr6 2003::1 32
Router(config-ip-if)# vlan 2
Router(config-ip-if)# enable
Router(config-ip-if)# exit
```

   The secondary IPv6 address is compressed, and the prefix length is 32.

2. Configure the IPv6 default gateway.

```
Router(config)# ip gateway6 1 address 2001:BA98:7654:BA98:FEDC:1234:ABCD:5412
Router(config)# ip gateway6 1 enable
```

3. Configure Router advertisements for the interface (optional)

```
Router(config)# interface ip 3
Router(config-ip-if)# no ipv6 nd suppress-ra
```

4.

```
Router(config-ip-if)# show layer3
```

Using 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, Networking OS 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 Pay- load (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
- 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 5-22
- "Using IPsec with 1/10Gb LAN Switch Module" on page 5-23

IPsec Protocols

The Networking OS implementation of IPsec supports the following protocols:

- Authentication Header (AH)
  AHs provide connectionless integrity outand 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 1/10Gb LAN Switch Module

IPsec supports the fragmentation and reassembly of IP packets that occurs when data goes to and comes from an external device. The Hitachi BladeSymphony 1/10Gb LAN Switch Module 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.

```
Router(config)# ikev2 proposal
```

2. Set the DES encryption algorithm.

```
Router(config-ikev2-prop)# encryption 3des|aes-cbc|des    (default: 3des)
```

3. Set the authentication integrity algorithm type.

```
Router(config-ikev2-prop)# integrity md5|sha1      (default: sha1)
```

4. Set the Diffie-Hellman group.

```
Router(config-ikev2-prop)# group 1|2|5|14|24 (default: 2)
```

Importing an IKEv2 Digital Certificate

To import an IKEv2 digital certificate for authentication:

1. Import the CA certificate file.

```
Router(config)# copy tftp ca-cert address <hostname or IPv4 address>
```

Source file name: *<path and filename of CA certificate file>*
Confirm download operation [y/n]: *y*

2. Import the host key file.

```
Router(config)# copy tftp host-key address <hostname or IPv4 address>
```

```
Source file name: <path and filename of host private key file>
Confirm download operation [y/n]: y
```

3. Import the host certificate file.

```
Router(config)# copy tftp host-cert address <hostname or IPv4 address>

Source file name: <path and filename of host certificate file>
Confirm download operation [y/n]: y
```

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 `mgt`, 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.

```
Router(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
generat‘eywing 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:

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

3. Enable IKEv2 RSA-signature authentication:

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

Enabling IKEv2 Preshared Key Authentication

To set up IKEv2 preshared key authentication:

1. Enter the local preshared key.

```
Router(config)# ikev2 preshare-key local <preshared key, a string of 1-256 chars>
```

2. If asymmetric authentication is supported, enter the remote key:

```
Router(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:

```
Router(config)# ikev2 identity local address (use an IPv6 address)
Router(config)# ikev2 identity local email <email address>
Router(config)# ikev2 identity local fqdn <domain name>
```

To disable IKEv2 RSA-signature authentication method and enable preshared key authentication, enter:

```
Router(config)# no access https
```

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-des` | `esp-3des` | `esp-aes-cbc` | `esp-null`
– *integrity algorithm*	One of the following: `esp-sha1` | `esp-md5` | `none`
– *AH authentication algorithm*	One of the following: `ah-sha1` | `ah-md5` | `none`

2. Decide whether to use tunnel or transport mode. The default mode is transport.

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

- `protoany`	apply the selector to any type of traffic
- `proto/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
- `proto/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).

3. 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.

```
Router(config)# ipsec manual-policy <policy number>
```

2. Configure the policy.

```
Router(config-ipsec-manual)#peer <peer's IPv6 address>
Router(config-ipsec-manual)#traffic-selector <IPsec traffic selector>
Router(config-ipsec-manual)#transform-set <IPsec transform set>
Router(config-ipsec-manual)#in-ah auth-key <inbound AH IPsec key>
Router(config-ipsec-manual)#in-ah auth-spi <inbound AH IPsec SPI>
Router(config-ipsec-manual)#in-esp cipher-key <inbound ESP cipher key>
Router(config-ipsec-manual)#in-esp auth-spi <inbound ESP SPI>
Router(config-ipsec-manual)#in-esp auth-key <inbound ESP authenticator key>
Router(config-ipsec-manual)#out-ah auth-key <outbound AH IPsec key>
Router(config-ipsec-manual)#out-ah auth-spi <outbound AH IPsec SPI>
Router(config-ipsec-manual)#out-esp cipher-key <outbound ESP cipher key>
Router(config-ipsec-manual)#out-esp auth-spi <outbound ESP SPI>
Router(config-ipsec-manual)#out-esp auth-key <outbound ESP authenticator key>
```

where the following parameters are used:

- *peer's IPv6 address* The IPv6 address of the peer (for example, 3000::1)
- *IPsec traffic-selector* *A number from 1-10*
- *IPsec of transform-set* A number from1-10
- *inbound AH IPsec key* The inbound AH key code, in hexadecimal
- *inbound AH IPsec SPI* A number from 256-4294967295
- *inbound ESP cipher key* The inbound ESP key code, in hexadecimal
- *inbound ESP SPI* A number from 256-4294967295
- *inbound ESP authenticator key* The inbound ESP authenticator key code, in hexadecimal
- *outbound AH IPsec key* The outbound AH key code, in hexadecimal
- *outbound AH IPsec SPI* A number from 256-4294967295
- *outbound ESP cipher key* The outbound ESP key code, in hexadecimal
- *outbound ESP SPI* A number from 256-4294967295
- *outbound ESP authenticator key* The outbound ESP authenticator key code, in hexadecimal

Note: When configuring a manual policy ESP, the ESP authenticator key is optional.

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.

```
Router(config)# ipsec dynamic-policy <policy number>
```

2. Configure the policy.

```
Router(config-ipsec-dynamic)# peer <peer’s IPv6 address>
Router(config-ipsec-dynamic)# traffic-selector <index of traffic selector>
Router(config-ipsec-dynamic)# transform-set <index of transform set>
Router(config-ipsec-dynamic)# sa-lifetime <SA lifetime, in seconds>
Router(config-ipsec-dynamic)# pfs enable|disable
```

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 from1-10
- *index of transform-set* A number from1-10
- *SA lifetime, in seconds* The length of time the SA is to remain in effect; an integer from120-86400
- **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:

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). Networking OS software supports RIP version 1 (RIPv1) and RIP version 2 (RIPv2) for exchanging TCP/IPv4 route information with other routers.

Note: Networking OS 7.8 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 cost 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 limiting 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 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 Menu, Routing Information Protocol Configuration in the *Networking OS Command Reference*.

RIPv1

RIP version 1 uses 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. Networking OS supports using clear password for RIPv2.

RIPv2 in RIPv1 Compatibility Mode

Networking OS allows you to configure RIPv2 in RIPv1compatibility 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

Networking OS provides the following features to support RIPv1 and RIPv2:

Poison Reverse

Simple split horizon in RIP omits routes learned from one neighbor in updates sent to that neighbor. That is the most common configuration used in RIP, with the Poison Reverse feature disabled. Split horizon with poisoned reverse enabled 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 updates. Multicast RIPv2 updates 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 Route

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 plain text 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 a 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 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 (`/cfg/l3/rip/if <x>/auth/password`); otherwise, the routing information from authenticated messages is propagated by RIPv1 routers in an unauthenticated manner.

RIP Configuration Example

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.

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

Router(config)# vlan 3
Router(config-vlan)# exit
Router(config)# interface port 3
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk allowed vlan add 3
Router(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.

```
Router(config)# interface ip 2
Router(config-ip-if)# enable
Router(config-ip-if)# ip address 102.1.1.1
Router(config-ip-if)# vlan 2
Router(config-ip-if)# exit
Router(config)# interface ip 3
Router(config-ip-if)# enable
Router(config-ip-if)# ip address 103.1.1.1
Router(config-ip-if)# vlan 3
Router(config-ip-if)# exit
```

3. Turn on RIP globally and enable RIP for each interface.

```
Router(config)# router rip
Router(config-router-rip)# enable
Router(config-router-rip)# exit
Router# interface ip 2
Router(config-ip-if)# ip rip enable
Router(config-ip-if)# exit
Router# interface ip 3
Router(config-ip-if)# ip rip enable
Router(config-ip-if)# exit
```

Use the following command to check the current valid routes in the routing table of the switch:

```
Router# show ip route
```

For those RIP learnt routes within the garbage collection period, that are routes phasing out of the routing table with metric 16, use the following command:

```
Router# show ip rip routes
```

Locally configured static routes do not appear in the RIP Routes table.

Internet Group Management Protocol

Internet Group Management Protocol (IGMP) is used by IPv4 Multicast routers to learn about the existence of host group members on their directly attached subnet (see RFC 2236). The IPv4 Multicast 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.

1/10Gb LAN Switch Module can perform IGMP Snooping, or act as an IGMP Relay (proxy) device.

The following topics are discussed in this chapter:

- "IGMP Snooping" on page 5-36
- "IGMP Querier" on page 5-42
- "Additional IGMP Features" on page 5-43

IGMP Snooping

IGMP Snooping allows the switch to forward multicast traffic only to those ports that request it. IGMP Snooping prevents multicast traffic from being flooded to all ports. The switch learns which server hosts are interested in receiving multicast traffic, and forwards it only to ports connected to those servers.

IGMP Snooping conserves bandwidth. With IGMP Snooping, the switch learns which ports are interested in receiving multicast data, and forwards multicast data only to those ports. In this way, other ports are not burdened with unwanted multicast traffic.

The switch can sense IGMP Membership Reports from attached clients and act as a proxy to set up a dedicated path between the requesting host and a local IPv4 Multicast router. After the pathway 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.

The client-server path is set up as follows:

- An IPv4 Multicast Router (Mrouter) sends Membership Queries to the switch, which forwards them to all ports in a given VLAN.
- Hosts that want to receive the multicast data stream send Membership Reports to the switch, which sends a proxy Membership Report to the Mrouter.
- The switch sets up a path between the Mrouter and the host, and blocks all other ports from receiving the multicast.
- Periodically, the Mrouter sends Membership Queries to ensure that the host wants to continue receiving the multicast. If a host fails to respond with a Membership Report, the Mrouter stops sending the multicast to that path.
- The host can send an IGMP Leave packet to the switch, which responds with an IGMP Groups Specific Query in order to check if there are other clients that want to receive the multicast traffic for the group referenced in the Leave packet. If an IGMP Report is not received, the group is deleted from the port and the multicast path is terminated. The switch then sends a Proxy Leave packet to the Mrouter in order to update it. If the FastLeave option is enabled on a VLAN, the multicast path is terminated immediately and the Leave packet is directly forwarded to the Mrouter.

IGMP Groups

1/10Gb LAN Switch Module supports a maximum of 4096 IGMP entries, on a maximum of 1024 VLANs. One IGMP entry is allocated for each unique join request, based on the VLAN and IGMP group address only (regardless of the port). If multiple ports join the same IGMP group using the same VLAN, only a single IGMP entry is used.

IGMPv3

IGMPv3 includes new membership report messages to extend IGMP functionality. 1/10Gb LAN Switch Module provides snooping capability for all types of IGMP version 3 (IGMPv3) Membership Reports, as described in RFC 3376.

IGMPv3 supports Source-Specific Multicast (SSM). SSM identifies session traffic by both source and group addresses. 1/10Gb LAN Switch Module uses *source filtering*, which allows hosts to report interest in receiving multicast packets only from specific source addresses, or from all but specific source addresses.

1/10Gb LAN Switch Module 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:

```
Router(config)# no ip igmp snoop igmpv3 exclude
```

By default, 1/10Gb LAN Switch Module snoops the first eight sources listed in the IGMPv3 Group Record. Use the following command to change the number of snooping sources:

```
Router(config)# ip igmp snoop igmpv3 sources <1-64>
```

IGMPv3 Snooping is compatible with IGMPv1 and IGMPv2 Snooping. You can disable snooping on version 1 and version 2 reports, using the following command:

```
Router(config)# no ip igmp snoop igmpv3 v1v2
```

IGMP Snooping Configuration Example

This section provides steps to configure IGMP Snooping on 1/10Gb LAN Switch Module, using the Command-Line Interface (CLI).

1. Configure port and VLAN membership on the switch.
2. Add VLANs to IGMP Snooping and enable IGMP Snooping.

```
Router(config)# ip igmp snoop vlan 1
Router(config)# ip igmp snoop enable
```

3. Enable IGMPv3 Snooping (optional).

```
Router(config)# ip igmp snoop igmpv3 enable
```

4. Enable IGMP.

```
Router(config)# ip igmp enable       (Turn on IGMP)
```

5. View dynamic IGMP information.
   To display information about IGMP Groups:

```
Router# show ip igmp groups

Total entries: 5 Total IGMP groups: 2
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     EXC    2:26     No
    *              235.0.0.1       9      1       V3     EXC      -     Yes
```

To display information about Mrouters learned by the switch:

```
Router# show ip igmp mrouter

Total entries: 3 Total number of dynamic mrouters: 2
Total number of installed static mrouters : 1
SrcIP                 VLAN   Port     Version   Expires   MRT    QRV  QQIC
-------------------- ------- ------- --------- -------- ------- ---- ----
  10.1.1.1              2     EXT18      V3      4:09       128    2   125
  10.1.1.5              2     EXT19      V2      4:09       125    -    -
  *                     9     EXT10      V2      Static      -     -    -
```

Note: If IGMP Snooping v1/v2 is enabled and IGMPv3 Snooping is disabled, the output of IGMPv3 reports and queries show some items as IGMPv3 (`V3`), though they retain v2 behavior. For example, the `Source` IPv4 address is not relevant for v2 entries.

Static Multicast Router

A static multicast router (Mrouter) can be configured for a particular port on a particular VLAN. A static Mrouter does not have to be learned through IGMP Snooping.

A total of 128 static Mrouters can be configured on 1/10Gb LAN Switch Module. Both internal and external ports can accept a static Mrouter.

Note: When static Mrouters are used, the switch will continue learning dynamic Mrouters via IGMP snooping. However, dynamic Mrouters may not replace static Mrouters. If a dynamic Mrouter has the same port and VLAN combination as a static Mrouter, the dynamic Mrouter will not be learned.

Following is an example of configuring a static multicast router:

1. For each Mrouter, configure a port, VLAN, and IGMP version of the multicast router.

```
Router(config)# ip igmp mrouter EXT5 1 2
```

2. Verify the configuration.

```
Router(config)# show ip igmp mrouter
```

IGMP Relay

1/10Gb LAN Switch Module 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 1/10Gb LAN Switch Module 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 1/10Gb LAN Switch Module, IGMP Relay appears to be an IGMP multicast router (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 a multicast router, 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 EN4093 uses ICMP health checks to determine if the primary and backup mrouters are reachable.

Configuration Guidelines

Consider the following guidelines when you configure IGMP Relay:

- IGMP Relay and IGMP Snooping/Querier are mutually exclusive—if you enable IGMP Relay, you must turn off IGMP Snooping/Querier.
- Add the upstream VLANs to the IGMP Relay list, using the following command:

```
Router(config)# ip igmp relay vlan <VLAN number>
```

- If IGMP hosts reside on different VLANs, you must:
  – Disable IGMP flooding.

```
Router(config)# vlan <vlan id>
Router(config-vlan)# no flood
```

  – Enable CPU forwarding to ensure that multicast data is forwarded across the VLANs.

```
Router(config)# vlan <vlan id>
Router(config-vlan)# cpu
```

Configure IGMP Relay

Use the following procedure to configure IGMP Relay.

1. Configure IP interfaces with IPv4 addresses, and assign VLANs.

```
Router(config)# interface ip 2
Router(config-ip-if)# ip address 10.10.1.1 255.255.255.0 enable
Router(config-ip-if)# vlan 2
Router(config-ip-if)# exit
Router(config)# interface ip 3
Router(config-ip-if)# ip address 10.10.2.1 255.255.255.0 enable
Router(config-ip-if)# vlan 3
Router(config-ip-if)# exit
```

2. Turn IGMP on.

```
Router(config)# ip igmp enable
```

3. Configure the upstream Mrouters with IPv4 addresses.

```
Router(config)# ip igmp relay mrouter 1 address 100.0.1.2
Router(config)# ip igmp relay mrouter 1 enable
Router(config)# ip igmp relay mrouter 2 address 100.0.2.4
Router(config)# ip igmp relay mrouter 2 enable
```

4. Add VLANs to the downstream network and enable IGMP Relay

```
Router(config)# ip igmp relay vlan 2
Router(config)# ip igmp relay vlan 3
Router(config)# ip igmp relay enable
```

IGMP Querier

IGMP Querier allows the switch to perform the multicast router (Mrouter) role and provide Mrouter discovery when the network or virtual LAN (VLAN) does not have a router.

When the IGMP Querier feature is enabled on a VLAN, the switch participates in the Querier election process and has the possibility to be elected as Querier for the VLAN. The IGMP querier periodically broadcasts IGMP Queries and listens for hosts to respond with IGMP Reports indicating their IGMP group memberships. If multiple Mrouters exist on a given network, the Mrouters elect one as the querier, which performs all periodic membership queries. The election process can be based on IPv4 address or MAC address.

Note: When IGMP Querier is enabled on a VLAN, the switch performs the role of IGMP querier only if it meets the IGMP querier election criteria.

Follow this procedure to configure IGMP Querier.

1. Enable IGMP and configure the source IPv4 address for IGMP Querier on a VLAN.

```
Router(config)# ip igmp enable
Router(config)# ip igmp querier enable
Router(config)# ip igmp querier vlan 2 source-ip 10.10.10.1
```

2. Enable IGMP Querier on the VLAN.

```
Router(config)# ip igmp querier vlan 2 enable
```

3. Configure the querier election type and define the address.

```
Router(config)# ip igmp querier vlan 2 election-type ipv4
```

4. Verify the configuration.

```
Router# show ip igmp querier vlan 2
Current VLAN 2 IGMP querier settings: ON
        querier type: ipv4
        max response time: 100
        querier interval: 125
        Querier robustness: 2
        source IP: 10.10.10.15
        startup count: 2
        startup query interval: 31
        version: v3
```

Additional IGMP Features

The following topics are discussed in this section:

- "FastLeave" on page 5-43
- "IGMP Filtering" on page 5-43

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 it 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, unless a multicast router was learned on the port.

Enable FastLeave only on VLANs that have only one host connected to each physical port.

IGMP Filtering

With IGMP Filtering, you can allow or deny a port to learn certain IGMP/IPMC groups. This allows you to restrict users from receiving certain multicast traffic.

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.

```
Router(config)# ip igmp filtering
```

2. Define an IGMP filter with IPv4 information.

```
Router(config)# ip igmp profile 1 range 224.0.0.0 226.0.0.0
Router(config)# ip igmp profile 1 action deny
Router(config)# ip igmp profile 1 enable
```

3. Assign the IGMP filter to a port.

```
Router(config)# interface port 3
Router(config-if)# ip igmp profile 1
Router(config-if)# ip igmp filtering
```

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 5-46
- "How MLD Works" on page 5-47
- "MLD Capacity and Default Values" on page 5-49
- "Configuring MLD" on page 5-50

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. 1/10Gb LAN Switch Module 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 1/10Gb LAN Switch Module:

- 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.

1/10Gb LAN Switch Module 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 options:

```
Router(config)# vlan <vlan number>
Router(config-vlan)# [no] flood
Router(config-vlan)# [no] cpu
Router(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:

```
Router(config)# interface ip <interface number>
Router(config-ip-if)# ipv6 mld dmrtr enable
```

MLD Capacity and Default Values

Table 23 lists the maximum and minimum values of 1/10Gb LAN Switch Module variables.

Table 23. 1/10Gb LAN Switch Module Capacity Table

Variable	Maximum Value
IPv6 Multicast Entries	256
IPv6 Interfaces for MLD	8

Table 24 lists the default settings for MLD features and variables.

Table 24. 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 + ½ QRI]
Start up Query Interval [SQI]	31.25 seconds [derived: ¼ * 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.

```
Router(config)# ipv6 mld
Router(config-router-mld)# enable
Router(config-router-mld)# exit
```

2. Create an IPv6 interface.

```
Router(config)# interface ip 2
Router(config-ip-if)# enable
Router(config-ip-if)# ipv6 address 2002:1:0:0:0:0:0:3
Router(config-ip-if)# ipv6 prefixlen 64
```

3. Enable MLD on the IPv6 interface.

```
Router(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.

```
Router(config-ip-if)# ipv6 mld version <1-2>(MLD version)
Router(config-ip-if)# ipv6 mld robust <1-10>(Robustness)
Router(config-ip-if)# ipv6 mld qri <1-256>   (In seconds)
Router(config-ip-if)# ipv6 mld qintrval <1-608>(In seconds)
Router(config-ip-if)# ipv6 mld llistnr <1-32>(In seconds)
```

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.

1/10Gb LAN Switch Module can advertise their IP interfaces and IPv4 addresses using BGP and take BGP feeds from as many as BGP router peers. This allows more resilience and flexibility in balancing traffic from the Internet.
Note: Networking OS 7.8 does not support IPv6 for BGP.

The following topics are discussed in this section:

- “Internal Routing Versus External Routing” on page 5-51
- “Forming BGP Peer Routers” on page 5-53
- “What is a Route Map?” on page 5-53
- “Aggregating Routes” on page 5-56
- “Redistributing Routes” on page 5-56
- “BGP Attributes” on page 5-56
- “Selecting Route Paths in BGP” on page 5-58
- “BGP Failover Configuration” on page 5-59
- “Default Redistribution and Route Aggregation Example” on page 5-61

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.

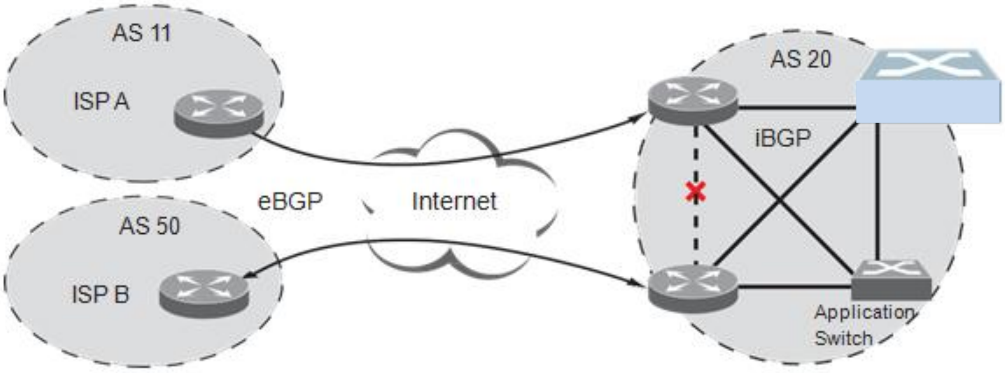
Static routes should have a higher degree of precedence than dynamic routing protocols. If the destination route is not in the route cache, then the packets are forwarded to the default gateway which may be incorrect if a dynamic routing protocol is enabled.

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) in order 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 21), 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 Hitachi BladeSymphony and the Application Switch.

Figure 21. iBGP and eBGP



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.

Forming BGP Peer Routers

Two BGP routers become peers or neighbors once you establish a TCP connection between them. 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.

What is a Route Map?

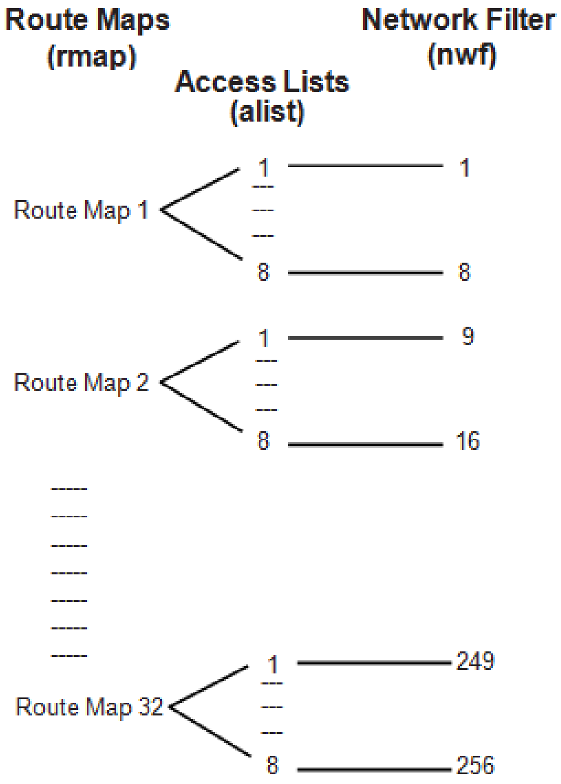
A route map is used to control and modify routing information. Route maps define conditions for redistributing routes from one routing protocol to another or controlling routing information when injecting it in and out of BGP. For example, a route map is used to set a preference value for a specific route from a peer router and another preference value for all other routes learned via the same peer router. For example, the following commands are used to define a route map:

```
Router(config)# route-map <map number>      (Select a route map)
Router(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 5-59.

Networking OS allows you to configure 32 route maps. Each route map can have up to eight 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 22 illustrates the relationship between route maps, access lists and network filters.

Figure 22. Distributing Network Filters in Access Lists and Route Maps



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 commands:

```
Router(config)# route-map <map number>   (Select a route map)
Router(config-route-map)# precedence <1-255>       (Specify a precedence)
Router(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 network filter.

```
Router(config)# ip match-address 1 <IPv4 address>  <IPv4 subnet mask>
Router(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 32 route maps each containing eight 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.

```
Router(config)# route-map 1
Router(config-route-map)# access-list 1 match-address 1
Router(config-route-map)# access-list 1 metric <metric value>
Router(config-route-map)# access-list 1 action deny
Router(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 attributes in the AS filter menu.

```
Router(config-route-map)# as-path-list 1 as 1
Router(config-route-map)# as-path-list 1 action deny
Router(config-route-map)# as-path-list 1 enable
```

4. Set up the BGP attributes.

   If you want to overwrite the attributes that the peer router is sending, then 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.

```
  Router(config-route-map)# as-path-preference <AS number> [<AS number>] [<AS
number>]
  Router(config-route-map)# local-preference <local preference value>
  Router(config-route-map)# metric <metric value>
```

5. Enable the route map.

```
Router(config-route-map)# enable
Router(config-route-map)# exit
```

6. Turn BGP on.

```
Router(config)# router bgp
Router(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,

```
Router(config-router-bgp)# neighbor 1 route-map in <1-255>
```

or to the outgoing route map list.

```
Router(config-router-bgp)# neighbor 1 route-map out <1-255>
```

8. Exit Router BGP mode.

```
Router(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(config)# router bgp
Router(config-router-bgp)# aggregate-address <1-16> <IPv4 address> <mask>
Router(config-router-bgp)# aggregate-address <1-16> enable
Router(config-router-bgp)# exit
```

An example of creating a BGP aggregate route is shown in “Default Redistribution and Route Aggregation Example” on page 5-61.

Redistributing Routes

In addition to running multiple routing protocols simultaneously, Networking 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 5-53. 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 5-61.

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

BGP Attributes

The following two BGP attributes are discussed in this section: Local preference and metric (Multi-Exit Discriminator).

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:

- Using the BGP default local preference method, affecting the outbound direction only.

```
Router(config)# router bgp
Router(config_router_bgp)# local-preference
Router(config_router_bgp)# exit
```

- Using the route map local preference method, which affects both inbound and outbound directions.

```
Router(config)# route-map 1
Router(config_route_map)# local-preference
Router(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.

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 1/10Gb LAN Switch Module 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. The lower cost to the next hop of routes is selected.
6. In the case of equal cost, the eBGP route is preferred over iBGP.
7. If all routes have same route type (eBGP or iBGP), the route with the lower router ID is selected.

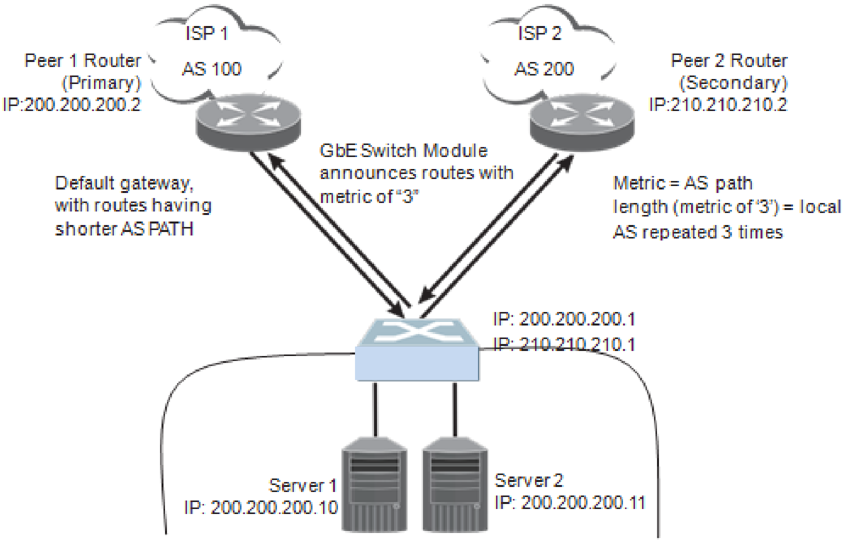
When the path is selected, BGP puts the selected path in its routing table and propagates the path to its neighbors.

BGP Failover Configuration

Use the following example to create redundant default gateways for 1/10Gb LAN Switch Module at a Web Host/ISP site, eliminating the possibility, should one gateway go down, that requests will be forwarded to an upstream router unknown to the switch.

As shown in Figure 23, 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 1/10Gb LAN Switch Module.

Figure 23. BGP Failover Configuration Example



On 1/10Gb LAN Switch Module, 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.

```
Router(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.

```
Router(config)# interface ip 1 address 200.200.200.1 255.255.255.0 enable
Router(config-ip-if)# exit

Router(config)# interface ip 2 address 210.210.210.1 255.255.255.0 enable
Router(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).

```
Router(config)# ip routing (Enable IP forwarding)
```

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.

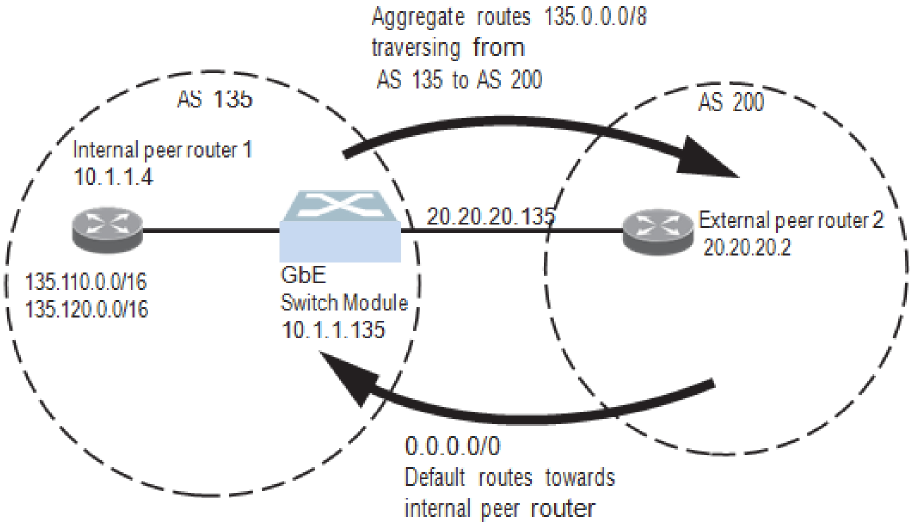
```
Router(config)# router bgp
Router(config-router-bgp)# neighbor 1 remote-address 200.200.200.2
Router(config-router-bgp)# neighbor 1 remote-as 100
Router(config-router-bgp)# neighbor 2 remote-address 210.210.210.2
Router(config-router-bgp)# neighbor 2 remote-as 200
```

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 24, you have two peer routers: an internal and an external peer router. Configure1/10Gb LAN Switch Module 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 24. Route Aggregation and Default Route Redistribution



1. Configure the IP interface.
2. Configure the AS number (AS 135) and router ID number (10.1.1.135).

```
Router(config)# router bgp
Router(config-router-bgp)# as 135
Router(config-router-bgp)# exit
Router(config)# ip router-id 10.1.1.135
```

3. Configure internal peer router 1 and external peer router 2.

```
Router(config)# router bgp
Router(config-router-bgp)# neighbor 1 remote-address 10.1.1.4
Router(config-router-bgp)# neighbor 1 remote-as 135
Router(config-router-bgp)# neighbor 2 remote-address 20.20.20.2
Router(config-router-bgp)# neighbor 2 remote-as 200
```

4. Configure redistribution for peer router 1.

```
Router(config-router-bgp)# neighbor 1 redistribute default-action redistribute
Router(config-router-bgp)# neighbor 1 redistribute fixed
```

5. Configure peer router 1 to import default routes from peer router 2

```
Router(config-router-bgp)# neighbor 2 redistribute default-action import
```

6. Configure external peer router 2 to export the default routes:

```
Router(config)# router bgp
Router(config-router-bgp)# neighbor 2 redistribute default-action originate

Or if the default route is learned via BGP or other protocol:

Router(config-router-bgp)# neighbor 2 redistribute default-action redistribute
```

7. Configure aggregation policy control.
   Configure the routes that you want aggregated.

```
Router(config-router-bgp)# aggregate-address 1 135.0.0.0 255.0.0.0
Router(config-router-bgp)# aggregate-address 1 enable
```

OSPF

Networking OS 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 1/10Gb LAN Switch Module:

- "OSPFv2 Overview" on page 5-64.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 5-69. This section describes how OSPFv2 is implemented in Networking OS, such as configuration parameters, electing the designated router, summarizing routes, defining route maps and so forth.
- "OSPFv2 Configuration Examples" on page 5-79.This section provides step-by-step instructions on configuring differentOSPFv2 examples:
  – Creating a simple OSPF domain
  – Creating virtual links
  – Summarizing routes
- "OSPFv3 Implementation in Networking OS" on page 5-86. 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). The LSDB for all routing devices within an area is identical but is not exchanged between different areas. Only routing updates are exchanged between areas, thereby significantly reducing the overhead for maintaining routing information on a large, dynamic network.

The following sections describe key OSPF concepts.

Types of OSPF Areas

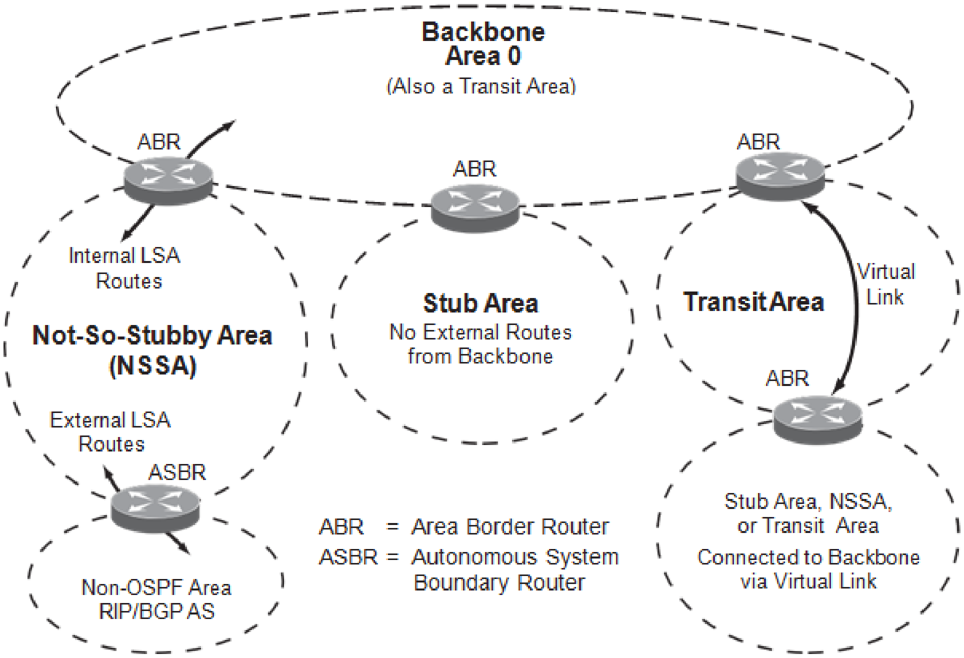
An AS can be broken into logical units known as *areas*. In any AS with multiple areas, one area must be designated as area 0, known as the *backbone*. The backbone acts as the central OSPF area. All other areas in the AS must be connected to the backbone. Areas inject summary routing information into the backbone, which then distributes it to other areas as needed.

As shown in Figure 25, OSPF defines the following types of areas:

- Stub Area—an area that is connected to only one other area. 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 can be configured to not be distributed into other areas.
- Transit Area—an area that carries data traffic which neither originates nor terminates in the area itself.

Figure 25. OSPF Area Types

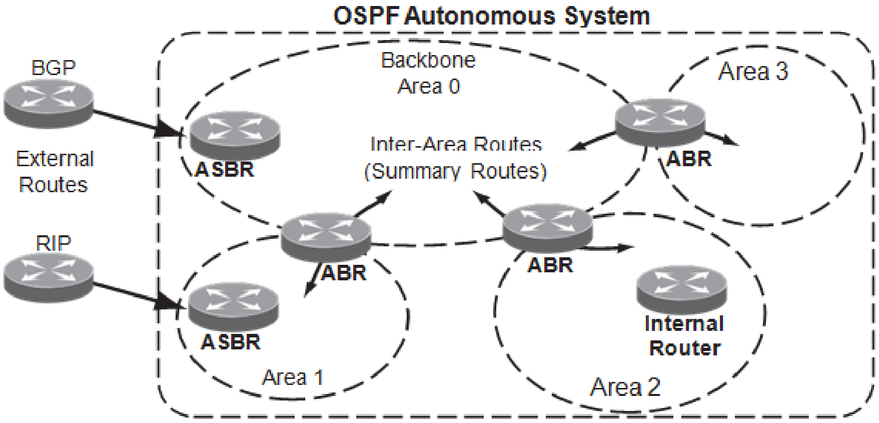


Types of OSPF Routing Devices

As shown in Figure 26, 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 26. 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' health. 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 deadintervals) 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 area's Designated Router (DR) and one as the area's Backup Designated Router (BDR). The DR is adjacent to all other neighbors 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 the DR, 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 an adjacency 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 an adjacency, 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

Networking OS supports a single instance of OSPF and up to 2K routes on the network. The following sections describe OSPF implementation in Networking OS:

- "Configurable Parameters" on page 5-69
- "Defining Areas" on page 5-69
- "Interface Cost" on page 5-71
- "Electing the Designated Router and Backup" on page 5-72
- "Summarizing Routes" on page 5-72
- "Default Routes" on page 5-73
- "Virtual Links" on page 5-74
- "Router ID" on page 4-75
- "Authentication" on page 4-75

Configurable Parameters

In Networking OS, OSPF parameters can be configured through the Command Line Interfaces (CLI), Browser-Based Interface (BBI), or through SNMP. For more information, see "Switch Administration" on page 1-2."

The CLI supports the following parameters: interface output cost, interface priority, dead and hello intervals, retransmission interval, and interface transmit delay.

In addition to the above parameters, you can also 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 (1/10Gb LAN Switch Module 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 5-74).

Up to three OSPF areas can be connected to 1/10Gb LAN Switch Module with Networking 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:

```
Router(config)# router ospf
Router(config-router-ospf)# area <area index> area-id <n.n.n.n>
Router(config-router-ospf)# area <area index> enable
Router(config-router-ospf)# exit
```

Note: The aindex option above 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 areaid portion of the command as explained in the following sections.

Assigning the Area Index

The `aindex<`*`area index`*`>` option is actually just an arbitrary index (0-2) used only by 1/10Gb LAN Switch Module. This index does not necessarily represent the OSPF area number, though for configuration simplicity, it should 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 `areaid` <*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 1/10Gb LAN Switch Module as "`area-id 0.0.0.2`".
On 1/10Gb LAN Switch Module, 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 commands are as follows:

```
Router(config)# interface ip <interface number>
Router(config-ip-if)# ip ospf area <area index>
Router(config-ip-if)# exit
```

For example, the following commands could be used to configure IPv4 interface 14 for a presence on the IPv4 10.10.10.1/24 network, to define OSPF area 1, and to attach the area to the network:

```
Router(config)# router ospf
Router(config-router-ospf)# area 1 area-id 0.0.0.1
Router(config-router-ospf)# area 1 enable
Router(config-router-ospf)# enable
Router(config-router-ospf)# exit
Router(config)# interface ip 14
Router(config-ip-if)# ip address 10.10.10.1 255.255.255.0 enable
Router(config-ip-if)# ip ospf area 1
Router(config-ip-if)# ip ospf enable
```

Note: OSPFv2 supports IPv4 only. IPv6 is supported in OSPFv3 (see "OSPFv3 Implementation in Networking OS" on page 5-86).

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:

```
Router(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 1/10Gb LAN Switch Module. The command is as follows:

```
Router(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:

```
Router(config-ip-if)# ip ospf passive-interface
Router(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:

```
Router(config)# router ospf
Router(config-router-ospf)# area-range <range number> address <IP address> <mask>
```

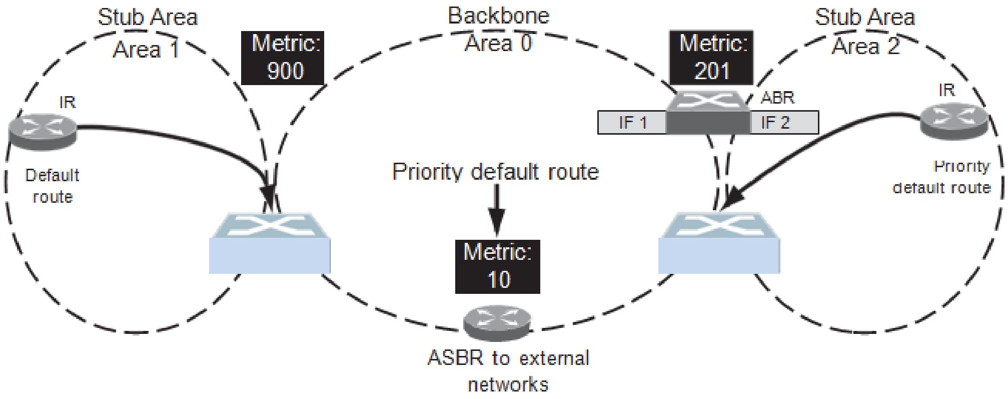
where `<range number>` is a number 1 to 16, `<IPv4 address>` is the base IP address for the range, and `<subnet mask>` is the IPv4 address mask for the range. For a detailed configuration example, see "Example 3: Summarizing Routes" on page 5-83.
Note: OSPFv2 supports IPv4 only. IPv6 is supported in OSPFv3 (see "OSPFv3 Implementation in Networking OS" on page 5-86).

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 1/10Gb LAN Switch Module 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 27), 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 27. 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(config-router-ospf)# default-information <metric value> <metric
type (1 or 2)>
```

In the command above, *<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:

```
Router(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 25 on page 5-65).

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 transitusing the following command:

```
Router(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 1/10Gb LAN Switch Module as one endpoint of a virtual link, use the following command:

```
Router(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 (nbr), the routing device at the target endpoint. Another router ID is needed when configuring a virtual link in the other direction. To provide 1/10Gb LAN Switch Module 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 5-80.

Router ID

Routing devices in OSPF areas are identified by a router ID, expressed in IP address format. The router ID is not required to be part of any IP interface range or in any OSPF area, and may even use 1/10Gb LAN Switch Module loopback interface (see "Loopback Interfaces in OSPF" on page 5-78).

The router ID can be configured in one of the following two ways:

- Dynamically (the default)—OSPF protocol configures the router ID as the lowest IP loopback interface IP address, if available, or else the lowest IP interface IP address, if available. Once dynamically configured, the router ID does not normally undergo further updates.
- Statically—Use the following command to manually configure the router ID:

```
Router(config-router-ospf)# ip router-id <IPv4 address>
```

To change the router ID from static to dynamic, set the router ID to 0.0.0.0, save the configuration, and reboot 1/10Gb LAN Switch Module. To view the router ID, enter:

```
Router(config-router-ospf)# show ip ospf
```

Authentication

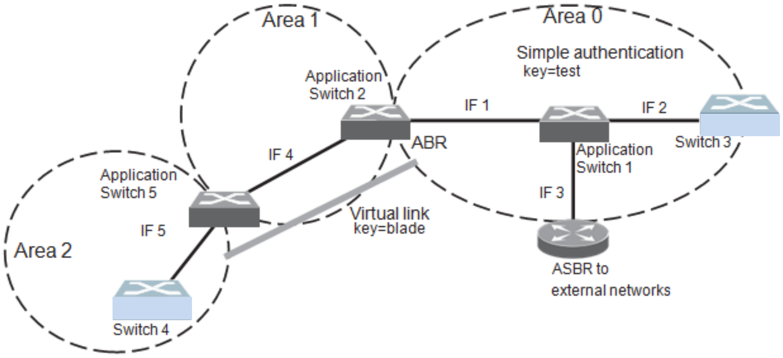
OSPF protocol exchanges can be authenticated so that only trusted routing devices can participate. This ensures less processing on routing devices that are not listening to OSPF packets

OSPF allows packet authentication and uses IP multicast when sending and receiving packets. Routers participate in routing domains based on pre-defined passwords. Networking 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 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 28. OSPF Authentication



Configuring Plain Text OSPF Passwords

To configure plain text OSPF passwords as shown in Figure use the following commands:

1. Enable OSPF authentication for Area 0 on switches 1, 2, and 3.

```
Router(config-router-ospf)# area 0 authentication-type password
Router(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.

```
Router(config)# interface ip 1
Router(config-ip-if)# ip ospf key test
Router(config-ip-if)# exit
Router(config)# interface ip 2
Router(config-ip-if)# ip ospf key test
Router(config-ip-if)# exit
Router(config)# interface ip 3
Router(config-ip-if)# ip ospf key test
Router(config-ip-if)# exit
```

3. Enable OSPF authentication for Area 2 on switch 4.

```
Router(config)# router ospf
Router(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.

```
Router(config-router-ospf)# area-virtual-link 1 key HITACHI
```

Configuring MD5 Authentication

Use the following commands to configure MD5 authentication on the switches shown in Figure :

1. Enable OSPF MD5 authentication for Area 0 on switches 1, 2, and 3.

```
Router(config-router-ospf)# area 0 authentication-type md5
```

2. Configure MD5 key ID for Area 0 on switches 1, 2, and 3.

```
Router(config-router-ospf)# message-digest-key 1 md5-key test
Router(config-router-ospf)# exit
```

3. Assign MD5 key ID to OSPF interfaces on switches 1, 2, and 3.

```
Router(config)# interface ip 1
Router(config-ip-if)# ip ospf message-digest-key 1
Router(config-ip-if)# exit
Router(config)# interface ip 2
Router(config-ip-if)# ip ospf message-digest-key 1
Router(config-ip-if)# exit
Router(config)# interface ip 3
Router(config-ip-if)# ip ospf message-digest-key 1
Router(config-ip-if)# exit
```

4. Enable OSPF MD5 authentication for Area 2 on switch 4.

```
Router(config)# router ospf
Router(config-router-ospf)# area 1 authentication-type md5
```

5. Configure MD5 key for the virtual link between Area 2 and Area 0 on switch 2 and switch 4.

```
Router(config-router-ospf)# message-digest-key 2 md5-key test
```

6. Assign MD5 key ID to OSPF virtual link on switches 2 and 4.

```
Router(config-router-ospf)# area-virtual-link 1 message-digest-key 2
Router(config-router-ospf)# exit
```

Host Routes for Load Balancing

Networking 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 should 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.

If redundant routes via multiple routing processes (such as OSPF, RIP, BGP, or static routes) exist on your network, the switch defaults to the OSPF-derived route.

Loopback Interfaces in OSPF

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 5-75), 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:

```
Router(config)# interface loopback <1-5>
Router(config-ip-loopback)# ip ospf area <area ID> enable
Router(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

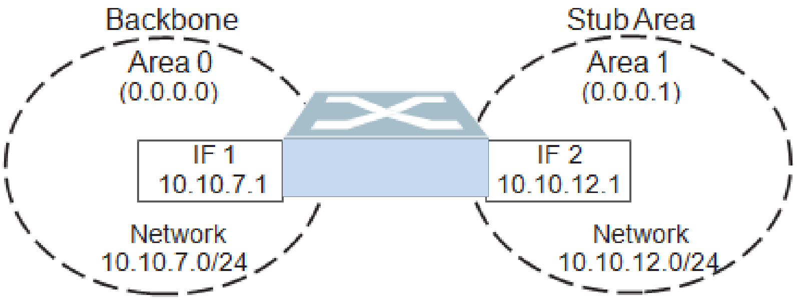
A summary of the basic steps for configuring OSPF on 1/10Gb LAN Switch Module is listed here.Detailed instructions for each of the steps is covered in the following sections:

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 29. A Simple OSPF Domain



Follow this procedure to configure OSPF support as shown in Figure 29:

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

```
Router(config)# interface ip 1
Router(config-ip-if)# ip address 10.10.7.1 255.255.255.0 enable
Router(config-ip-if)# exit
Router(config)# interface ip 2
Router(config-ip-if)# ip address 10.10.12.1 255.255.255.0 enable
Router(config-ip-if)# exit
```

Note: OSPFv2 supports IPv4 only. IPv6 is supported in OSPFv3 (see “OSPFv3 Implementation in Networking OS” on page 5-86).

2. Enable OSPF.

```
Router(config)# router ospf
Router(config-router-ospf)# enable
```

3. Define the backbone.
   The backbone is always configured as a transit area using `areaid 0.0.0.0`.

```
Router(config-router-ospf)# area 0 area-id 0.0.0.0
Router(config-router-ospf)# area 0 type transit
Router(config-router-ospf)# area 0 enable
```

4. Define the stub area.

```
Router(config-router-ospf)# area 1 area-id 0.0.0.1
Router(config-router-ospf)# area 1 type stub
Router(config-router-ospf)# area 1 enable
Router(config-router-ospf)# exit
```

5. Attach the network interface to the backbone.

```
Router(config)# interface ip 1
Router(config-ip-if)# ip ospf area 0
Router(config-ip-if)# ip ospf enable
Router(config-ip-if)# exit
```

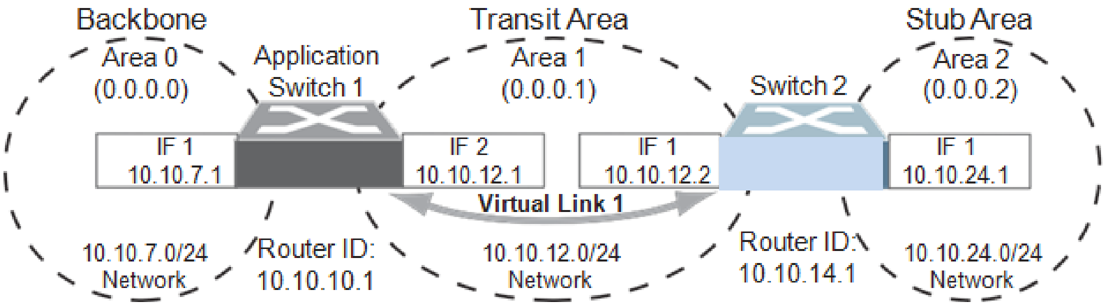
6. Attach the network interface to the stub area.

```
Router(config)# interface ip 2
Router(config-ip-if)# ip ospf area 1
Router(config-ip-if)# ip ospf enable
Router(config-ip-if)# exit
```

Example 2: Virtual Links

In the example shown in Figure 30, 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 30. Configuring a Virtual Link



Note: OSPFv2 supports IPv4 only. IPv6 is supported in OSPFv3 (see “OSPFv3 Implementation in Networking OS” on page 5-86).

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

```
Router(config)# interface ip 1
Router(config-ip-if)# ip address 10.10.7.1 255.255.255.0 enable
Router(config-ip-if)# exit
Router(config)# interface ip 2
Router(config-ip-if)# ip address 10.10.12.1 255.255.255.0 enable
Router(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.

```
Router(config)# ip router-id 10.10.10.1
```

3. Enable OSPF.

```
Router(config)# router ospf
Router(config-router-ospf)# enable
```

4. Define the backbone.

```
Router(config-router-ospf)# area 0 area-id 0.0.0.0
Router(config-router-ospf)# area 0 type transit
Router(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.

```
Router(config-router-ospf)# area 1 area-id 0.0.0.1
Router(config-router-ospf)# area 1 type transit
Router(config-router-ospf)# area 1 enable
Router(config-router-ospf)# exit
```

6. Attach the network interface to the backbone.

```
Router(config)# interface ip 1
Router(config-ip-if)# ip ospf area 0
Router(config-ip-if)# ip ospf enable
Router(config-ip-if)# exit
```

7. Attach the network interface to the transit area.

```
Router(config)# interface ip 2
Router(config-ip-if)# ip ospf area 1
Router(config-ip-if)# ip ospf enable
Router(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 5-82.

```
Router(config)# router ospf
Router(config-router-ospf)# area-virtual-link 1 area 1
Router(config-router-ospf)# area-virtual-link 1 neighbor-router 10.10.14.1
Router(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

```
Router(config)# interface ip 1
Router(config-ip-if)# ip address 10.10.12.2 255.255.255.0 enable
Router(config-ip-if)# exit
Router(config)# interface ip 2
Router(config-ip-if)# ip address 10.10.24.1 255.255.255.0 enable
Router(config-ip-if)# exit
```

2. Configure the router ID.
   A router ID is required when configuring virtual links. This router ID should be the same one specified as the target virtual neighbor (`nbr`) on switch 1 in Step 8 on page 5-83.

```
Router(config)# ip router-id 10.10.14.1
```

3. Enable OSPF.

```
Router(config)# router ospf
Router(config-router-ospf)# enable
```

4. Define the backbone.
   This version of Networking OS requires that a backbone index be configured on the non-backbone end of the virtual link as follows:

```
Router(config-router-ospf)# area 0 area-id 0.0.0.0
Router(config-router-ospf)# area 0 enable
```

5. Define the transit area.

```
Router(config-router-ospf)# area 1 area-id 0.0.0.1
Router(config-router-ospf)# area 1 type transit
Router(config-router-ospf)# area 1 enable
```

6. Define the stub area.

```
Router(config-router-ospf)# area 2 area-id 0.0.0.2
Router(config-router-ospf)# area 2 type stub
Router(config-router-ospf)# area 2 enable
Router(config-router-ospf)# exit
```

7. Attach the network interface to the transmit area:

```
Router(config)# interface ip 1
Router(config-ip-if)# ip ospf area 1
Router(config-ip-if)# ip ospf enable
Router(config-ip-if)# exit
```

8. Attach the network interface to the stub area.

```
Router(config)# interface ip 2
Router(config-ip-if)# ip ospf area 2
Router(config-ip-if)# ip ospf enable
Router(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 5-81.

```
Router(config)# router ospf
Router(config-router-ospf)# area-virtual-link 1 area 1
Router(config-router-ospf)# area-virtual-link 1 neighbor-router 10.10.10.1
Router(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

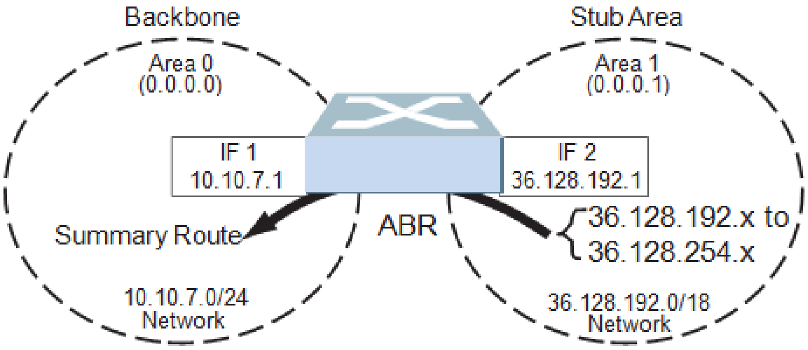
By default, ABRs advertise all the network addresses from one area into another area. Route summarization can be used for consolidating advertised addresses and reducing the perceived complexity of the network.

If the 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 the 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 5-86).

Figure 31. 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.

Follow this procedure to configure OSPF support as shown in Figure 31:

1. Configure IP interfaces for each network which will be attached to OSPF areas.

```
Router(config)# interface ip 1
Router(config-ip-if)# ip address 10.10.7.1 255.255.255.0 enable
Router(config-ip-if)# exit
Router(config)# interface ip 2
Router(config-ip-if)# ip address 36.128.192.1 255.255.255.0 enable
Router(config-ip-if)# exit
```

2. Enable OSPF.

```
Router(config)# router ospf
Router(config-router-ospf)# enable
```

3. Define the backbone.

```
Router(config-router-ospf)# area 0 area-id 0.0.0.0
Router(config-router-ospf)# area 0 type transit
Router(config-router-ospf)# area 0 enable
```

4. Define the stub area.

```
Router(config-router-ospf)# area 1 area-id 0.0.0.1
Router(config-router-ospf)# area 1 type stub
Router(config-router-ospf)# area 1 enable
Router(config-router-ospf)# exit
```

5. Attach the network interface to the backbone.

```
Router(config)# interface ip 1
Router(config-ip-if)# ip ospf area 0
Router(config-ip-if)# ip ospf enable
Router(config-ip-if)# exit
```

6. Attach the network interface to the stub area.

```
Router(config)# interface ip 2
Router(config-ip-if)# ip ospf area 1
Router(config-ip-if)# ip ospf enable
Router(config-ip-if)# exit
```

7. Configure route summarization by specifying the starting address and mask of the range of addresses to be summarized.

```
Router(config)# router ospf
Router(config-router-ospf)# area-range 1 address 36.128.192.0 255.255.192.0
Router(config-router-ospf)# area-range 1 area 1
Router(config-router-ospf)# area-range 1 enable
Router(config-router-ospf)# exit
```

8. Use the hide command to prevent a range of addresses from advertising to the backbone.

```
Router(config)# router ospf
Router(config-router-ospf)# area-range 2 address 36.128.200.0 255.255.255.0
Router(config-router-ospf)# area-range 2 area 1
Router(config-router-ospf)# area-range 2 hide
Router(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 *Networking OS 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 should 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 1/10Gb LAN Switch Module. 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.

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 5-12 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 CLI

```
Router(config)# ipv6 router ospf    (OSPFv3 router config mode)
Router(config-router-ospf3)# ?

Router(config)# interface ip <Interface number> (Configure OSPFv3)
Router(config-ip-if)# ipv6 ospf ?   (OSPFv3 interface config)

Router# 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:

```
Router(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: `Router(config-ip-if)# ipv6 ospf linklsasuppress`

OSPFv3 Limitations

Networking OS 7.8 does not currently support the following OSPFv3 features:

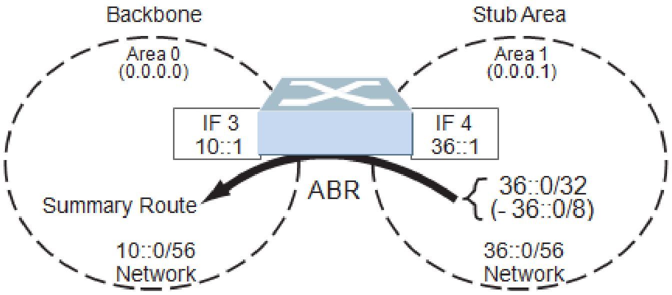
- Multiple instances of OSPFv3 on one IPv6 link.
- Authentication of OSPFv3 packets via IPv6 Security (IPsec) for virtual links.

OSPFv3 Configuration Example

The following example depicts the OSPFv3 equivalent configuration of “Example 3: Summarizing Routes” on page 5-83 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 for the 36.:0/32 portion of the 36::0/56 network except for the routes in the 36::0/8 range.

Figure 32. 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 31:

1. Configure IPv6 interfaces for each link which will be attached to OSPFv3 areas.

```
Router(config)# interface ip 3
Router(config-ip-if)# ipv6 address 10:0:0:0:0:0:0:1
Router(config-ip-if)# ipv6 prefixlen 56
Router(config-ip-if)# enable
Router(config-ip-if)# exit
Router(config)# interface ip 4
Router(config-ip-if)# ip address 36:0:0:0:0:0:1
Router(config-ip-if)# ipv6 prefixlen 56
Router(config-ip-if)# enable
Router(config-ip-if)# exit
```

   This is equivalent to configuring the IP address and netmask for IPv4 interfaces.

2. Enable OSPFv3.

```
Router(config)# ipv6 router ospf
Router(config-router-ospf3)# enable
```

   This is equivalent to the OSPFv2 enable option in the router ospf command path.

3. Define the backbone.

```
Router(config-router-ospf3)# area 0 area-id 0.0.0.0
Router(config-router-ospf3)# area 0 type transit
Router(config-router-ospf3)# area 0 enable
```

   This is identical to OSPFv2 configuration.

4. Define the stub area.

```
Router(config-router-ospf3)# area 1 area-id 0.0.0.1
Router(config-router-ospf3)# area 1 type stub
Router(config-router-ospf3)# area 1 enable
Router(config-router-ospf3)# exit
```

   This is identical to OSPFv2 configuration.

5. Attach the network interface to the backbone.

```
Router(config)# interface ip 3
Router(config-ip-if)# ipv6 ospf area 0
Router(config-ip-if)# ipv6 ospf enable
Router(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.

```
Router(config)# interface ip 4
Router(config-ip-if)# ipv6 ospf area 1
Router(config-ip-if)# ipv6 ospf enable
Router(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.

```
Router(config)# ipv6 router ospf
Router(config-router-ospf3)# area-range 1 address 36:0:0:0:0:0:0:0 32
Router(config-router-ospf3)# area-range 1 area 0
Router(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.

```
Router(config-router-ospf)# area-range 2 address 36:0:0:0:0:0:0:0 8
Router(config-router-ospf)# area-range 2 area 0
Router(config-router-ospf)# area-range 2 hide
Router(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:

```
Router(config# interface ip 10
Router(config-ip-if)# ipv6 address 10:0:0:0:0:0:0:12 64
Router(config-ip-if)# vlan 10
Router(config-ip-if)# enable
Router(config-ip-if)# ipv6 ospf area 0
Router(config-ip-if)# ipv6 ospf retransmit-interval 5
Router(config-ip-if)# ipv6 ospf transmit-delay 1
Router(config-ip-if)# ipv6 ospf priority 1
Router(config-ip-if)# ipv6 ospf hello-interval 10
Router(config-ip-if)# ipv6 ospf dead-interval 40
Router(config-ip-if)# ipv6 ospf network point-to-multipoint
Router(config-ip-if)# ipv6 ospf poll-interval 120
Router(config-ip-if)# ipv6 ospf enable
Router(config-ip-if)# exit
```

2. Enable OSPFv3:

```
Router(config# ipv6 router ospf
Router(config-router-ospf3)# router-id 12.12.12.12
Router(config-router-ospf3)# enable
```

3. Define the backbone.

```
Router(config-router-ospf3)# area 0 area-id 0.0.0.0
Router(config-router-ospf3)# area 0 stability-interval 40
Router(config-router-ospf3)# area 0 default-metric 1
Router(config-router-ospf3)# area 0 default-metric type 1
Router(config-router-ospf3)# area 0 translation-role candidate
Router(config-router-ospf3)# area 0 type transit
Router(config-router-ospf3)# area 0 enable
```

4. Configure neighbor entry:

```
Router(config-router-ospf3)# neighbor 1 address fe80:0:0:0:dceb:ff:fe00:9
Router(config-router-ospf3)# neighbor 1 interface 10
Router(config-router-ospf3)# neighbor 1 priority 1
Router(config-router-ospf3)# neighbor 1 enable
```

Protocol Independent Multicast

Networking OS supports Protocol Independent Multicast (PIM) in Sparse Mode (PIM-SM) and Dense Mode (PIM-DM).

Note: Networking OS 7.8 does not support IPv6 for PIM.

The following sections discuss PIM support for 1/10Gb LAN Switch Module:

- "PIM Overview" on page 5-91
- "Supported PIM Modes and Features" on page 5-92
- "Basic PIM Settings" on page 5-92
- "Additional Sparse Mode Settings" on page 5-95
- "Using PIM with Other Features" on page 5-96
- "PIM Configuration Examples" on page 5-97

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 Networking OS 7.8:

- 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 sup- ported 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 cur- rently not available using the menu-based CLI, 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: PIM can be configured through the CLI only. PIM configuration and information are not available using the menu-based CLI, 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 CLI commands:

```
Router(config)# [no] ip pim enable
```

Defining a PIM Network Component

1/10Gb LAN Switch Module 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:

```
Router(config)# ip pim component <1-2>
Router(config-ip-pim-comp)# mode {sparse|dense}
Router(config-ip-pim-comp)# exit
```

The sparseoption 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 5-95).

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, but each interface can belong to only one VLAN.

To define an IP interface for use with PIM, first configured the interface with an IPv4 address and VLAN as follows:

```
Router(config)# interface ip <Interface number>
Router(config-ip-if)# ip address <IPv4 address>  <IPv4 mask>
Router(config-ip-if)# vlan <VLAN number>
Router(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:

```
Router(config-ip-if)# ip pim enable
Router(config-ip-if)# ip pim component-id <1-2>
Router(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

1/10Gb LAN Switch Module accepts connection to up to 8 PIM interfaces. By default, the switch accepts all PIM neighbors attached to the PIM-enabled interfaces, up to the maximum number (24 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:

```
Router(config)# interface ip <Interface number>
Router(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:

```
Router(config-ip-if)# ip pim neighbor-addr <neighbor IPv4 address>  allow
```

To remove a PIM neighbor from the accepted list, use the following command.

```
Router(config-ip-if)# ip pim neighbor-addr <neighbor IPv4 address>  deny
Router(config-ip-if)# exit
```

You can view configured PIM neighbor filters globally or for a specific IP interface using the following commands:

```
Router(config)# show ip pim neighbor-filters
Router(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, 1/10Gb LAN Switch Module 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:

```
Router(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:

```
Router(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 5-96).
Alternately, if no election is desired, the switch can provide a static RP, specified using the following command:

```
Router(config-ip-pim-comp)# rp-static rp-address <group address>  <group
address mask> <static RP IPv4 address>
```

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:

```
Router(config-ip-pim-comp)# rp-candidate holdtime <0-255>
Router(config-ip-pim-comp)# exit
```

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):

```
Router(config)# interface ip <Interface number>
Router(config-ip-if)# ip pim dr-priority <value (0-4294967294)>
Router(config-ip-if)# exit
```

Note: A value of 0 (zero) specifies that 1/10Gb LAN Switch Module will not act as the DR. This setting requires 1/10Gb LAN Switch Module to be connected to a peer that has a DR priority setting of 1 or higher in order 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:

```
Router(config)# interface ip <Interface number>
Router(config-ip-if)# ip pim cbsr-preference <0 to 255>
Router(config-ip-if)# exit
```

A value of 255 highly prefers the local interface as a BSR. A value of -1 indicates that the PIM CBSR preference is not configured on the local interface.

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 5-35):

- IGMP static joins can be configured with a PIM-SM or PIM-DM multicast group IPv4 address. Using the CLI:

```
Router(config)# ip mroute <multicast group IPv4 address>  <VLAN>  <port>
```

- IGMP Query 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 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:

```
Router(config)# ip pim enable
```

2. Configure a PIM network component with dynamic RP settings, and set it for PIM Sparse Mode:

```
Router(config)# ip pim component 1
Router(config-ip-pim-comp)# mode sparse
Router(config-ip-pim-comp)# rp-candidate rp-address 225.1.0.0 255.255.0.0
        10.10.1.1
Router(config-ip-pim-comp)# rp-candidate holdtime 100
Router(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:

```
Router(config)# interface ip 111
Router(config-ip-if)# ip address 10.10.1.1 255.255.255.255
Router(config-ip-if)# vlan 11
Router(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:

```
Router(config-ip-if)# ip pim enable
Router(config-ip-if)# ip pim component-id 1
```

Note: Because, PIM component 1 is assigned to the interface by default, the component-idcommand is needed only if the setting has been previously changed.

5. Set the Bootstrap Router (BSR) preference:

```
Router(config-ip-if)# ip pim cbsr-preference 135
Router(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:

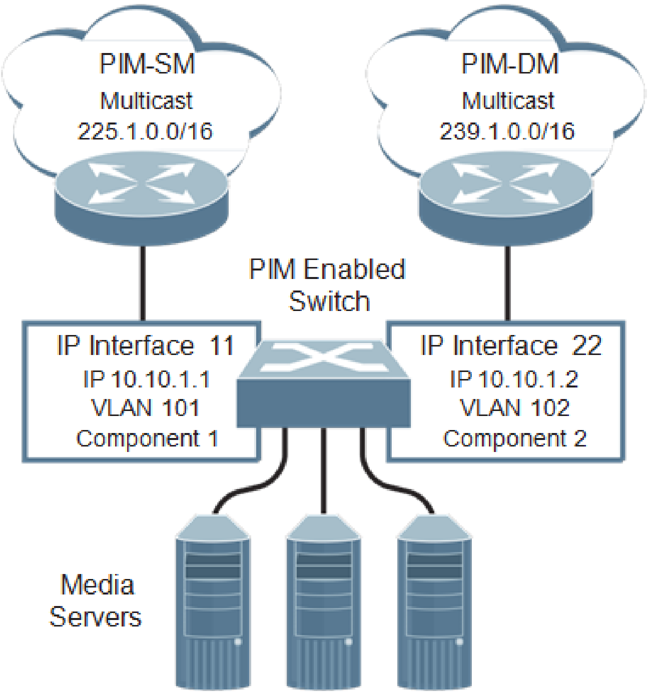
```
Router(config)# ip pim static-rp enable
Router(config)# ip pim component 1
Router(config-ip-pim-comp)# rp-static rp-address 225.1.0.0 255.255.0.0
10.10.1.1
Router(config-ip-pim-comp)# exit
```

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 should 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 33.
Note: In the following example, since the receivers and sources are connected in different areas, the border router must be configured for the IPMC traffic to be forwarded. Networking OS supports only partial configuration of PIM border router.

Figure 33. 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.

```
Router(config)# ip pim enable
```

2. Configure a PIM component and set the PIM mode:

```
Router(config)# ip pim component 2
Router(config-ip-pim-comp)# mode dense
Router(config-ip-pim-comp)# exit
```

3. Define an IP interface for use with PIM:

```
Router(config)# interface ip 22
Router(config-ip-if)# ip address 10.10.1.2 255.255.255.255
Router(config-ip-if)# vlan 102
Router(config-ip-if)# enable
```

4. Enable PIM on the IP interface and assign the PIM component:

```
Router(config-ip-if)# ip pim enable
Router(config-ip-if)# ip pim component-id 2
Router(config-ip-if)# exit
```

5. (Optional) Configure PIM border router if the IPMC traffic is flowing between PIM domains:

```
Router(config)# ip pim pmbr enable
Router(config)# interface ip 22
Router(config-ip-if)# ip pim border-bit
Router(config-ip-if)# exit
Router(config)# interface ip 11
Router(config-ip-if)# ip pim border-bit
Router(config-ip-if)# exit
```

Note: For PIM Dense Mode, the DR, RP, and BSR settings do not apply.

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.

□ Basic Redundancy

□ Layer 2 Failover

□ Virtual Router Redundancy Protocol

Basic Redundancy

Networking OS 7.8 includes various features for providing basic link or device redundancy:

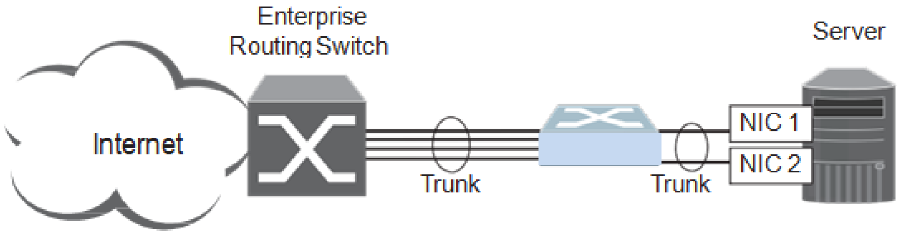
- “Trunking for Link Redundancy” on page 6-2
- “Hot Links” on page 6-3

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 34, 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 34. Trunking Ports for Link Redundancy



For more information on trunking, see “Ports and Trunking” on page 3-14.

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 is set to active and forwards traffic, while the Backup interface is set to standby. If preemption is disabled, the Master interface transitions to the standby state and blocks traffic until the Backup interface fails.

You may select a physical port, static trunk, or an LACP adminkey as a Hot Link interface. Only external uplink ports can be members of a Hot Links trigger 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 (`Router(config)#hotlinks trigger <x>  forward-delay 10`), 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 FBD update option uses the station update rateto determine the rate at which to send FDB packets.

Configuration Guidelines

The following configuration guidelines apply to Hot links:

- Only external ports and inter-switch links can be configured as Hot Links.
- When Hot Links is turned on, STP should be disabled on hotlinks interfaces.
- 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.

```
Router(config)# hotlinks trigger 1 enable     (Enable Hot Links Trigger 1)
Router(config)# hotlinks trigger 1 master port 38     (Add port to Master interface)
Router(config)# hotlinks trigger 1 backup port 39    (Add port to Backup interface)
Router(config)# hotlinks enable      (Turn on Hot Links)
```

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 blade 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.

Auto Monitoring Trunk Links

Layer 2 Failover can be enabled on any trunk group in 1/10Gb LAN Switch Module, including LACP trunks. Trunks can be added to failover trigger groups. Then, if some specified number of trigger links fail, the switch disables all the internal ports in the switch (unless VLAN Monitor is turned on). When the 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 trigger group return to service, the switch enables the internal ports. This causes the NIC team on the affected server blades 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 internal links come up, which can take up to five seconds.

VLAN Monitor

The VLAN Monitor allows Layer 2 Failover to discern different VLANs. With VLAN Monitor turned on:

- If enough links in a trigger fail (see "Setting the Failover Limit" on page 6-7), the switch disables all internal ports that reside in the same VLAN membership as the trunk(s) in the trigger.
- When enough links in the trigger return to service, the switch enables the internal ports that reside in the same VLAN membership as the trunk(s) in the trigger.

If you turn off the VLAN Monitor (`Router# no failover vlan`), only one failover trigger is allowed. When a link failure occurs on the trigger, the switch disables all internal server-blade ports.

Auto Monitor Configurations

Figure 35 is a simple example of Layer 2 Failover. One 1/10Gb LAN Switch Module 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 ports. This action causes a failover event on Server 1 and Server 2.

Figure 35. Basic Layer 2 Failover

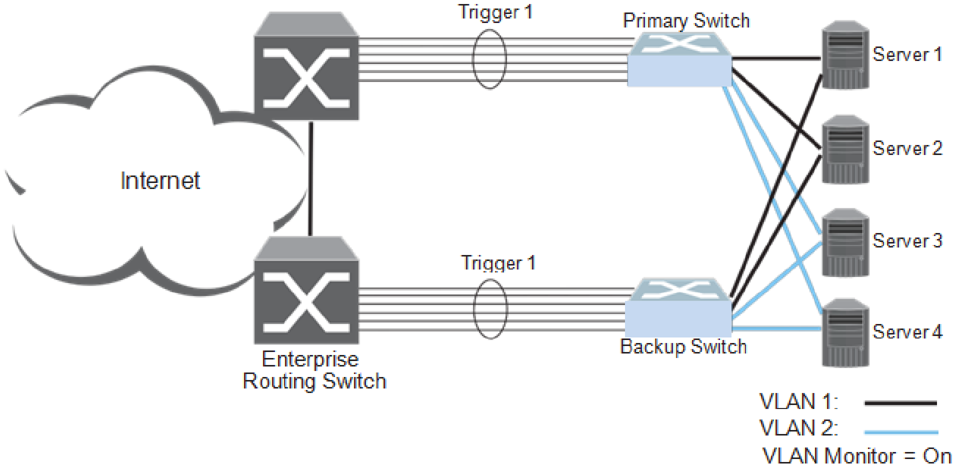


Figure 58 shows a configuration with two trunks, each in a different Failover Trigger. Switch 1 is the primary switch for Server 1 and Server 2. Switch 2 is the primary switch for Server 3 and Server 4. VLAN Monitor is turned on. STP is turned off.

If all links go down in trigger 1, Switch 1 disables all internal ports that reside in VLAN 1. If all links in trigger 2 go down, Switch 1 disables all internal ports that reside in VLAN 2.

Figure 36. Two trunks, each in a different Failover Trigger

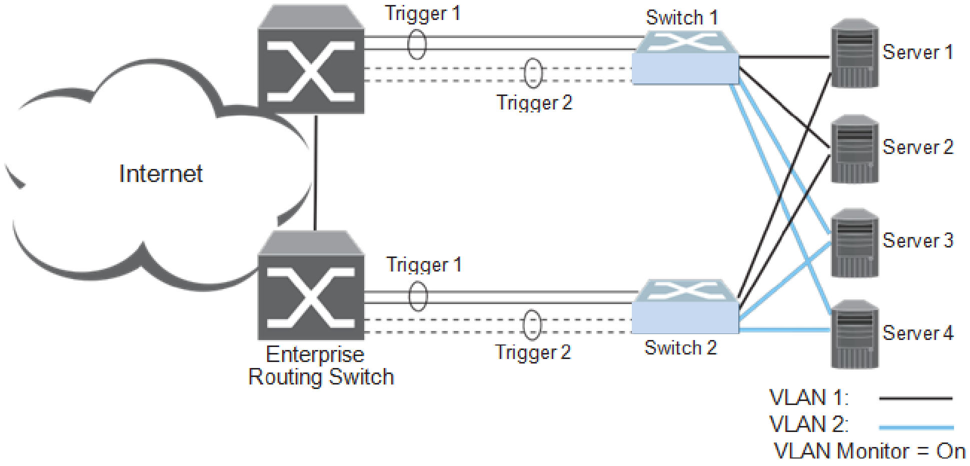
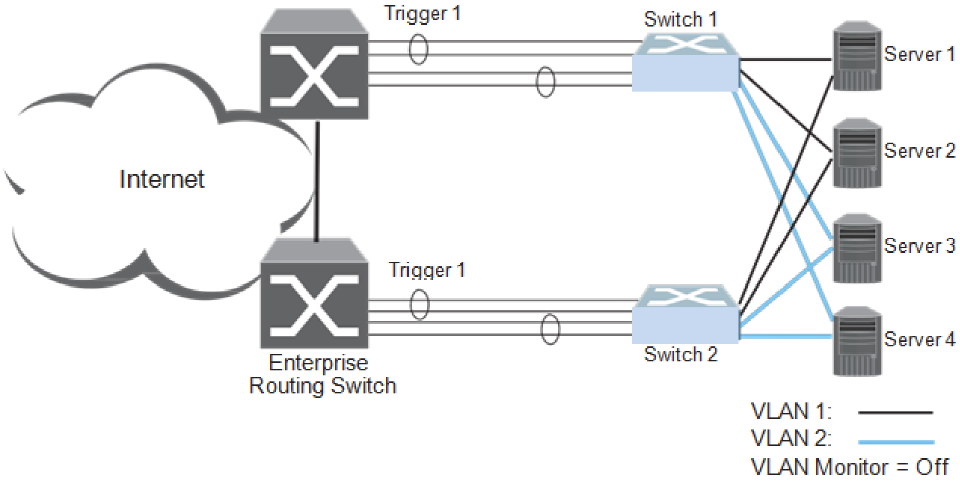


Figure 59 shows a configuration with two trunks. VLAN Monitor is turned off, so only one Failover Trigger is configured on each switch. Switch 1 is the primary switch for Server 1 and Server 2. Switch 2 is the primary switch for Server 3 and Server 4. STP is turned off.

If all links in trigger 1 go down, switch 1 disables all internal links to server blades.

Figure 37. Two trunks, one Failover Trigger



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/or trunks to monitor for link failures (a monitor list), and another set of ports and/or 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 in the chassis.

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 LinkUpstate.
- If STP is enabled, the port must be in the Forwardingstate.
- If the port is part of an LACP trunk, the port must be in the Aggregated state.

If any of the above 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 Downstate, Blockingstate (if STP is enabled on the port), or NotAggregatedstate (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:

```
Router# show interface link       (View port link status)
Router# show interface port <x> spanning-tree stp <x> (View port
administrative status)
Router# show lacp information     (View port LACP status)
```

L2 Failover with Other Features

L2 Failover works together with Link Aggregation Control Protocol (LACP) and with Spanning Tree Protocol (STP), as described in the next sections.

LACP

Link Aggregation Control Protocol allows the switch to form dynamic trunks. You can use the admin key to add up to 64 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). The switch automatically disables the appropriate ports.

When the switch determines that ports in the trigger are in STP Forwarding state,then it automatically enables the appropriate internal ports, based on the VLAN monitor. The switch *fails back* to normal operation*.*

Configuration Guidelines

This section provides important information about configuring Layer 2 Failover.
Note: Auto Monitor and Manual Monitor are mutually exclusive. They cannot both be configured on the switch.

Auto Monitor Guidelines

- Any specific failover trigger may monitor static trunks only or LACP trunks only, but not both.
- All external ports in all static or LACP trunks added to any specific failover trigger must belong to the same VLAN.
- A maximum of two LACP keys can be added per trigger.
- When VLAN Monitor is on, the following additional guidelines apply:
  – All external ports in all static or LACP trunks added to a specific failover trigger must belong to the same VLAN and have the same PVID.
  – Different triggers are not permitted to operate on the same VLAN.
  – Different triggers are not permitted to operate on the same internal port.
  – For each portchannel in a specific failover trigger, that is a memeber in multiple VLANs/STGs, the trigger considers it down only if it is down/blocking in all STGs.

Manual Monitor Guidelines

- A Manual Monitor can monitor only external ports.
- Any specific failover trigger can monitor external ports only.
- A maximum of two LACP keys can be added per trigger.
- Management ports, FC ports, and stacking ports cannot be monitored.
- Control ports for different triggers must not overlap. Monitor ports may overlap.

Configuring Layer 2 Failover

Auto Monitor Example

The following procedure pertains to the configuration shown in Figure 35.

1. Configure Network Adapter Teaming on the servers.
2. Define a trunk group on 1/10Gb LAN Switch Module.

```
Router(config)# portchannel 1 port EXT1,EXT2,EXT3 enable
```

3. Configure Failover parameters.

```
Router(config)# failover trigger 1 enable
Router(config)# failover trigger 1 limit <0-1024>
Router(config)# failover trigger 1 amon portchannel 1
```

4. Verify the configuration.

```
Router(config)# show failover trigger 1 information
```

Manual Monitor Example

Use the following procedure to configure a Layer 2 Failover Manual Monitor.

1. Configure Network Adapter Teaming on the servers.
2. Specify the links to monitor.

```
Router(config)# failover trigger 1 mmon monitor member EXT4,EXT5,EXT6
```

3. Specify the links to disable when the failover limit is reached.

```
Router(config)# failover trigger 1 mmon control member INT13,INT14
```

4. Configure general Layer 2 Failover parameters.

```
Router(config)# failover trigger 1 enable
Router(config)# failover trigger 1 limit <0-1024>
```

5. Enable failover globally.

```
Router(config)# failover enable
```

6. Verify the configuration.

```
Router(config)# show failover trigger 1 information
```

Virtual Router Redundancy Protocol

1/10Gb LAN Switch Module supports IPv4
high-availability network topologies through an enhanced implementation of the Virtual Router Redundancy Protocol (VRRP).
Note: Networking OS 7.8 does not support IPv6 for VRRP.

The following topics are discussed in this chapter:

- "VRRP Overview" on page 6-11. This section discusses VRRP operation and Networking OS redundancy configurations.
- "Failover Methods" on page 6-14. This section describes the three modes of high availability.
- " Networking OS Extensions to VRRP" on page 6-17. This section describes VRRP enhancements implemented in Networking OS.
- "Virtual Router Deployment Considerations" on page 6-18. This section describes issues to consider when deploying virtual routers.
- "High Availability Configurations" on page 6-19. This section discusses the more useful and easily deployed redundant configurations.
  - "Active-Active Configuration" on page 6-19
  - "Hot-Standby Configuration" on page 6-23

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 server blades.

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 6-13 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. Should the virtual router master fail, 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.

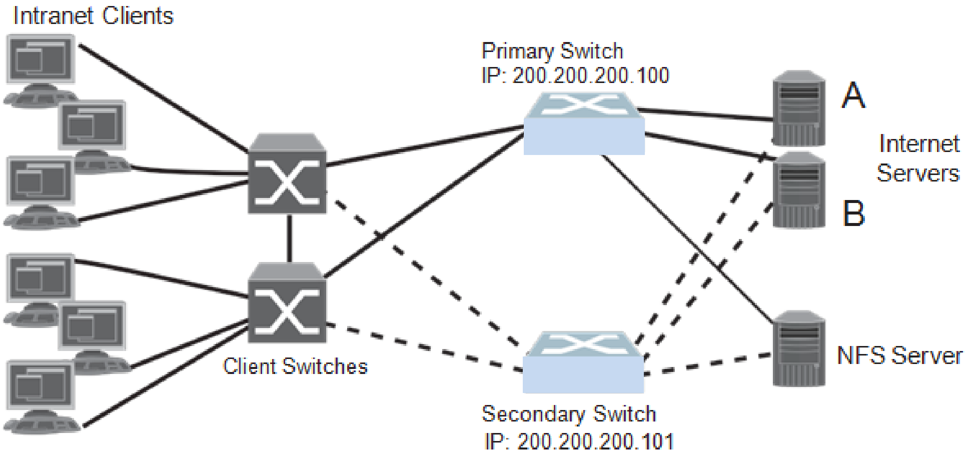
A backup router can stop receiving advertisements for one of two reasons—the master can be down, or all communications links between the master and the backup can be down. If the master has failed, it is clearly desirable for the backup (or one of the backups, if there is more than one) to become the 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. Traditionally, these configurations have been *hot-standby* configurations, where one switch is active and the other is in a standby mode. A non-VRRP hot-standby configuration is shown in the figure below:

Figure 38. A Non-VRRP, Hot-Standby Configuration



While hot-standby configurations increase site availability by removing single points-of-failure, service providers increasingly view them as an inefficient use of network resources because one functional application switch sits by idly until a failure calls it into action. Service providers now demand that vendors' equipment support redundant configurations where all devices can process traffic when they are healthy, increasing site throughput and decreasing user response times when no device has failed.

Networking OS high availability configurations are based on VRRP. The implementation of VRRP includes proprietary extensions.

The Networking OS implementation of VRRP supports the following modes of high availability:

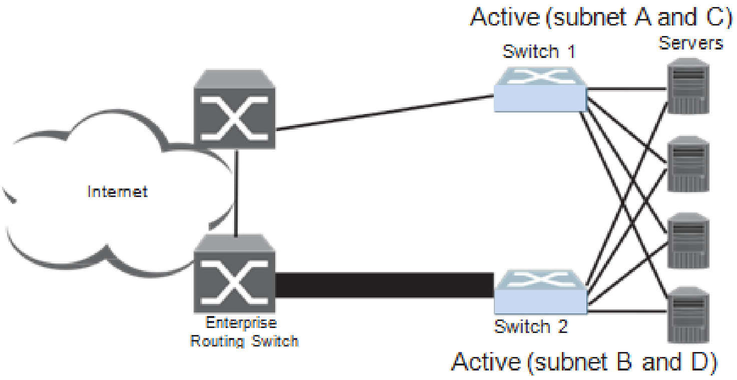
- Active-Active—based on proprietary Networking OS extensions to VRRP
- Hot-Standby—supports Network Adapter Teaming on your server blades

Active-Active Redundancy

In an active-active configuration, shown in Figure 39, 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 6-19.

Figure 39. Active-Active Redundancy

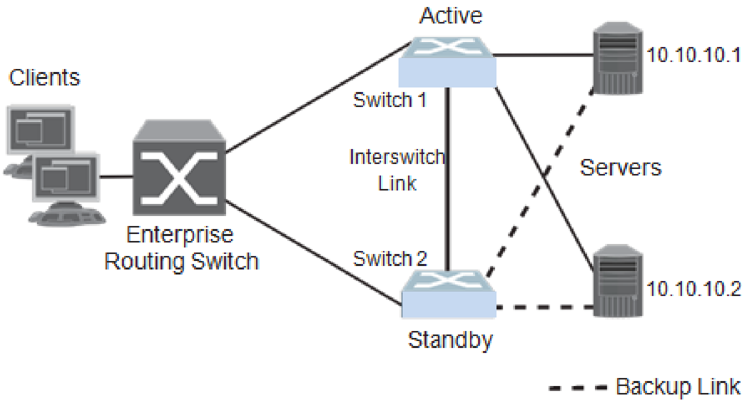


Hot-Standby Redundancy

The primary application for VRRP-based hot-standby is to support Server Load Balancing when you have configured Network Adapter Teaming on your server blades. With Network Adapter Teaming, the NICs on each server share the same IPv4 address, and are configured into a team. One NIC is the primary link, and the others are backup links. For more details, refer to the relevant network adapter documentation.

The hot-standby model is shown in Figure 40.

Figure 40. Hot-Standby Redundancy



Virtual Router Group

The virtual router group ties all virtual routers on the switch together as a single entity. By definition, hot-standby requires that all virtual routers failover as a group, and not individually. As members of a group, all virtual routers on the switch (and therefore the switch itself), are in either a master or standby state.

The virtual router group cannot be used for active-active configurations or any other configuration that require shared interfaces.

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 128 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 Networking OS.

Networking 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.

Networking OS can track the attributes listed in Table 25 :

Table 25. VRRP Tracking Parameters

Parameter	**Description**
Number of IP interfaces on the switch that are active ("up") `tracking-priority-increment interfaces`	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 `tracking-priority-increment ports`	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. **Note**: In a hot-standby configuration, only external ports are tracked.
Number of virtual routers in master mode on the switch `tracking-priority-increment virtual-routers`	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.

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 6-18 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 255. Use the following commands to configure the virtual router ID:

```
Router(config)# router vrrp
Router(config-vrrp)# virtual-router 1 virtual-router-id <1-255>
```

Configuring the Switch for Tracking

Tracking configuration largely depends on user preferences and network environment. Consider the configuration shown in Figure 39 on page 6-15. 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.

The user 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 (ports), interfaces (ifs), or virtual routers (vr). 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.

High Availability Configurations

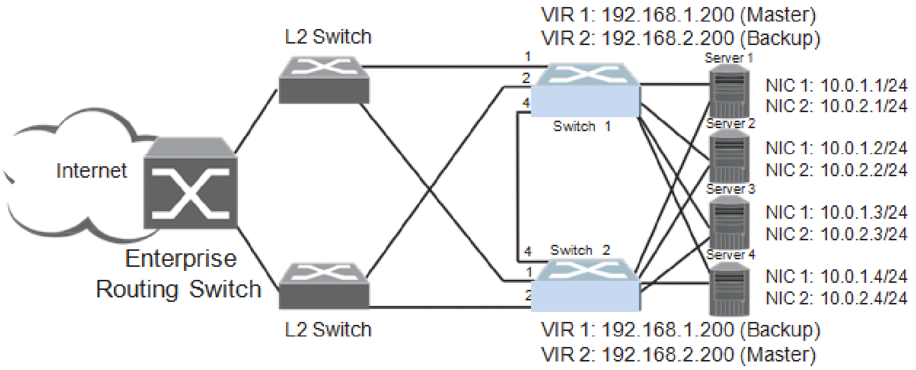
1/10Gb LAN Switch Modules offer flexibility in implementing redundant configurations. This section discusses the more useful and easily deployed configurations:

- "Active-Active Configuration" on page 6-19
- "Hot-Standby Configuration" on page 6-23

Active-Active Configuration

Figure 41 shows an example configuration where two 1/10Gb LAN Switch Modules are used as VRRP routers in an active-active configuration. In this configuration, both switches respond to packets.

Figure 41. Active-Active High-Availability Configuration



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 41, traffic destined for IPv4 address 10.0.1.1 is forwarded through the Layer 2 switch at the top of the drawing, and ingresses 1/10Gb LAN Switch Module 1 on port EXT1. Return traffic uses default gateway 1 (192.168.1.1).

If the link between 1/10Gb LAN Switch Module 1 and the Layer 2 switch fails, 1/10Gb LAN Switch Module 2 becomes the Master because it has a higher priority. Traffic is forwarded to 1/10Gb LAN Switch Module 2, which forwards it to 1/10Gb LAN Switch Module 1 through port EXT4. Return traffic uses default gateway 2 (192.168.2.1), 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 1/10Gb LAN Switch Module 1

1. Configure client and server interfaces.

```
Router(config)# interface ip 1
Router(config-ip-if)# ip address 192.168.1.100 255.255.255.0
Router(config-ip-if)# vlan 10
Router(config-ip-if)# enable
Router(config-ip-if)# exit
Router(config)# interface ip 2
Router(config-ip-if)# ip address 192.168.2.101 255.255.255.0
Router(config-ip-if)# vlan 20
Router(config-ip-if)# enable
Router(config-ip-if)# exit
Router(config)# interface ip 3
Router(config-ip-if)# ip address 10.0.1.100 255.255.255.0
Router(config-ip-if)# enable
Router(config-ip-if)# exit
Router(config)# interface ip 4
Router(config-ip-if)# ip address 10.0.2.101 255.255.255.0
Router(config-ip-if)# enable
Router(config-ip-if)# exit
```

2. Configure the default gateways. Each default gateway points to a Layer 3 router.

```
Router(config)# ip gateway 1 address 192.168.1.1
Router(config)# ip gateway 1 enable
Router(config)# ip gateway 2 address 192.168.2.1
Router(config)# ip gateway 2 enable
```

3. Turn on VRRP and configure two Virtual Interface Routers.

```
Router(config)# router vrrp
Router(config-vrrp)# enable
Router(config-vrrp)# virtual-router 1 virtual-router-id 1
Router(config-vrrp)# virtual-router 1 interface 1
Router(config-vrrp)# virtual-router 1 address 192.168.1.200
Router(config-vrrp)# virtual-router 1 enable
Router(config-vrrp)# virtual-router 2 virtual-router-id 2
Router(config-vrrp)# virtual-router 2 interface 2
Router(config-vrrp)# virtual-router 2 address 192.168.2.200
Router(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.

```
Router(config-vrrp)# virtual-router 1 track ports
Router(config-vrrp)# virtual-router 1 priority 101
Router(config-vrrp)# virtual-router 2 track ports
Router(config-vrrp)# exit
```

5. Configure ports.

```
Router(config)# vlan 10
Router(config-vlan)# exit
Router(config)# interface port EXT1
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk allowed vlan add 10
Router(config-if)# exit

Router(config)# vlan 20
Router(config-vlan)# exit
Router(config)# interface port EXT2
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk allowed vlan add 20
Router(config-if)# exit
```

6. Turn off Spanning Tree Protocol globally.

```
Router(config)# no spanning-tree stp 1
```

Task 2: Configure 1/10Gb LAN Switch Module 2

1. Configure client and server interfaces.

```
Router(config)# interface ip 1
Router(config-ip-if)# ip address 192.168.1.101 255.255.255.0
Router(config-ip-if)# vlan 10
Router(config-ip-if)# enable
Router(config-ip-if)# exit
Router(config)# interface ip 2
Router(config-ip-if)# ip address 192.168.2.100 255.255.255.0
Router(config-ip-if)# vlan 20
Router(config-ip-if)# enable
Router(config-ip-if)# exit
Router(config)# interface ip 3
Router(config-ip-if)# ip address 10.0.1.101 255.255.255.0
Router(config-ip-if)# enable
Router(config-ip-if)# exit
Router(config)# interface ip 4
Router(config-ip-if)# ip address 10.0.2.100 255.255.255.0
Router(config-ip-if)# enable
Router(config-ip-if)# exit
```

2. Configure the default gateways. Each default gateway points to a Layer 3 router.

```
Router(config)# ip gateway 1 address 192.168.2.1
Router(config)# ip gateway 1 enable
Router(config)# ip gateway 2 address 192.168.1.1
Router(config)# ip gateway 2 enable
```

3. Turn on VRRP and configure two Virtual Interface Routers.

```
Router(config)# router vrrp
Router(config-vrrp)# enable
Router(config-vrrp)# virtual-router 1 virtual-router-id 1
Router(config-vrrp)# virtual-router 1 interface 1
Router(config-vrrp)# virtual-router 1 address 192.168.1.200
Router(config-vrrp)# virtual-router 1 enable
Router(config-vrrp)# virtual-router 2 virtual-router-id 2
Router(config-vrrp)# virtual-router 2 interface 2
Router(config-vrrp)# virtual-router 2 address 192.168.2.200
Router(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.

```
Router(config-vrrp)# virtual-router 1 track ports
Router(config-vrrp)# virtual-router 2 track ports
Router(config-vrrp)# virtual-router 2 priority 101
Router(config-vrrp)# exit
```

5. Configure ports.

```
Router(config)# vlan 10
Router(config-vlan)# exit
Router(config)# interface port EXT1
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk allowed vlan add 10
Router(config-if)# exit

Router(config)# vlan 20
Router(config-vlan)# exit
Router(config)# interface port EXT2
Router(config-if)# switchport mode trunk
Router(config-if)# switchport trunk allowed vlan add 20
Router(config-if)# exit
```

6. Turn off Spanning Tree Protocol globally.

```
Router(config)# no spanning-tree stp 1
```

Hot-Standby Configuration

The primary application for VRRP-based hot-standby is to support Network Adapter Teaming on your server blades. With Network Adapter Teaming, the NICs on each server share the same IPv4 address, and are configured into a team. One NIC is the primary link, and the others are backup links. For more details, refer to the 10Gb Ethernet Adapter documentation.

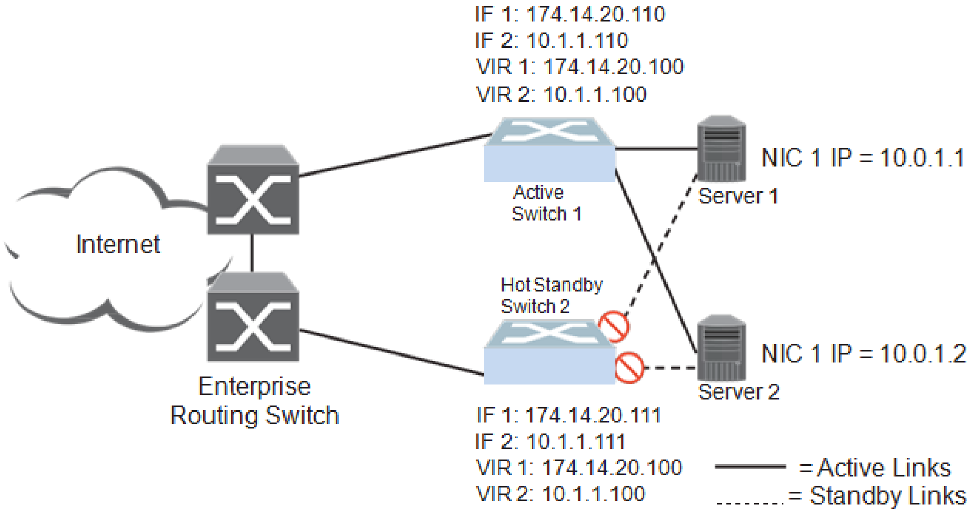
A hot-standby configuration allows all processes to failover to a standby switch if any type of failure should occur. All Virtual Interface Routers (VIRs) are bundled into one Virtual Router group, and then they failover together. When there is a failure that causes the VRRP Master to failover to the Standby, then the original primary switch temporarily disables the internal server links, which, in turn, causes the NIC teams to failover as well.

Note: When using hot-standby redundancy, peer switches should have an equal number of connected ports.

If hot-standby is implemented in a looped environment, the hot-standby feature automatically disables the hot-standby ports on the VRRP Standby. If the Master switch should failover to the Standby switch, it would change the hot-standby ports from *disabled* to *forwarding*, without relying on Spanning Tree or manual intervention. Therefore, Spanning Tree must be disabled.

Figure 42 illustrates a common hot-standby implementation on a single blade server. Notice that the blade server NICs are configured into a team that shares the same IPv4 address across both NICs. Because only one link can be active at a time, the hot-standby feature controls the NIC failover by having the Standby switch disable its internal ports (holding down the server links).

Figure 42. Hot-Standby Configuration



Task 1: Configure 1/10Gb LAN Switch Module 1

1. On 1/10Gb LAN Switch Module 1, configure the interfaces for clients (174.14.20.110) and servers (10.1.1.110).

```
Router(config)# interface ip 1
Router(config-ip-if)# ip address 174.14.20.110   (Define IPv4 address for
interface 1)
Router(config-ip-if)# enable
Router(config-ip-if)# exit
Router(config)# interface ip 2
Router(config-ip-if)# ip address 10.1.1.110      (Define IPv4 address for
interface 2)
Router(config-ip-if)# enable
Router(config-ip-if)# exit
```

2. Configure Virtual Interface Routers.

```
Router(config)# router vrrp
Router(config-vrrp)# enable
Router(config-vrrp)# virtual-router 1 virtual-router-id 1
Router(config-vrrp)# virtual-router 1 interface 1
Router(config-vrrp)# virtual-router 1 address 174.14.20.100
Router(config-vrrp)# virtual-router 1 enable
Router(config-vrrp)# virtual-router 2 virtual-router-id 2
Router(config-vrrp)# virtual-router 2 interface 2
Router(config-vrrp)# virtual-router 2 address 10.1.1.100
Router(config-vrrp)# virtual-router 2 enable
```

3. Enable VRRP Hot Standby.

```
Router(config-vrrp)# hot-standby    (Enable Hot Standby)
```

4. Configure VRRP Group parameters. Set the VRRP priority to 101, so that this switch is the Master.

```
Router(config-vrrp)# group enable                  (Enable Virtual Router Group)
Router(config-vrrp)# group virtual-router-id 1     (Set Virtual Router ID for Group)
Router(config-vrrp)# group interface 1             (Set interface for Group)
Router(config-vrrp)# group priority 101            (Set VRRP priority to 101)
Router(config-vrrp)# group track ports             (Enable tracking on ports)
```

5. Turn off Spanning Tree Protocol globally.

```
Router(config)# no spanning-tree stp 1
```

Task 2: Configure 1/10Gb LAN Switch Module 2

1. On 1/10Gb LAN Switch Module 2, configure the interfaces for clients (174.14.20.111) and servers (10.1.1.111).

```
Router(config)# interface ip 1
Router(config-ip-if)# ip address 174.14.20.111   (Define IPv4 address for
interface 1)
Router(config-ip-if)# enable
Router(config-ip-if)# exit
Router(config)# interface ip 2
Router(config-ip-if)# ip address 10.1.1.111      (Define IPv4 address for
interface 2)
Router(config-ip-if)# enable
Router(config-ip-if)# exit
```

2. Configure Virtual Interface Routers.

```
Router(config)# router vrrp
Router(config-vrrp)# enable
Router(config-vrrp)# virtual-router 1 virtual-router-id 1
Router(config-vrrp)# virtual-router 1 interface 1
Router(config-vrrp)# virtual-router 1 address 174.14.20.100
Router(config-vrrp)# virtual-router 1 enable
Router(config-vrrp)# virtual-router 2 virtual-router-id 2
Router(config-vrrp)# virtual-router 2 interface 2
Router(config-vrrp)# virtual-router 2 address 10.1.1.100
Router(config-vrrp)# virtual-router 2 enable
```

3. Enable VRRP Hot Standby.

```
Router(config-vrrp)# hot-standby
```

4. Configure VRRP Group parameters. Use the default VRRP priority of 100, so that this switch is the Standby.

```
Router(config-vrrp)# group enable                (Enable Virtual Router Group)
Router(config-vrrp)# group virtual-router id 1   (Set Virtual Router ID for Group)
Router(config-vrrp)# group interface 1           (Set interface for Group)
Router(config-vrrp)# group track ports           (Enable tracking on ports)
Router(config-vrrp)# group enable                (Enable Virtual Router Group)
```

5. Turn off Spanning Tree Protocol globally.

```
Router(config)# spanning-tree mode disable
```

7

Network-Management

This section discusses the Network-Management features.

□ Link Layer Discovery Protocol

□ Simple Network Management Protocol

Link Layer Discovery Protocol

The Networking OS software support Link Layer Discovery Protocol (LLDP). This chapter discusses the use and configuration of LLDP on the switch:

- "LLDP Overview" on page 7-2
- "Enabling or Disabling LLDP" on page 7-3
- "LLDP Transmit Features" on page 7-4
- "LLDP Receive Features" on page 7-8
- "LLDP Example Configuration" on page 7-11

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, 1/10Gb LAN Switch Module 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 1/10Gb LAN Switch Module 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 should be consistent in their LLDP configuration.

Enabling or Disabling LLDP

Global LLDP Setting

By default, LLDP is enabled on 1/10Gb LAN Switch Module. To turn LLDP off or on, use the following command:

```
Router(config)# [no] lldp enable      (Turn LLDP on or off globally)
```

Transmit and Receive Control

1/10Gb LAN Switch Module 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, 1/10Gb LAN Switch Module ports transmit and receive LLDP information (see the tx_rxoption below). To change the LLDP transmit and receive state, the following commands are available:

```
Router(config)# interface port <x>         (Select a switch port)
Router(config-if)# lldp admin-status tx_rx (Transmit and receive LLDP)
Router(config-if)# lldp admin-status tx_only (Only transmit LLDP)
Router(config-if)# lldp admin-status rx_only (Only receive LLDP)
Router(config-if)# no lldp admin-status    (Do not participate in LLDP)
Router(config-if)# exit                    (Exit port mode)
```

To view the LLDP transmit and receive status, use the following commands:

```
Router(config)# show lldp port              (status of all ports)
Router(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

1/10Gb LAN Switch Module 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:

```
Router(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 1/10Gb LAN Switch Module detects relevant changes to its configuration or status (such as when ports are enabled or disabled). To prevent 1/10Gb LAN Switch Module 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:

```
Router(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 (lldprefresh-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 should be held before it expires. The hold time is configured as a multiple of the configured transmission interval.

```
Router(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 msgtxintvalue, or 120 seconds, before removing it from their MIB.

Trap Notifications

If SNMP is enabled on 1/10Gb LAN Switch Module (see "Using Simple Network Management Protocol" on page 1-9), 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:

```
Router(config)# interface port <x>
Router(config-if)# [no] lldp trap-notification
Router(config-if)# exit
```

In addition to sending LLDP information at scheduled intervals, LLDP information is also sent when 1/10Gb LAN Switch Module detects relevant changes to its configuration or status (such as when ports are enabled or disabled). To prevent 1/10Gb LAN Switch Module 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:

```
Router(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 commands:

```
Router(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 admstat command’s rx_only or disabled options (see “Transmit and Receive Control” on page 7-3), 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 1/10Gb LAN Switch Module port from their MIB.

In addition, if LLDP is fully disabled on a port (using admstat disabled) and later re-enabled, 1/10Gb LAN Switch Module will temporarily delay resuming LLDP transmissions on the port in order to allow the port LLDP information to stabilize. The reinitialization delay interval can be globally configured for all ports using the following command:

```
Router(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 7-3), 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:

```
Router(config)# interface port <x>
Router(config-if)# [no] lldp tlv <type>
Router(config-if)# exit
```

where *type* is an LLDP information option from Table 26:

Table 26. 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
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 7-3), 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

1/10Gb LAN Switch Module 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 1/10Gb LAN Switch Module MIB
- Using 1/10Gb LAN Switch Module Browser-Based Interface (BBI)
- Using CLI commands on 1/10Gb LAN Switch Module

Using the CLI the following command displays remote LLDP information:

```
Router(config)# show lldp remote-device [<index number>]
```

To view a summary of remote information, omit the *Index number* parameter. For example:

```
Router(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: 1
LocalPort |Index |Remote Chassis ID |Remote Port |Remote System Name|DMAC

----------|------|------------------|------------|------------------|----
EXT3      | 1    |00 18 b1 33 1d 00 |   23       |    C12           | 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:

```
Router(config)# show lldp remote-device 1
Local Port Alias: EXT3
       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    : EXT7

       System Name   :
       System Description : 1/10Gb LAN Switch Module, Networking OS:
       version 7.8, boot image: version 6.9.1.14

       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:

```
Router(config)# show lldp remote-device detail
Local Port Alias: EXT22
       Remote Device Index : 1
       Rmote Device TTL    : 94
       Remote Device RxChanges : false Chassis Type
                           : Mac Address
       Chassis Id: 74-99-75-74-c5-00
       Port Type     : Locally Assigned
       Port Id       : 42
       Port Description    : 42

       System Name   : GFC
       System Description : Networking Operating System Switch,
Networking OS: version 7.8.0.43, Boot image: version 7.8.0.43
       System Capabilities Supported : bridge, router
       System Capabilities Enabled      : bridge, router

       Remote Management Address:
              Subtype       : IPv4
              Address       : 11.1.58.5
              Interface Subtype   : ifIndex
              Interface Number    : 58
              Object Identifier   :

Local Port Alias: EXT24
       Remote Device Index : 2
       Remote Device TTL   : 108
       Remote Device RxChanges : false Chassis Type
                               : Mac Address
       Chassis Id    : 74-99-75-1c-71-00
       Port Type     : Locally Assigned
       Port Id       : 56
       Port Description    : EXT14

       System Name   : CFC
       System Description : 1/10Gb LAN Switch Module for Hitachi
BladeSymphony, Networking OS: version 7.8.0.48, Boot image: version
7.8.0.48
       System Capabilities Supported : bridge, router
       System Capabilities Enabled      : bridge, router

       Remote Management Address:
              Subtype       : IPv4
              Address       : 11.1.78.7
              Interface Subtype   : ifIndex
              Interface Number    : 78
              Object Identifier   :
```

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 should be immediately removed.

LLDP Example Configuration

1. Turn LLDP on globally.

```
Router(config)# lldp enable
```

2. Set the global LLDP timer features.

```
Router(config)# lldp refresh-interval 30       (Transmit each 30 seconds)
Router(config)# lldp transmission-delay 2      (No more often than 2 sec.)
Router(config)# lldp holdtime-multiplier 4     (Remote hold 4 intervals)
Router(config)# lldp reinit-delay 2            (Wait 2 sec. after reinit.)
Router(config)# lldp trap-notification-interval  5     (Minimum 5 sec. between)
```

3. Set LLDP options for each port.

```
Router(config)# interface port <n>    (Select a switch port)
Router(config-if)# lldp admin-status tx_rx  (Transmit and receive LLDP)
Router(config-if)# lldp trap-notification   (Enable SNMP trap notifications)
Router(config-if)# lldp tlv all       (Transmit all optional information)
Router(config-if)# exit
```

4. Enable syslog reporting.

```
Router(config)# logging log lldp
```

5. Verify the configuration settings:

```
Router(config)# show lldp
```

6. View remote device information as needed.

```
Router(config)# show lldp remote-device
                Or
Router(config)# show lldp remote-device <index number>
                or
Router(config)# show lldp remote-devices detail
```

Simple Network Management Protocol

Networking OS provides Simple Network Management Protocol (SNMP) version 1, version 2, and version 3 support for access through any network management software.

SNMP Version 1

To access the SNMP agent on 1/10Gb LAN Switch Module, the read and write community strings on the SNMP manager should be configured to match those on the switch.

SNMPv1 and SNMPv2 have no default read and write communities. The read and write community strings on the switch can be configured using the following commands:

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

The SNMP manager should 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:

```
Router(config)# snmp-server trap-source <trap source IP interface>
Router(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:

```
Router(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:

```
Router(config)# snmp-server ?
```

For more information on SNMP MIBs and the commands used to configure SNMP on the switch, see the *Networking OS 7.8 Command Reference*.

Default Configuration

Networking OS has four SNMPv3 users by default. All the four 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). Authenti- cation used is SHA. Privacy protocol used is AES-128.

In boot strict mode (See “Boot Strict Mode” on page 1-13), Networking OS has only one SNMPv3 user:

- User 1 name is adminshaaes(password Edpq132x!#9Zpx432w). Authenti- cation 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:

```
Router(config)# snmp-server user <1-16> name <1-32 characters>
```

Users can be configured to use the authentication/privacy options. 1/10Gb LAN Switch Module support two authentication algorithms: MD5 and SHA, as specified in the following command:

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.

```
Router(config)# snmp-server user 5 name admin
Router(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>
Re-enter new authentication password:   <auth. password>
New authentication password accepted.

Router(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>
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.

```
Router(config)# snmp-server access 5 name admingrp
Router(config)# snmp-server access 5 level authpriv
Router(config)# snmp-server access 5 read-view iso
Router(config)# snmp-server access 5 write-view iso
Router(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.

```
Router(config)# snmp-server group 5 user-name admin
Router(config)# snmp-server group 5 group-name admingrp
```

If you want to allow user access only to certain MIBs, see “View-Based Configuration,” next.

View-Based Configurations

- Switch User equivalent

To configure an SNMP user equivalent to the switch “user” login, use the following configuration:

```
Router(config)# snmp-server user 4 name usr

Router(config)# snmp-server access 3 name usrgrp
Router(config)# snmp-server access 3 read-view usr
Router(config)# snmp-server access 3 write-view usr
Router(config)# snmp-server access 3 notify-view usr

Router(config)# snmp-server group 3 user-name usr
Router(config)# snmp-server group 3 group-name usrgrp

Router(config)# snmp-server view 6 name usr

(Configure the user) (Configure access group 3)

(Assign user to access group 3) (Create views for user)

Router(config)# snmp-server view 6 tree 1.3.6.1.4.1.116.5.52.20.2.5.1.2
(Agent information)
Router(config)# snmp-server view 7 name usr
Router(config)# snmp-server view 7 tree 1.3.6.1.4.1.116.5.52.20.2.5.1.3
(L2 statistics)
Router(config)# snmp-server view 8 name usr
Router(config)# snmp-server view 8 tree 1.3.6.1.4.1.116.5.52.20.2.5.2.2
(L2 information)
Router(config)# snmp-server view 9 name usr
Router(config)# snmp-server view 9 tree 1.3.6.1.4.1.116.5.52.20.2.5.2.3
(L3 statistics)
Router(config)# snmp-server view 10 name usr
Router(config)# snmp-server view 10 tree 1.3.6.1.4.1.116.5.52.20.2.5.2.3
(L3 information)
Router(config)# snmp-server view 11 name usr
Router(config)# snmp-server view 11 tree 1.3.6.1.4.1.116.5.52.20.2.5.3.3
```

- Switch Oper equivalent

```
Router(config)# snmp-server user 5 name usr

Router(config)# snmp-server access 4 name opergrp
Router(config)# snmp-server access 4 read-view oper
Router(config)# snmp-server access 4 write-view oper
Router(config)# snmp-server access 4 notify-view oper

Router(config)# snmp-server group 4 user-name oper
Router(config)# snmp-server group 4 group-name opergrp

Router(config)# snmp-server view 20 name oper

(Configure the user) (Configure access group 3)

(Assign oper to access group 4) (Create views for oper)

Router(config)# snmp-server view 20 tree 1.3.6.1.4.1.116.5.52.20.2.5.1.2
(Agent information)
Router(config)# snmp-server view 21 name oper
Router(config)# snmp-server view 21 tree 1.3.6.1.4.1.116.5.52.20.2.5.1.3
(L2 statistics)
Router(config)# snmp-server view 22 name oper
Router(config)# snmp-server view 22 tree 1.3.6.1.4.1.116.5.52.20.2.5.2.2
(L2 information)
Router(config)# snmp-server view 23 name oper
Router(config)# snmp-server view 23 tree 1.3.6.1.4.1.116.5.52.20.2.5.2.3
(L3 statistics)
Router(config)# snmp-server view 24 name oper
Router(config)# snmp-server view 24 tree 1.3.6.1.4.1.116.5.52.20.2.5.2.3
(L3 information)
Router(config)# snmp-server view 25 name oper
Router(config)# snmp-server view 25 tree 1.3.6.1.4.1.116.5.52.20.2.5.3.3
```

Configuring SNMP Trap Hosts

SNMPv1 Trap Host

1. Configure a user with no authentication and password.

```
Router(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:

```
Router(config)# snmp-server access <user number>
```

In the following example the user will receive the traps sent by the switch.

```
Router(config)# snmp-server access 10       (Access group to view SNMPv1 traps)
    name v1trap
    security
    snmpv1 notify-
    view iso
Router(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.

```
Router(config)# snmp-server notify 10 name v1trap
Router(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:

```
Router(config)# snmp-server target-address 10 name v1trap address
10.70.70.190
Router(config)# snmp-server target-address 10 parameters-name v1param
Router(config)# snmp-server target-address 10 taglist v1param
Router(config)# snmp-server target-parameters 10 name v1param
Router(config)# snmp-server target-parameters 10 user-name v1only
Router(config)# snmp-server target-parameters 10 message snmpv1
```

Note: Networking OS 7.8 supports only IPv4 addresses for SNMP trap hosts.

5. Use the community table to specify which community string is used in the trap.

```
Router(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.

```
Router(config)# snmp-server user 10 name v2trap

Router(config)# snmp-server group 10 security snmpv2
Router(config)# snmp-server group 10 user-name v2trap
Router(config)# snmp-server group 10 group-name v2trap
Router(config)# snmp-server access 10 name v2trap
Router(config)# snmp-server access 10 security snmpv2
Router(config)# snmp-server access 10 notify-view iso

Router(config)# snmp-server notify 10 name v2trap
Router(config)# snmp-server notify 10 tag v2trap

Router(config)# snmp-server target-address 10 name v2trap address 100.10.2.1
Router(config)# snmp-server target-address 10 taglist v2trap
Router(config)# snmp-server target-address 10 parameters-name v2param
Router(config)# snmp-server target-parameters 10 name v2param
Router(config)# snmp-server target-parameters 10 message snmpv2c
Router(config)# snmp-server target-parameters 10 user-name v2trap
Router(config)# snmp-server target-parameters 10 security snmpv2

Router(config)# snmp-server community 10 index v2trap
Router(config)# snmp-server community 10 user-name v2trap
```

Note: Networking OS 7.8 supports only IPv4 addresses for SNMPv1 and SNMP v2 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:

```
Router(config)# snmp-server access <1-32> level
Router(config)# snmp-server target-parameters <1-16>
```

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 a SNMPv3 user v3trap with authentication only:

```
Router(config)# snmp-server user 11 name v3trap
Router(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.
Router(config)# snmp-server access 11 notify-view iso
Router(config)# snmp-server access 11 level authnopriv
Router(config)# snmp-server group 11 user-name v3trap
Router(config)# snmp-server group 11 tag v3trap
Router(config)# snmp-server notify 11 name v3trap
Router(config)# snmp-server notify 11 tag v3trap
Router(config)# snmp-server target-address 11 name v3trap address 47.81.25.66
Router(config)# snmp-server target-address 11 taglist v3trap
Router(config)# snmp-server target-address 11 parameters-name v3param
Router(config)# snmp-server target-parameters 11 name v3param
Router(config)# snmp-server target-parameters 11 user-name v3trap
Router(config)# snmp-server target-parameters 11 level authNoPriv
```

SNMP MIBs

The Networking OS SNMP agent supports SNMP version 3. Security is provided through SNMP community strings. The default community strings are "public" for SNMPGEToperation and "private" for SNMPSEToperation. The community string can be modified only through the Command Line Interface (CLI). Detailed SNMP MIBs and trap definitions of the Networking OS SNMP agent are contained in the following Networking OS enterprise MIB document:

`gbiosw-1-10G-L2L3.mib`The Networking OS SNMP agent supports the following standard MIBs:

- `dot1x.mib`
- `ieee8021ab.mib`
- `ieee8023ad.mib`
- `lldpxdcbx.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`
- `rfc3176.mib`

The Networking 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 Networking OS:

Table 27.   Networking OS-Supported Enterprise SNMP Traps

Trap Name	Description
altSwLoginFailure	Signifies that someone failed to enter a valid username/password combination. altSwTrapDisplayString 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.
altSwValidLogin	Signifies that a user login has occurred.
altSwApplyComplete	Signifies that new configuration has been applied.
altSwSaveComplete	Signifies that new configuration has been saved.
altSwFwDownloadSucess	Signifies that firmware has been downloaded to [image1|image2|boot image].
altSwFwDownloadFailure	Signifies that firmware downloaded failed to [image1|image2|boot image].
altSwValidLogout	Signifies that a user logout has occurred.
altSwDefAdminDisable	Signifies that the default admin account has been disabled.
altSwAcntStrngPswdNotMet	Signifies that the configured password does not match strong password complexity.
altSwAcntLocked	Signifies that account has been locked.
altSwAcntUnlocked	Signifies that account has been unlocked.
altSwStgNewRoot	Signifies that the bridge has become the new root of the STG.
altSwCistNewRoot	Signifies that the bridge has become the new root of the CIST.
altSwStgTopologyChanged	Signifies that there was a STG topology change.

Table 27. Networking OS-Supported Enterprise SNMP Traps (continued)

Trap Name	Description
altSwCistTopologyChanged	Signifies that there was a CIST topology change.
altSwHotlinksMasterUp	Signifies that the Master interface is active.
altSwHotlinksMasterDn	Signifies that the Master interface is not active.
altSwHotlinksBackupUp	Signifies that the Backup interface is active.
altSwHotlinksBackupDn	Signifies that the Backup interface is not active.
altSwHotlinksNone	Signifies that there are no active interfaces.
altSwStgBlockingState	Signifies port state has changed to blocking state.
altSwTeamingCtrlUp	Signifies that the teaming is up.
altSwTeamingCtrlDown	Signifies that the teaming control is down.
altSwTeamingCtrlDownTearDownBlked	Signifies that the teaming control is down but teardown is blocked.
altSwTeamingCtrlError	Signifies error, action is undefined.
altSwLACPPortBlocked	Signifies that LACP is operationally down on a port, and traffic is blocked on the port.
altSwLACPPortUnblocked	Signifies that LACP is operationally up on a port, and traffic is no longer blocked on the port.
altSwLFDPortErrdisabled	Signifies that a port is error-disabled due to excessive link flaps.
altSwDefGwUp	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.

Table 27. Networking OS-Supported Enterprise SNMP Traps (continued)

Trap Name	Description
altSwDefGwDown	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.
altSwDefGwInService	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.
altSwDefGwNotInService	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.
altSwVrrpNewMaster	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.
altSwVrrpNewBackup	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.

Table 27. Networking OS-Supported Enterprise SNMP Traps (continued)

Trap Name	Description
altSwVrrpAuthFailure	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. Implementation of this trap is optional. 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.
altSwNtpNotServer	Signifies that the primary or secondary NTP server cannot be reached.
altSwNTPUpdateClock	Signifies that the system clock is updated with NTP server.
altSwECMPGatewayUp	Signifies that the ECMP gateway is up.
altSwECMPGatewayDown	Signifies that the ECMP gateway is down.
altSwTempExceedThreshold	Signifies that the switch temperature has exceeded maximum safety limits.
altSwTempReturnThreshold	Signifies that the switch temperature has returned to under maximum safety limits.
altVMGroupVMotion	Signifies that a virtual machine has moved from a port to another.
altVMGroupVMOnline	Signifies that an advance provisioned virtual machine has came online.
altVMGroupVMVlanChange	Signifies that a virtual machine has entered a VLAN, or changed the VLAN.
vmCheckSpoofedvm	Signifies that a spoofed VM MAC was found.

8

Monitoring

The ability to monitor traffic passing through 1/10Gb LAN Switch Module can be invaluable for troubleshooting some types of networking problems. This section cover the monitoring features.

- □ Remote Monitoring
- □ sFLOW
- □ Port Mirroring

Remote Monitoring

Remote Monitoring (RMON) allows network devices to exchange network monitoring data.

RMON performs the following major functions:

- Gathers cumulative statistics for Ethernet interfaces
- Tracks a history of statistics for Ethernet interfaces
- Creates and triggers alarms for user-defined events

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:

- RMON Group 1–Statistics
- RMON Group 2–History
- RMON Group 3–Alarms
- RMON 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. 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.

To configure RMON Statistics:

1. Enable RMON on each port where you wish to collect RMON statistics.

```
Router(config)# interface port 23
Router(config-if)# rmon
```

2. View RMON statistics for the port.

```
Router(config-if)#  show interface port 23 rmon-counters
------------------------------------------------------------------
RMON statistics for port 23:
 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.
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 instance 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.

Use an SNMP browser to view History samples.

History MIB Objects

The type of data that can be sampled must be of an ifIndexobject type, as described in RFC1213 and RFC1573. The most common data type for the history sample is as follows:

```
1.3.6.1.2.1.2.2.1.1.<x>
-mgmt.interfaces.ifTable.ifIndex.interface
```

The last digit (*x*) represents the interface on which to monitor, which corresponds to the switch port number. History sampling is done per port, by utilizing the interface number to specify the port number.

Configuring RMON History

This example configuration creates an RMON History Group to monitor port 1. It takes a data sample every two minutes, and places the data into one of the 30 requested buckets. After 30 samples are gathered, the new samples overwrite the previous samples, beginning with the first bucket.

1. Enable RMON on each port where you wish to collect RMON History.

```
Router(config)# interface port 1
Router(config-if)# rmon
Router(config-if)# exit
```

2. Configure the RMON History parameters.

```
Router(config)#  rmon history 1 interface-oid 1.3.6.1.2.1.2.2.1.1.<x>
Router(config)#  rmon history 1 requested-buckets 30
Router(config)#  rmon history 1 polling-interval 120
Router(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: `1.3.6.1.2.1.2.2.1.1.1`

3. View RMON history for the port.

```
Router(config)# show rmon history
RMON History group configuration:
 Index         IFOID            Interval   Rbnum   Gbnum
 -----   ----------------------   --------   -----   -----
    1    1.3.6.1.2.1.2.2.1.1.1       120       30      30
Router(config)# show rmon history
RMON History group configuration:
 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:

```
Router(config)#  rmon alarm <alarm number> rising-crossing-index  <event number>
Router(config)#  rmon alarm <alarm number> falling-crossing-index  <event number>
```

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 of an ICMP stat is as follows:

```
1.3.6.1.2.1.5.1.<x> - mgmt.icmp.icmpInMsgs
```

where *x* represents the interface on which to monitor, which corresponds to the switch interface number or port number, as follows:

- 1 through 128 = Switch interface number
- 129 = Switch port 1
- 130 = Switch port 2
- 131 = Switch port 3, and so on.

This value represents the alarm's MIB OID, as a string. Note that for non-tables, you must supply a .0to specify an end node.

Configuring RMON Alarms

Alarm Example 1

This example configuration creates an RMON alarm that checks ifInOctets on port 20 once every hour. If the statistic exceeds two billion, an alarm is generated that triggers event index 6.

1. Configure the RMON Alarm parameters to track the number of packets received on a port.

```
Router(config)#  rmon alarm 1 oid 1.3.6.1.2.1.2.2.1.10.129
Router(config)#  rmon alarm 1 alarm-type rising
Router(config)#  rmon alarm 1 rising-crossing-index 100
Router(config)#  rmon alarm 1 interval 3600
Router(config)#  rmon alarm 1 rising-limit 2000000000
Router(config)#  rmon alarm 1 owner "Alarm for ifInOctets"
```

Alarm Example 2

This example configuration creates an RMON alarm that checks icmpInEchoson the switch once every minute. If the statistic exceeds 200 within a 60 second interval, an alarm is generated that triggers event index 5.

Configure the RMON Alarm parameters to track ICMP messages.

```
Router(config)#  rmon alarm 1 oid 1.3.6.1.2.1.5.8.0
Router(config)#  rmon alarm 1 alarm-type rising
Router(config)#  rmon alarm 1 rising-crossing-index 110
Router(config)#  rmon alarm 1 interval-time 60
Router(config)#  rmon alarm 1 rising-limit 200
Router(config)#  rmon alarm 1 sample delta
Router(config)#  rmon alarm 1 owner "Alarm for icmpInEchos"
```

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 message, 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:

```
Router(config)#  rmon alarm <alarm number> rising-crossing-index  <event number>
Router(config)#  rmon alarm <alarm number> falling-crossing-index  <event number>
```

RMON events use SNMP and system logs 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 system log message of type `RMON` that corresponds to the event.

For example, to configure the RMON event parameters.

```
Router(config)# rmon event 110 type log
Router(config)# rmon event 110 description "SYSLOG_this_alarm"
Router(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.

sFLOW

1/10Gb LAN Switch Module supports sFlow technology for monitoring traffic in data networks. The switch includes an embedded sFlow agent which can be configured to sample network traffic and 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 in order to interpret sFlow data.

Note: Networking OS 7.8 does not support IPv6 for sFLOW.

sFlow Statistical Counters

1/10Gb LAN Switch Module 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, 1/10Gb LAN Switch Module reports general port statistics and port Ethernet statistics.

sFlow Network Sampling

In addition to statistical counters, 1/10Gb LAN Switch Module 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.

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):

```
Router(config)# sflow server <IPv4 address>     (sFlow server address)
Router(config)# sflow port <service port>       (Set the optional service port)
Router(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 following command:

```
Router(config)# no sflow enable
```

2. On a per-port basis, define the statistics polling rate:

```
Router(config)# interface port <port>
Router(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:

```
Router(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.

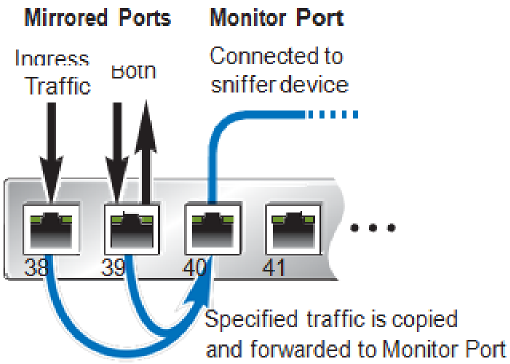
4. Save the configuration.

Port Mirroring

The Networking OS 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 in order to detect intruders attacking the network.

The Router supports a “many to one” mirroring model. As shown in Figure 43, selected traffic for ports EXT1 and EXT2 is being monitored by port EXT3. In the example, both ingress traffic and egress traffic on port EXT2 are copied and forwarded to the monitor. However, port EXT1 mirroring is configured so that only ingress traffic is copied and forwarded to the monitor. A device attached to port EXT3 can analyze the resulting mirrored traffic.

Figure 43. Mirroring Ports


Mirrored Ports
Monitor Port
Ingress Traffic
Both
Connected to sniffer device
38
39
40
41
Specified traffic is copied and forwarded to Monitor Port

1/10Gb LAN Switch Module supports two monitor ports with two-way mirroring, or four monitor ports with one-way mirroring. 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 EXT1 traffic cannot be monitored by both port EXT3 and EXT4 at the same time, nor can port EXT2 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.
Note: 1/10Gb LAN Switch Module cannot mirror LACPDU packets. Also, traffic on management VLANs is not mirrored to the external ports.

Port Mirroring Behavior

This section describes the composition of monitored packets in 1/10Gb LAN Switch Module, based on the configuration of the ports.

- Packets mirrored at port egress are mirrored prior to VLAN tag processing and may have a different PVID than packets that egress the port toward their actual network destination.
- Packets mirrored at port ingress are not modified.

Configuring Port Mirroring

The following procedure may be used to configure port mirroring for the example shown in Figure 43 on page 8-10:

1. Specify the monitoring port, the mirroring port(s), and the port-mirror direction.

```
Router(config)# port-mirroring monitor-port EXT3 mirroring-port EXT1 in
Router(config)# port-mirroring monitor-port EXT3 mirroring-port EXT2 both
```

2. Enable port mirroring.

```
Router(config)# port-mirroring enable
```

3. View the current configuration.

```
Router# show port-mirroring          (Display the current settings)
Port mirroring is enabled
Monitoring Ports Mirrored Ports
INTA1             none
INTA2
none               INTA3
none               INTA4
none
...
EXT1              none
EXT2              none
EXT3              EXT1, in
                   EXT2, both
EXT4              none
...
```

Glossary

This glossary defines the special terms used in this document. Click the desired letter below to display the glossary entries that start with that letter.

#	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

D

DIP

The destination IP address of a frame.

Dport

The destination port (application socket: for example, http-80/https-443/DNS-53)

H

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).

N

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.

P

Preemption

In VRRP, preemption will cause a Virtual Router that has a lower priority to go into backup should a peer Virtual Router start 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.)

S

SIP

The source IP address of a frame.

SPort

The source port (application socket: for example, HTTP-80/HTTPS-443/DNS-53).

T

Tracking

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.

You can track the following:

- 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

V

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 1/10Gb LAN Switch Modules 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

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*>.

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.

VRRP

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.

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. Should the master stop 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.