S-Series® Configuration Guide

Firmware Version 8.31

Published September 2014

9034730-03

Legal Notices

Extreme Networks, Inc., on behalf of or through its wholly-owned subsidiary, Enterasys Networks, Inc., reserves the right to make changes in specifications and other information contained in this document and its website without prior notice. The reader should in all cases consult representatives of Extreme Networks to determine whether any such changes have been made.

The hardware, firmware, software or any specifications described or referred to in this document are subject to change without notice.

Trademarks

Extreme Networks and the Extreme Networks logo are trademarks or registered trademarks of Extreme Networks, Inc. in the United States and/or other countries.

All other names (including any product names) mentioned in this document are the property of their respective owners and may be trademarks or registered trademarks of their respective companies/owners.

For additional information about Extreme Networks trademarks, go to:
www.extremenetworks.com/company/legal/trademarks/

Support

For product support, including documentation, visit:
http://support.extremenetworks.com/

Contact

Extreme Networks, Inc.
145 Rio Robles
San Jose, CA 19534
Tel: +1 408-579-2800

Toll-free: +1 888-257-3000

Enterasys Networks, Inc. Firmware License Agreement

BEFORE OPENING OR UTILIZING THE ENCLOSED PRODUCT, CAREFULLY READ THIS LICENSE AGREEMENT.

This document is an agreement ("Agreement") between the end user ("You") and Enterasys Networks, Inc., a wholly-owned subsidiary of Extreme Networks, Inc., on behalf of itself and its Affiliates (as hereinafter defined) ("Enterasys"), that sets forth Your rights and obligations with respect to the Enterasys software program/firmware (including any accompanying documentation, hardware or media) (collectively, the "Program") in the package and prevails over any additional, conflicting or inconsistent terms and conditions appearing on any purchase order or other document submitted by You. "Affiliate" means any person, partnership, corporation, limited liability company, other form of enterprise that directly or indirectly through one or more intermediaries, controls, or is controlled by, or is under common control with the party specified.

This Agreement constitutes the entire understanding between the parties, with respect to the subject matter of this Agreement. The Program may be contained in firmware, chips or other media.

BY INSTALLING OR OTHERWISE USING THE PROGRAM, YOU REPRESENT THAT YOU ARE AUTHORIZED TO ACCEPT THESE TERMS ON BEHALF OF THE END USER (IF THE END USER IS AN ENTITY ON WHOSE BEHALF YOU ARE AUTHORIZED TO ACT, “YOU” AND “YOUR” SHALL BE DEEMED TO REFER TO SUCH ENTITY) AND THAT YOU AGREE THAT YOU ARE BOUND BY THE TERMS OF THIS AGREEMENT, WHICH INCLUDES, AMONG OTHER PROVISIONS, THE LICENSE, THE DISCLAIMER OF WARRANTY AND THE LIMITATION OF LIABILITY. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT OR ARE NOT AUTHORIZED TO ENTER INTO THIS AGREEMENT, ENTERASYS IS UNWILLING TO LICENSE THE PROGRAM TO YOU AND YOU AGREE TO RETURN THE UNOPENED PRODUCT TO ENTERASYS OR YOUR DEALER, IF ANY, WITHIN TEN (10) DAYS FOLLOWING THE DATE OF RECEIPT FOR A FULL REFUND.

IF YOU HAVE ANY QUESTIONS ABOUT THIS AGREEMENT, CONTACT EXTREME NETWORKS, LEGAL DEPARTMENT AT (408) 579-2800.

You and Enterasys agree as follows:

1. LICENSE. You are granted a revocable, non-exclusive and non-transferable right to use only one (1) copy of the Program provided in this package subject to the terms and conditions of this Agreement.
2. RESTRICTIONS. Except as otherwise authorized in writing by Enterasys, You may not, nor may You permit any third party to:
   (a) Reverse engineer, decompile, disassemble or modify the Program, in whole or in part, including for reasons of error correction or interoperability, except to the extent expressly permitted by applicable law and to the extent the parties shall not be permitted by that applicable law, such rights are expressly excluded. Information necessary to achieve interoperability or correct errors is available from Enterasys upon request and payment of Enterasys’ applicable fee.
   (b) Incorporate the Program in whole or in part, in any other product or create derivative works based on the Program, in whole or in part.
   (c) Publish, disclose, copy reproduce or transmit the Program, in whole or in part.
   (d) Assign, sell, license, sublicense, rent, lease, encumber by way of security interest, pledge or otherwise transfer the Program, in whole or in part.
   (e) Remove any copyright, trademark, proprietary rights, disclaimer or warning notice included on or embedded in any part of the Program.

3. APPLICABLE LAW. This Agreement shall be interpreted and governed under the laws and in the state and federal courts of the State of California without regard to its conflicts of laws provisions. You accept the personal jurisdiction and venue of the Superior Court of California in Santa Clara County or the United States District Court for the Northern District of California in San Jose, California. None of the 1980 United Nations Convention on the Limitation Period in the International Sale of Goods, and the Uniform Computer Information Transactions Act shall apply to this Agreement.
4. EXPORT RESTRICTIONS. You acknowledge and agree that the Program and its accompanying materials/documentation are subject to the export control laws and regulations of the United States, including but not limited to the Export Administration Regulations (EAR), the International Traffic in Arms Regulations (ITAR), and the sanction regimes of the U.S. Department of Treasury, Office of Foreign Assets Control's Foreign Assets Control Regulations (FACR). You agree that You will comply with these laws and regulations.

   You agree that You will not, without prior U.S. Government authorization, export, reexport, or transfer the Program, either directly or indirectly, to any country subject to a U.S. trade embargo or sanction (e.g. Cuba, N. Korea, Iran, Syria, Sudan) or to any resident or national of said countries, or to any person, organization, or entity on any of the restricted parties lists maintained by the U.S. Departments of State, Treasury, or Commerce. In addition, You agree that You will not export, reexport or transfer the Program to any end-user engaged in activities, or for any end-use, directly or indirectly related to the design, development, production, use, or stockpiling of weapons of mass destruction, e.g. nuclear, chemical, or biological weapons, and the missile technology to deliver them.
5. UNITED STATES GOVERNMENT RESTRICTED RIGHTS. The enclosed Program (i) was developed solely at private expense; (ii) contains "restricted computer software" submitted with restricted rights in accordance with section 52.227-19 (a) through (d) of the Commercial Computer Software-Restricted Rights Clause and its successors, and (iii) in all respects is proprietary data belonging to Enterasys, its Affiliates and/or its suppliers. For Department of Defense units, the Program is considered commercial computer software in accordance with DFARS section 227.7202-3 and its successors, and use, duplication, or disclosure by the U.S. Government is subject to restrictions set forth herein.
6. DISCLAIMER OF WARRANTY. EXCEPT FOR THOSE WARRANTIES EXPRESSLY PROVIDED TO YOU IN WRITING BY ENTERASYS, ENTERASYS DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT WITH RESPECT TO THE PROGRAM. IF THE IMPLIED WARRANTIES MAY NOT BE DISCLAIMED BY APPLICABLE LAW, THEN ANY IMPLIED WARRANTIES ARE LIMITED IN DURATION TO THIRTY (30) DAYS AFTER DELIVERY OF THE PROGRAM TO YOU.
7. LIMITATION OF LIABILITY. IN NO EVENT SHALL ENTERASYS OR ITS AFFILIATES AND SUPPLIERS BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING, WITHOUT LIMITATION, DAMAGES FOR LOSS OF BUSINESS, PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, SPECIAL, INCIDENTAL, CONSEQUENTIAL, OR RELIANCE DAMAGES, OR OTHER LOSS) ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM, EVEN IF ENTERASYS HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. THIS FOREGOING LIMITATION SHALL APPLY REGARDLESS OF THE CAUSE OF ACTION UNDER WHICH DAMAGES ARE SOUGHT.

NOTWITHSTANDING THE FOREGOING, THE CUMULATIVE LIABILITY OF ENTERASYS TO YOU FOR ALL CLAIMS RELATING TO THE PROGRAM, IN CONTRACT,

TORT OR OTHERWISE, SHALL NOT EXCEED THE TOTAL AMOUNT OF FEES PAID TO ENTERASYS BY YOU FOR THE RIGHTS GRANTED HEREIN.

8. AUDIT RIGHTS. You hereby acknowledge that the intellectual property rights associated with the Program are of critical value to Enterasys and its Affiliates, and, accordingly, You hereby agree to maintain complete books, records and accounts showing: (i) license fees due and paid, and (ii) the use, copying and deployment of the Program. You also grant to Enterasys and its authorized representatives, upon reasonable notice, the right to audit and examine during Your normal business hours, Your books, records, accounts and hardware devices upon which the Program may be deployed to verify compliance with this Agreement, including the verification of the license fees due and paid to Enterasys and the use, copying and deployment of the Program. Enterasys' right of examination shall be exercised reasonably, in good faith and in a manner calculated to not unreasonably interfere with Your business. In the event such an audit discovers any non-compliance with this Agreement, including copies of the Program made, used or deployed in breach of this Agreement, You shall promptly cease such unauthorized conduct, pay to Enterasys the appropriate license fees and be subject to any other available claim from Enterasys pursuant to applicable law. Enterasys reserves the right, to be exercised in its sole discretion and without prior notice, to terminate this Agreement, including the license, effective immediately, for failure to comply with this Agreement. Upon any such termination, You shall immediately cease all use of the Program and shall return to Enterasys the Program and all copies of the Program.
9. OWNERSHIP. This is a license agreement and not an agreement for sale. You acknowledge and agree that the Program constitutes trade secrets and/or copyrighted material of Enterasys and/or its suppliers. You agree to implement reasonable security measures to protect such trade secrets and copyrighted material. All right, title and interest in and to the Program shall remain with Enterasys and/or its Affiliates and suppliers. All rights not specifically granted to You shall be reserved to Enterasys.
10. TRADEMARKS. ENTERASYS, ENTERASYS NETWORKS, ENTERASYS SECURE NETWORKS, NETSIGHT, ENTERASYS NETSIGHT, and any logos associated therewith, are trademarks or registered trademarks of Enterasys Networks, Inc., in the United States and/or other countries. Extreme Networks and the Extreme Networks logo are trademarks or registered trademarks of Extreme Networks, Inc. in the United States and/or other countries. For more information on Enterasys and Extreme trademarks, see: www.extremenetworks.com/about-extreme/trademarks.aspx.

    All other product names mentioned in Program may be trademarks or registered trademarks of their respective companies.
11. ENFORCEMENT. You acknowledge and agree that any breach of this Agreement by You may cause Enterasys irreparable damage for which recovery of money damages would be inadequate, and that Enterasys may be entitled to seek timely injunctive relief to protect Enterasys' rights under this Agreement in addition to any and all remedies available at law.
12. ASSIGNMENT. You may not assign, transfer or sublicense this Agreement or any of Your rights or obligations under this Agreement, except that You may assign this Agreement to any person or entity that acquires substantially all of Your stock assets. Enterasys may assign this Agreement in its sole discretion to anyone or any entity without Your consent and without notice, including assigning this Agreement to its parent company, Extreme Networks, Inc. This Agreement shall be binding upon and inure to the benefit of the parties, their legal representatives, permitted transferees, successors and assigns as permitted by this Agreement. Any attempted assignment, transfer or sublicense in violation of the terms of this Agreement shall be void and a breach of this Agreement.

13. WAIVER. A waiver by Enterasys of a breach of any of the terms and conditions of this Agreement must be in writing and will not be construed as a waiver of any subsequent breach of such term or condition. Enterasys' failure to enforce a term upon Your breach of such term shall not be construed as a waiver of Your breach or prevent enforcement on any other occasion.
14. SEVERABILITY. In the event any provision of this Agreement is found to be invalid, illegal or unenforceable, the validity, legality and enforceability of any of the remaining provisions shall not in any way be affected or impaired thereby, and that provision shall be reformed, construed and enforced to the maximum extent permissible. Any such invalidity, illegality, or unenforceability in any jurisdiction shall not invalidate or render illegal or unenforceable such provision in any other jurisdiction.
15. TERMINATION. Enterasys may terminate this Agreement immediately upon Your breach of any of the terms and conditions of this Agreement. Upon any such termination, You shall immediately cease all use of the Program and shall return to Enterasys the Program and all copies of the Program.

Contents

Chapter 5: Virtual Switch Bonding (VSB) Configuration

Chapter 6: Port Configuration

Chapter 7: Ethernet Operations, Administration, and Maintenance (OAM) Configuration

Chapter 8: Port Mirroring Configuration

Chapter 9: System Configuration

Chapter 13: Tracked Object Manager Configuration

Chapter 14: Bidirectional Forwarding Detection (BFD)Configuration

Chapter 15: Link-State Configuration

Chapter 16: IP SLA Configuration

Chapter 17: Power over Ethernet Configuration

Chapter 18: Discovery Protocol Configuration

Chapter 19: Data Center Bridging Configuration

Chapter 20: Simple Network Management Protocol (SNMP) Configuration

Chapter 21: Spanning Tree Configuration

Chapter 24: VLAN Configuration

Chapter 25: Link Aggregation Control Protocol (LACP) Configuration

Chapter 26: Policy Configuration

Chapter 27: Multicast Configuration

Chapter 28: MSDP Configuration

Chapter 29: Multi-Topology Configuration

Chapter 30: Multicast Listener Discovery (MLD) Configuration

Chapter 31: System Logging Configuration

Chapter 32: Network Monitoring Configuration

Chapter 33: NetFlow Configuration

Chapter 34: Connectivity Fault Management Configuration

Chapter 37: Tunneling Configuration

Chapter 38: Layer 3 Virtual Private Network (VPN) Configuration

Chapter 39: Routing Information Protocol (RIP) Configuration

Chapter 40: Routing Information Protocol Next Generation (RIPng) Configuration

Chapter 41: Open Shortest Path First (OSPFv2) Configuration

Chapter 44: Border Gateway Protocol (BGP) Configuration

Chapter 45: Network Address Translation (NAT) Configuration

Chapter 46: Load Sharing Network Address Translation (LSNAT) Configuration

Chapter 47: Transparent Web Cache Balancing (TWCB) Configuration

Chapter 48: Virtual Router Redundancy Protocol (VRRP) Configuration

Chapter 49: Security Configuration

Chapter 50: Flow Setup Throttling Configuration

Chapter 51: Route-Map Manager Configuration

Chapter 52: Access Control List Configuration

Chapter 53: Quality of Service (QoS) Configuration

Chapter 54: Anti-Spoofing Configuration

Chapter 55: RADIUS-Snooping Configuration

Chapter 56: Authentication Configuration

Procedures

Figures

Tables

About This Guide

This manual explains how to configure Extreme Networks S-Series® switch/router devices.

How to Use This Guide

Read through this guide completely to familiarize yourself with its contents and to gain an understanding of the features and capabilities of the S-Series modules. A general working knowledge of data communications networks is helpful when setting up these modules.

Related Documents

The manuals listed below can be obtained from the World Wide Web in Adobe Acrobat Portable Document Format (PDF) at the following site:

http://support.extremenetworks.com/

- *Extreme Networks S-Series CLI Reference* provides information on how to use the Command Line Interface for the S-Series switch/routers.

Conventions Used in This Guide

The following conventions are used in the text of this document:

Convention	Description
Bold font	Indicates mandatory keywords, parameters or keyboard keys.
italic font	Indicates complete document titles.
Courier font	Used for examples of information displayed on the screen.
Courier font in italics	Indicates a user-supplied value, either required or optional.
[ ]	Square brackets indicate an optional value.
{ }	Braces indicate required values. One or more values may be required.
|	A vertical bar indicates a choice in values.
[x | y | z]	Square brackets with a vertical bar indicates a choice of a value.
{x | y | z}	Braces with a vertical bar indicate a choice of a required value.
[x {y | z}]	A combination of square brackets with braces and vertical bars indicates a required choice of an optional value.

The following icons are used in this guide:

Note: Calls the reader's attention to any item of information that may be of special importance.

Router: Calls the reader's attention to router-specific configuration information.

Caution: Contains information essential to avoid damage to the equipment.

Precaución: Contiene información esencial para prevenir dañar el equipo.

Achtung: Verweißt auf wichtige Informationen zum Schutz gegen Beschädigungen.

Commonly Used Acronyms

The following acronyms are used extensively throughout this guide:

- IOM – Input/Output Module
- FM – Fabric Module
- LED – Light Emitting Diode
- USB – Universal Serial Bus

Getting Help

For additional support related to S-Series switch/router or to this document, contact Enterasys Networks using one of the following methods:

World Wide Web	http://support.extremenetworks.com/
Phone	1-800-872-8440 (toll-free in U.S. and Canada) or 1-603-952-5000 For the Extreme Networks Support toll-free number in your country: www.extremenetworks.com/support/contact/
Internet mail	support@extremenetworks.com To expedite your message, please type **[S-SERIES]** in the subject line.

Before contacting Enterasys Networks **for technical support, have the following data ready:**

- Your Enterasys Networks service contract number
- A description of the failure
- A description of any action(s) already taken to resolve the problem (for example, changing mode switches or rebooting the unit)
- The serial and revision numbers of all involved Enterasys Networks products in the network
- A description of your network environment (such as layout, cable type, other relevant environmental information)
- Network load and frame size at the time of trouble (if known)
- The device history (for example, if you have returned the device before, or if this is a recurring problem)
- Any previous Return Material Authorization (RMA) numbers

1

Getting Started

This chapter provides the procedures to start the S-Series device once the hardware is installed. Initially, the system can only be configured using the Command Line Interface (CLI) from a device connected directly to the console port on the chassis.

This chapter also provides an overview of configuring the S-Series as a switch and router to fit into your network.

Notes: See the default parameters table located in the relevant chapter for factory default values.

Device Management Methods

The S-Series device can be managed using the following methods:

- Locally using a VT type terminal connected to the console port.
- Remotely using a VT type terminal connected through a modem.
- Remotely using an SNMP management station.
- In-band through a Telnet connection.
- In-band using Extreme Networks' NetSight management application.
- Remotely using WebView™, Extreme Networks' embedded web server application.

The *Hardware Installation Guide* for your S-Series device provides setup instructions for connecting a terminal or modem to the device.

Initial Configuration

To initially configure the S-Series device, you must have connected a terminal to the local console port as described in the *Hardware Installation Guide* for your S-Series device. Procedure 1-1 contains the steps to assign an IP address and configure basic system parameters. For information on the command syntax and parameters, refer to the online help or the *Extreme Networks S-Series CLI Reference*.

Note: When configuring any string or name parameter input for any command, do not use any letters with diacritical marks (an ancillary glyph added to a letter). Diacritical marked letters are not supported by SNMP.

Procedure 1-1 Initial Setup

Step	Task	Command
1.	Log in as an administrator.	• At the login prompt, enter **admin**. • Press **Enter** for the password (no password string by default).
2.	For security, change the password.	**set password**
3.	Optionally, check the version of the firmware image then check the Extreme Networks web site to verify that you have the latest version.	**show version**
4.	Optionally, define a name for the system, the location of the system, and contact information for system issues.	**set system name** [*string*] **set system location** [*string*] **set system contact** [*string*]
5.	Optionally, define a message that displays whenever a user logs in.	**set banner** {**motd** | **login**} *message*
6.	Optionally, change the default prompt.	**set prompt** *"prompt_string"*
7.	Display the system's setting for the date and time. If necessary, change the setting. **NOTE**: Instead of manually setting the time, you can configure the system as an SNTP client, as described in "SNTP Overview" on page 9-13.	**show time** **set time** [*mm/dd/yyyy*] [*hh:mm:ss*]
8.	Assign a management IP address.	**set ip interface** **set ip address**
9.	If desired, configure additional user accounts and passwords. Up to 32 user accounts may be registered with the local database.	**set system login** *username*

Advanced Configuration Overview

The S-Series device can be configured to provide various system services, Layer 2 switching, Layer 3 routing, and security. Table 1-1 provides an overview of configuring the S-Series device for each area.

Note: Though it is possible to configure policy by using the CLI, Extreme Networks recommends that you use NetSight Policy Manager instead.

Table 1-1 Advanced Configuration

Task	Refer to page...
System Services	
Configure the Simple Network Time Protocol (SNTP) client.	9-13
Configure the Domain Name Server (DNS) client.	9-24

Table 1-1 Advanced Configuration (continued)

Task	Refer to page...
Configure the Telnet client and server. (Telnet client is enabled by default.) **Note:** For security, you may wish to disable Telnet and only use SSH.	9-19
Configure the Secure Shell V2 (SSHv2) client and server.	9-20
Configure the Dynamic Host Configuration Protocol (DHCP) client and server.	9-27
Configure the port parameters, such as speed and duplex mode.	6-1
Enable SNMP and create a community string. By default, the SNMP master agent is disabled and no defined public community string is configured.	20-18
Configure RMON to provide comprehensive network fault diagnosis, planning, and performance tuning information, and allow for interoperability between SNMP management stations and monitoring agents.	32-4
Change the interactive login authentication method, from local to remote (RADIUS authentication).	56-29
If RADIUS authentication is configured, configure the remote RADIUS servers to be used by the RADIUS client on the S-Series	56-29
Layer 2 Switching	
Enable desired ports for switching.	6-4
Set port configurations and port-based Virtual Local Area Networks (VLANs). VLANs can be created statically or dynamically.	24-3
Configure Spanning Trees using STP, RSTP, or MSTP.	21-21
Configure LLDP or CDP.	18-1
Layer 3 Routing	
Configure the router id. Refer to the **router id** command in the *Extreme Networks S-Series CLI Reference*.	
Configure interfaces for IP routing.	36-3
Configure the ARP table.	36-22
Configure UDP broadcast forwarding, including DHCP/BOOTP relay agent.	36-26
Configure routes.	36-13
Configure interior gateway protocols: RIP and OSPF.	39-1, 41-1
Configure multicast protocols IGMP, DVMRP, and PIM, and general multicast parameters.	27-20
Configure VRRP.	48-1
Configure policy-based routing.	26-1
Security and General Management	
Configure Access Control Lists (ACLs).	44-1
Configure RADIUS servers.	56-29
Manage user accounts and passwords.	1-1
Configure system logging.	31-6
Configure the S-Series using text files.	3-1

Table 1-1 Advanced Configuration (continued)

Task	Refer to page...
Upgrade system firmware.	3-1
Configure QoS features.	53-10
Configure policy.	26-15

2

Using the CLI

This chapter provides information about CLI conventions for S-Series devices and CLI properties that you can configure.

CLI Conventions

Getting Help with CLI Syntax

The S-Series device allows you to display usage and syntax information for individual commands by typing **help** or **?** after the command.

Using Context-Sensitive Help

Entering **help** after a specific command will display usage and syntax information for that command. This example shows how to display context-sensitive help for the **set length** command:

```
S Chassis(rw)->set length help
Command: set length Number of lines
Usage:  set length  <screenlength>
        screenlength     Length of the screen (5..512, 0 to disable 'more')
```

Performing Keyword Lookups

Entering a space and a question mark (**?**) after a keyword will display all commands beginning with the keyword. The following example shows how to perform a keyword lookup for the **show snmp** command. In this case, 13 additional keywords are used by the **show snmp** command. Entering a space and a question mark (**?**) after any of these parameters (such as **show snmp user**) will display additional parameters nested within the syntax.

```
S Chassis(rw)->show snmp ?
  access            SNMP VACM access configuration
  community         SNMP v1/v2c community name configuration
  context           SNMP VACM context list
  counters          SNMP counters
  engineid          SNMP engine properties
  group             SNMP VACM security to group configuration
  notify            SNMP notify configuration
  notifyfilter      SNMP notify filter configuration
  notifyprofile     SNMP notify profile configuration
  targetaddr        SNMP target address configuration
  targetparams      SNMP target parameters configuration
  user              SNMP USM user configuration
  view              SNMP VACM view tree configuration
S Chassis(rw)->show snmp
S Chassis(rw)->show snmp user ?
  list              List usernames
  <user>            User name
  remote            Show users with remote SNMP engine ID
  volatile          Show temporary entries
  nonvolatile       Show permanent entries
  read-only         Show r/o entries
  <cr>
S Chassis(rw)->show snmp user
```

Entering a question mark (**?**) without a space after a partial keyword will display a list of commands that begin with the partial keyword. The following example shows how to use this function for all commands beginning with **co**:

```
S Chassis(rw)->co?
  configure         Execute a configuration file
  copy              Upload or download an image or configuration file
S Chassis(rw)->co
```

Note: At the end of the lookup display, the system will repeat the command you entered without the ?.

Displaying Scrolling Screens

If the CLI screen length has been set using the **set length** command as described in Table 2-1 on page 2-4, CLI output requiring more than one screen will display --More-- to indicate continuing screens. To display additional screen output:

- Press any key other than ENTER to advance the output one screen at a time.
- Press ENTER to advance the output one line at a time.

The following example shows how the **show mac** command indicates that output continues on more than one screen.

```
S Chassis(rw)->show mac


MAC Address           FID       Port      Type
 ------------------------------------------------------------
 00-00-1d-67-68-69      1       host.0.1  learned
 00-00-02-00-00-00      1       ge.1.2    learned
 00-00-02-00-00-01      1       ge.1.3    learned
 00-00-02-00-00-02      1       ge.1.4    learned
 00-00-02-00-00-03      1       ge.1.5    learned
 00-00-02-00-00-04      1       ge.1.6    learned
 00-00-02-00-00-05      1       ge.1.7    learned
 00-00-02-00-00-06      1       ge.1.8    learned
 00-00-02-00-00-07      1       ge.1.9    learned
 00-00-02-00-00-08      1       ge.1.10   learned
--More--
```

Abbreviating and Completing Commands

The S-Series device allows you to abbreviate CLI commands and keywords down to the number of characters that will allow for a unique abbreviation. The following example shows how to abbreviate the **show netstat** command to **show net**.

```
S Chassis(rw)->show net
Active Internet connections (including servers)
Proto Recv-Q Send-Q  Local Address          Foreign Address        State
----- ------ ------  --------------------- ---------------------- -------
TCP        0      0  10.21.73.13.23        134.141.190.94.51246   ESTABLISHED
TCP        0    275  10.21.73.13.23        134.141.192.119.4724   ESTABLISHED
TCP        0      0  *.80                  *.*                    LISTEN
TCP        0      0  *.23                  *.*                    LISTEN
UDP        0      0  10.21.73.13.1030      134.141.89.113.514
UDP        0      0  *.161                 *.*
UDP        0      0  *.1025                *.*
UDP        0      0  *.123                 *.*
```

Using the Spacebar Auto Complete Function

When the spacebar auto complete function is enabled, pressing the spacebar after a CLI command fragment will allow you to determine if the fragment is unique. If it is, the CLI will complete the fragment on the current display line.

By default, this function is disabled. For more information on enabling it using the **set cli completion** command, refer to Table 2-1 on page 2-4. The following example shows how, when the function is enabled, entering **conf** and pressing the spacebar would be completed as configure:

```
S Chassis(rw)->conf<SPACEBAR>
S Chassis(rw)->configure
```

Configuring CLI Properties

CLI properties are options that you can configure and customize in the CLI, such as the command prompt, command completion, banner messages, and session idle timeout.

Table 2-1 lists CLI properties configuration commands.

Table 2-1 CLI Properties Configuration Commands

Task	Command
Modify the command prompt	**set prompt** *prompt-string*
Enable or disable the CLI command completion function. When enabled, this allows you to complete a unique CLI command fragment using the keyboard spacebar.	**set cli completion** {**enable** | **disable**} [**default**]
Set the banner message for pre and post session login.	**set banner** {**login** *message* **| motd** *message*}
Clear the banner message displayed at pre and post session login to a blank string.	**clear banner** {**login | motd**}
Set the number of columns for the terminal connected to the device's console port.	**set width** *screenwidth* [*default*]
Set the number of lines the CLI will display.	**set length** *screenlength* [*default*]
Set the time (in minutes) an idle console or Telnet CLI session will remain connected before timing out.	**set logout** *timeout* [*default*]
Set the current and default line editing mode or the way the Delete character is treated by the line editor. You can also set the persistence of your line editing selections.	**set line-editor** {**emacs** | **vi** | **default** | **delete** {**backspace | delete**}} [**default**]

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

Example CLI Properties Configuration

In this example, the prompt is changed and a login banner is added.

```
S Chassis(rw)->set prompt "Switch 1"
Switch 1(rw)->
Switch 1(rw)->set banner login There is nothing more important than our customers
```

CLI Properties Display Commands

Table 2-2 lists CLI properties show commands.

Table 2-2 CLI Properties Show Commands

Task	Command
Display the current and default line-editor mode and Delete character mode.	**show line-editor**
Display the banner message that will display at pre and post session login.	**show banner**
Display the number of columns for the terminal connected to the device's console port.	**show width**
Display the current screen length.	**show length**
Display the time (in seconds) an idle console or Telnet CLI session will remain connected before timing out.	**show logout**

Refer to the *Extreme Networks S-Series CLI Reference* for a description of the output of each command.

3

Image Configuration and File Management

This chapter provides information about configuration and image file management on the S-Series devices.

Configuration and Image File Management on Your System

On S-Series devices, configuration and image file management includes the following:

- Saving a configuration
- Executing a configuration
- Deleting an image file, configuration file, or script file
- Downloading an image file, configuration file, or a script file
- Uploading a configuration file
- Setting the boot firmware image
- Running a configuration script created on a PC

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

Automated Deployment

The automated deployment feature allows a newly installed device, with no administrative configuration (default configuration), to automatically obtain the latest firmware revision and configuration from the network. The DHCP client, using a default VLAN and base MAC, obtains a temporary IP address and mask and the IP address of the NetSight server from the DHCP server. Use DHCP Vendor Class Identifier (VCI) code option 60 to identify "Extreme Networks, Inc." as the vendor. Use DHCP vendor specific code option 125 to specify the information required for the switch to send the trap to start the Netsight OneView application.

Upon obtaining the NetSight IP address, automated deployment sends a notification to NetSight, informing NetSight that its status is READY. NetSight, using its inventory manager, drives the needed changes to the device by determining the configuration to apply and querying the device as to any required upgrades. Refer to the NetSight help page "New Device Configuration in OneView" for NetSight device configuration information.

The automated deployment feature has no CLI input associated with it. Its only dependency is whether the device is currently at the default configuration.

To determine if a device is currently at default configuration, login to the device and a banner will display stating that the system is running with default configuration with a temporary IP address assigned by the DHCP server.

If any CLI configuration is entered for a device configured at default configuration, the device releases the IP address, any SNMP and DHCP settings are backed out, and the device exits automated deployment mode.

To administratively set the device to its default configuration enter the **clear config all** command. If any licenses are configured, also enter the **clear license all** command.

Saving a Configuration

You can save the S-Series device configuration by doing one of the following:

- Creating a configuration restore-point. The configuration restore-point resides on your system. You cannot save a configuration restore-point to a file. Any additional configuration settings that you change after creating this restore-point will not be included when the restore-point configuration is applied, such as when the system reboots. You can configure only one restore-point.

  To create a configuration restore-point of the current configuration, use the **set config restore-point** command.

  set config restore-point *<description>*

- Write the configuration to a file.

  To write the configuration to a file, use the **show config** command.

  show config outfile *outfile*

  outfile must include the slotN/ local file path directory.

Example: Creating a Configuration Restore-Point

```
S Chassis(rw)-> set config restore-point 25June2009_0800
```

Example: Creating a Configuration File

```
S Chassis(rw)-> show config outfile slot1/newconfig
```

Executing a Configuration

You can execute the S-Series device configuration by doing one of the following:

- Execute the configuration restore-point. Any changes that you made to the configuration after you created the configuration restore-point will be overwritten. See Procedure 3-1.
- Execute a configuration file that was created on, or downloaded to, the S-Series device.

Procedure 3-1 Executing the Configuration Restore-Point

Step	Task	Command(s)
1.	View the index of the configuration restore-point.	**show config restore-point**
2.	Indicate that the restore-point will be applied when the S-Series device reboots. When the S-Series device reboots, any configuration changes made after the restore-point was set will be lost.	**configure restore-point** *index*
3.	Reboot the S-Series device.	**reset**
4.	(Optional) Append the current configuration with the configuration in a previously downloaded or created configuration file. **Note:** If you do not specify **append**, the current running configuration will be replaced with the contents of the configuration file, which will require an automated reset of the chassis.	**configure** *filename* [**append**] [**chassis-id** *chassis-id*]

To execute the configuration in a configuration file stored on the S-Series device, use the **configure** command.

configure *filename* [**append**] [**chassis-id** *chassis-id*]

filename must include the slotN/ file path.

The **append** option executes the configuration as an appendage to the current configuration. This is equivalent to typing the contents of the config file directly into the CLI and can be used, for example, to make incremental adjustments to the current configuration.

The **chassis-id** option is used when replacing a chassis in a VSB system. A replacement chassis will have a serial-number that no longer agrees with the serial-number associated with the configuration file chassis ID entry. Specifying the **chassis-id** option will tell the system to replace chassis specific information in the configuration file with settings of the chassis being configured.

Example: Executing a Configuration Restore-Point

```
S Chassis(rw)->show config restore-point

 Index:          1245935343
 Creation Date:  THU JUN 25 13:09:03 2009
 Description:    test
S Chassis(rw)-> configure restore-point 1245935343
S Chassis(rw)->reset
```

Example: Executing a Configuration File

```
S Chassis(rw)->configure slot1/myconfig
```

Deleting a Configuration Restore-Point or File

You can delete the S-Series device configuration by doing one of the following:

- Delete the configuration restore-point. See Procedure 3-2.
- Delete a configuration file.

Procedure 3-2 Deleting the Configuration Restore-Point

Step	Task	Command(s)
1.	View the index of the configuration restore-point.	**show config restore-point**
2.	Delete the current restore-point. Because the system currently supports only one restore-point, you must delete the current restore-point before creating a new one.	**clear config restore-point** *index*
3.	(Optional) Create a new restore-point.	**set config restore-point** <*description*>

To delete a configuration file, image file, or script file, use the **delete** command.

delete *filename*

filename must include the slotN/ or images/ file path directory.

Example: Deleting a Configuration Restore-Point

```
S Chassis(rw)-> clear config restore-point 1245935343
```

Example: Deleting a Configuration File

```
S Chassis(rw)->delete slot3/myconfig
```

Example: Deleting an Image File

```
S Chassis(rw)->delete images/010300
```

Downloading a File from an FTP, TFTP, or SCP Server

You can download an image file, a configuration file, or a script file from an FTP, TFTP, or SCP server to the S-Series device.

To download an image file, configuration file, or script file from an FTP, TFTP, or SCP server, use the **copy** command.

copy *source destination*

- *source* is the URL of an FTP, TFTP, or SCP server.
- *destination* is the local file path. For a configuration or script file, *destination* must include slotN/.

The S-Series module to which a configuration file is downloaded must have the same hardware configuration as the S-Series module from which it was uploaded.

For reasons of security, passwords are not allowed in **copy** command URLs. A password prompt displays upon entering a **copy** command. For example:

```
S Chassis(rw)->copy scp://doc@banshee.extremenetworks.com:22/myconfig
slot3/myconfig
Password:
###########
S Chassis(rw)->
```

Once you have downloaded an image file, set the device to load the new image file at startup using the **set boot system** command. See "Setting the Boot Firmware Image" on page 3-8.

For information on downloading

Example: Downloading an Image File

```
S Chassis(rw)->copy tftp://134.141.89.34/ets-mtxe7-msi newimage
```

Example: Downloading a Configuration File

```
S Chassis(rw)->copy tftp://134.141.89.34/myconfig slot3/myconfig
```

Downloading a Firmware Image via the Serial Port

Besides using FTP, TFTP, or SCP for downloading firmware images, you can also download firmware images via the serial (console) port. This procedure is an out-of-band operation that copies the firmware through the serial port to the device. It should be used in cases when you cannot connect to the device to perform the in-band **copy** download procedure via FTP, TFTP or SCP. Serial console download has been successfully tested with the following applications:

- HyperTerminal
- TeraTerm

Any other terminal applications may work but are not explicitly supported.

Important Notice

The S-Series device allows you to download and store multiple image files. This feature is useful for reverting back to a previous version in the event that a firmware upgrade fails to boot successfully. After downloading firmware as described above, you can select which image file you want the device to load at startup using the **setboot** command in the System Image Loader menu or the **set boot system** command.

To download device firmware via the serial (console) port, proceed as follows:

1. With the console port connected, power up the device. The following message displays:

```
Boot ROM Initialization, Version 01.00.02


Copyright (c) 2003  Enterasys Networks, Inc.
SDRAM size: 1024 MB
Testing SDRAM....                       PASSED.
Loading Boot Image: 01.00.19...         DONE.
Uncompressing Boot Image...             DONE.
```

2. Once the boot image is finished uncompressing, you receive a message indicating you have 3 seconds to access the bootloader menu by pressing any key. Press a key and the system image loader prompt displays:

```
###You have 3 seconds to access the bootloader menu###
Press any key to enter System Image Loader menu
PressAnyKey
[System Image Loader]:
```

3. To display help for all the system image loader mode commands, enter a question mark (**?**):

```
[System Image Loader]:?
 ?, help                        - print this list
```

```
 boot                          - boot (load and go)
 delete                        - delete an image file
 download                      - start ZMODEM download
 list                          - display available images
 log                           - message log
 setbaud <rate>                - set baud rate, (9600,38400,57600,115200)
 setboot <filename>            - change boot image file
 showboot                      - display boot image file
 clearnvram                    - clear persistent storage
[System Image Loader]:
```

4. Use the **list** command to display the images currently on this device.
5. The baud rate can be set to 9600, 38400, 57600, or 115200. Using the **setbaud** command, set the baud rate to **115200**:

```
[System Image Loader]: setbaud 1152000
###Change the baud of the terminal program to 1152000###
[System Image Loader]:
```

6. Use the **download** command to start the ZMODEM receive process. Send the image file using the ZModem protocol from your terminal application. (This procedure will vary depending on your application.) When the ZModem download is finished, the following message displays:

```
[System Image Loader]: download
Preparing to receive file...
**xxxxxxxxxxxxxxx
###Start the ZMODEM transfer from the terminal software###
Writing file...
Download successful.
[System Image Loader]:
```

7. Use the **list** command to confirm the images that are currently on the device, and confirm the image currently listed as the boot image. If the current boot image is not the image you want to boot with, use the **setboot** *filename* command to set the correct boot image:

```
[System Image Loader]: list
Filename:      720010001 (Boot)
Version:       07.20.01.0001
Size:          4527490 (bytes)
Date:          FRI DEC 10 15:32:24 2010
CheckSum:      d89ace409317bc765789fce1c73b8745
Compatibility: listOfCompatibleDevices

Filename:      720010025
Version:       07.20.01.0025
Size:          4529790 (bytes)
Date:          THU DEC 09 22:38:54 2010
CheckSum:      6ccaaf8a5b77d7d34c6c3d972b381024
Compatibility: listOfCompatibleDevices
[System Image Loader]:setboot 720010025
```

```
[System Image Loader]:list
Filename:      720010001
Version:       07.20.01.0001
Size:          4527490 (bytes)
Date:          FRI DEC 10 15:32:24 2010
CheckSum:      d89ace409317bc765789fce1c73b8745
Compatibility: listOfCompatibleDevices

Filename:      720010025 (Boot)
Version:       07.20.01.0025
Size:          4529790 (bytes)
Date:          THU DEC 09 22:38:54 2010
CheckSum:      6ccaaf8a5b77d7d34c6c3d972b381024
Compatibility: listOfCompatibleDevices
[System Image Loader]:
```

8. When a device is booted, the device baud rate is reset to 9600. Reset the terminal application baud rate to 9600 so that it will continue to display output from the device:

```
[System Image Loader]: setbaud 9600
[System Image Loader]:
```

9. Use the **boot** command to boot the image:

```
[System Image Loader]: boot
###The unit will boot normally###
                                                        /flash0/  - Volume is OK
Loading 61205...                        DONE.
Uncompressing System Image...           DONE.
Loading System Image...                 DONE.
Initializing Platform Hardware


.
.
.
Enterasys Networks, Inc.
50 Minuteman Rd.
Andover, MA 01810-1008 USA

Phone:  +1 978 684 1000
E-mail: support@enterasys.com
WWW:    http://www.enterasys.com
(c) Copyright Enterasys Networks, Inc. 2014
Chassis Serial Number:      00e063937c7d
Chassis Firmware Revision:  07.20.01.0025
Username:
```

Note: If you reboot without specifying the image to boot with **setboot** as described above, the device will attempt to load whatever image is currently stored in the bootstring via the **setboot system** command. If the device cannot find the image, or it is not set, it will search through available images and attempt to boot the newest one. If the device finds and successfully boots an image file, it will set the bootstring to the name of that image file.

Uploading a Configuration File

You can upload a configuration file from the S-Series device.

To upload a configuration file, use the **copy** command.

copy *source destination*

- *source* is the local file path and must include slotN/.
- *destination* is the URL of an FTP, TFTP, or SCP server.

Example

```
S Chassis(rw)->copy slot3/myconfig ftp://134.141.89.34/myconfig
```

Setting the Boot Firmware Image

You can set the boot firmware image, which is the image that will be loaded automatically after the system has been reset.

To set the boot firmware image, use the **set boot system** command.

set boot system *filename*

The system must be reset by software for the new boot image to take effect at startup. If the chassis is powered OFF and then back ON, the current active image will just reload at startup

Although it is not necessary to choose to reset the system and activate the new boot image immediately, the CLI will prompt you whether or not you want to do so. You can choose "Yes" at the question prompt to have the system reset and load the new boot image immediately, or choose "No" to load the new boot image at a later scheduled time by issuing one of the following commands: **clear config**, **reset**, or **configure**. The new boot setting will be remembered through resets and power downs, and will not take effect until the **clear config**, **reset**, or **configure** command is given.

Example

```
S Chassis(rw)->set boot system newimage
This command can optionally reset the system to boot the new image.
Do you want to reset now (y/n) [n]?y
Resetting system ...
```

Running a Configuration Script

You can run a configuration script that you have downloaded to the S-Series device. See Procedure 3-3.

Procedure 3-3 Running a Configuration Script

Step	Task	Command(s)
1.	Download the configuration script. *source* is the URL of an FTP, TFTP, or SCP server. *destination* is the local file path and must include slotN/.	**copy** *source destination*
2.	Run the configuration script.	**script** *filename* [*arg1*] [*arg2*] [*arg3*] [*arg4*] [*arg5*] [*arg6*] [*arg7*]

Example

This example uses the **copy** command to copy the script file named "setport.scr" from IP address 10.1.221.3 to slot 4. Next, the contents of the file is displayed with the **show file** command. The script file requires two arguments, a port string (%1) and a VLAN id (%2). Finally, the script is executed, by specifying ge.1.1 as the first argument and 100 as the second argument.

```
S Chassis(rw)->copy tftp://10.1.221.3/setport.scr slot4/setport.scr


S Chassis(rw)->show file slot4/setport.scr
set port alias %1 script_set_port
set port vlan %1 %2 modify-egress
set port jumbo enable %1
set port disable %1
set port lacp port %1 disable


S Chassis(rw)->script slot4/setport.scr ge.1.1 100
```

Configuration and Image File Display Commands

Table 3-1 lists configuration and image file display commands for S-Series devices.

Table 3-1 Configuration and Image File Management and Display Commands

Task	Command
Display the index, creation date, and description of the currently configured restore-point. If "(Boot)" is listed after the index entry, this restore-point will be used when the system reboots next.	**show config restore-point**
Display the firmware image the system will load at the next system reset.	**show boot system**
List files stored in the file system.	**dir** [*filename*]
Display the contents of an image or configuration file.	**show file** *filename*
Display the system configuration.	**show config** [**all**] [*facility*]

Refer to the device's *CLI Reference Guide* for a description of the output of each command.

4

High Availability Firmware Upgrade (HAU) Configuration

This chapter provides information about configuring and monitoring a High Availability Firmware Upgrade (HAU) on S-Series devices.

Using High Availability Firmware Upgrade in Your Network

High Availability Firmware Upgrade (HAU) is an S-Series feature that provides for a rolling firmware upgrade for maintenance releases that are HAU compatible with the current system firmware.

There are two methods for loading a system firmware image:

- Standard – The specified image is loaded after a system reset
- High Availability – Provides a rolling firmware upgrade

Using the standard upgrade method, the image is loaded automatically after the system has been reset. The standard method takes the system out of service for the duration of the firmware upgrade. Using the HAU method, all populated system slots are assigned to HAU groups. The firmware upgrade takes place one HAU group at a time with all modules belonging to HAU groups not currently being upgraded remaining operational. As each HAU group completes its upgrade, a mix of slots running the original firmware and slots running the upgraded firmware are simultaneously operating on the device. To avoid potential feature conflicts between multiple firmware versions, the HAU firmware upgrade feature is limited to maintenance firmware upgrades and will not be available when upgrading to major feature releases.

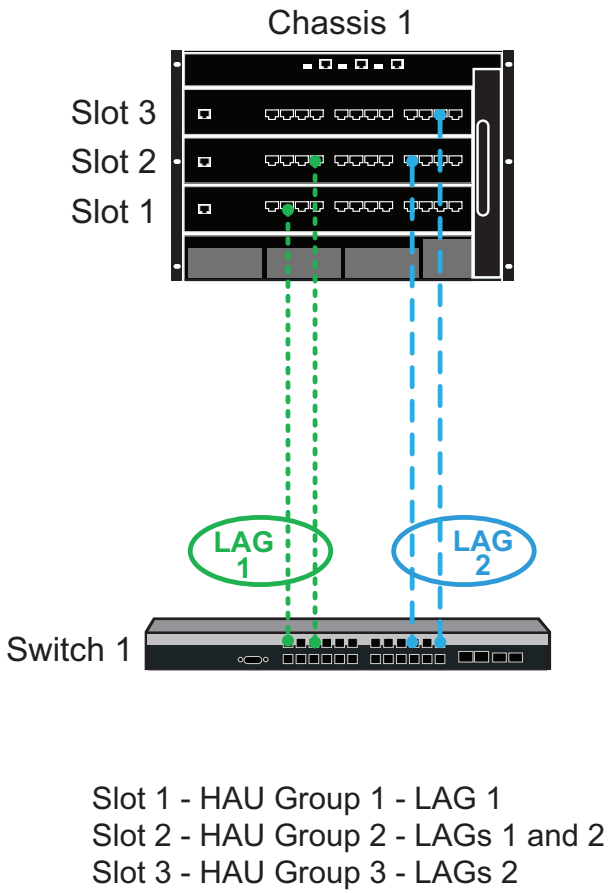
Figure 4-1 displays an example of a default HAU configuration. Chassis 1 is being firmware upgraded. In a default HAU configuration, each slot belongs to a separate HAU group:

- Slot 1 – HAU group1
- Slot 2 – HAU group 2

- Slot 3 – HAU group 3

There are two LAGs configured between Switch 1 and Chassis 1. both LAGs are distributed between two Chassis 1 HAU groups. LAG 1 is configured on Slots 1 and 2. LAG 2 is configured on Slots 2 and 3. As each HAU group upgrades, packets for both LAGs continue to forward over connections to non-upgrading HAU groups.

Figure 4-1 System High Availability Firmware Upgrade Overview



HAU groups can be administratively configured for multiple slots. All slots belonging to the updating HAU group are upgraded simultaneously. Configuring multiple slots to an HAU group shortens the total amount of time required for the system upgrade.

As presented in Figure 4-1, all LAGs on the device must be associated with multiple HAU groups to assure that packets will continue to forward on an HAU group that is not updating.

A number of switch and routing applications require a period of time after the completion of a firmware upgrade to become fully operational. During this period, some resources associated with the just completed HAU group will not be available. By default a 5 second delay takes place between the completion of an HAU group upgrade and the reset of the next HAU group.

When specifying the image to use when booting the system, you can optionally specify whether a standard or high availability system upgrade should be performed.

If no system boot method is specified when setting the system boot image, there are three configurable HAU default modes that determine upgrade behavior:

- A high availability upgrade will never be performed. In this case the standard upgrade method is used in all cases. This is the default system boot behavior.

- A high availability upgrade will be performed if all HAU preconditions are met. If any HAU preconditions are not met, a standard upgrade is performed. See "High Availability Upgrade Preconditions" on page 4-3 for HAU precondition details.
- A high availability upgrade will be performed if all HAU preconditions are met. If any HAU preconditions are not met, no upgrade is performed. See "High Availability Upgrade Preconditions" on page 4-3 for HAU precondition details.

You can not halt a high availability upgrade once it has begun, but you can force all remaining HAU groups that have not yet been upgraded to immediately upgrade. See "Forcing Early Completion of a Running HAU" on page 4-6 for details.

Implementing HAU

To implement HAU:

1. Optionally, modify the default HAU mode. By default a standard (non-high availability) upgrade is performed, unless high availability is set when specifying the boot image.
2. Optionally, modify the default HAU group configuration by placing slots which can be simultaneously upgraded into the same HAU group.
3. Optionally, modify the delay between the completion of one HAU group upgrade and the start of the next HAU group upgrade.
4. Specify the system image that will boot the next time the system resets, and optionally specify whether the image will load using the standard or high availability method. If you do not specify the boot mode, the image will load based upon the default HAU mode referred to in Step 1.

High Availability Upgrade Preconditions

The following preconditions must be met for an high availability upgrade to occur:

- **HAU Compatibility Key** - The target image must have the same HAU Compatibility Key as the active image. To display the HAU key, use the **dir** command, specifying the image to display, or use the **dir** command **image** option to display all images. The HAU key field in the display specifies whether the image displayed is compatible with the current image. If "HAU compatible" is appended to the key field, a high availability upgrade can be performed between the displayed image and the current image.
- **Configuration restore-points** - Configuration restore-points may be set, but must not be configured. A configured restore-point would cause upgraded slots to boot with different configuration data, and all slots must be running the same configuration data.
- **Upgrade Groups** - At least two upgrade groups are required, and each group must contain at least one operational module at the start of a high availability upgrade.
- **Platform** – S-Series S4, S6, and S8 platforms require the presence of at least 2 fabric modules in the system. See the following bullet for an exception to this rule.
- **Virtual Switch Bonding (VSB)** – High availability upgrade is not allowed if the reset of any single upgrade group would break all VSB interconnect bond links. An exception to this rule:
  - High availability upgrade is allowed in a bonded system that would break either the two fabric module restriction or the all VSB interconnect links restrictions, if:
    - A single HAU group is configured per chassis
    - All chassis slots are members of that upgrade group

    In this case, the upgrade is performed per physical chassis.

System Limitations During a High Availability Upgrade

Changes to system configuration cannot be performed while a high availability upgrade is in progress. While a high availability upgrade is running:

- All SNMP set operations are rejected. A “noAccess” reason will be given for the rejection.
- All CLI commands are unavailable with the exception of:
  - reset
  - loop
  - show
  - exit
  - dir
  - history
  - ping
  - traceroute
  - telnet
  - ssh
  - set boot high-availability force-complete

HAU Configuration Overview

Configuring System Boot Image and Mode

When a system is powered on or reset, the current system boot image is loaded on to all system modules. To perform a system upgrade, change the current system boot image to the upgrade image, also referred to as the target image. Image upgrade can occur immediately, the next time the system boots, or by issuing a **reset** command. When specifying the new target image, you can optionally, specify the system boot mode parameter:

- **Standard** – All system slots are simultaneously upgraded taking the system out of operation for the duration of the upgrade. This is a non-high availability upgrade.
- **High-availability** – Providing all HAU preconditions are met, HAU groups are upgraded sequentially (one after another, but not in any specific order). If any HAU precondition is not met, an upgrade does not occur. See “High Availability Upgrade Preconditions” on page 4-3 for HAU precondition details.

If the system boot mode is not specified, the boot mode is determined by the HAU default mode configuration. By default, the HAU default mode executes a **standard** system upgrade. See "Configuring HAU Default Mode" on page 4-5 for HAU default mode details.

Use the **set boot system** *image-file-name* command in any command mode to set the target image for the firmware upgrade, optionally specifying the system boot mode.

Use the **set boot system** *image-file-name* **high-availability** command in any command mode to set the target image for the firmware upgrade and enable HAU, assuring that if a high availability upgrade is possible it will be performed, otherwise no upgrade will occur.

When entering the **set boot system** command, you are asked if you want to reset the system or start a high availability upgrade depending upon the command variation you entered. Answering **y** will immediately reset the system and begin the upgrade. Answering **n** will perform the upgrade the next time the system is reset.

Use the **show boot system** command to determine the current boot image.

Use the **dir** *image-file-name* command to display image details for the specified image.

Use the **dir images** command to display details for all available images on the system.

Configuring HAU Default Mode

HAU default mode determines HAU behavior if a system boot mode is not set when configuring the system boot image. See "Configuring System Boot Image and Mode" on page 4-4 for system boot mode details. There are three HAU default modes:

- **never** – A standard (non-high availability) upgrade is always performed unless over-ridden by the system boot mode **high-availability** setting
- **if-possible** – A high availability upgrade is always performed unless:
  - All HAU preconditions are not met, in which case a standard upgrade is performed
  - Over-ridden by the system boot mode **standard** or **high-availability** settings
- **always** – A high availability upgrade is always performed unless:
  - All HAU preconditions are not met, in which case no upgrade occurs
  - Over-ridden by the system boot mode **standard** setting

See "High Availability Upgrade Preconditions" on page 4-3 for HAU precondition details.

Note: HAU default mode should always be set to **never** unless you intend to perform a high availability upgrade. An **if-possible** or **always** HAU default mode setting in conjunction with no system boot mode specified results in a high availability firmware upgrade each time you reboot your system, if all HAU preconditions are met.

If you want an HAU default mode change to affect a firmware upgrade, the change must take place before configuring a pending upgrade. Changing the HAU default mode after setting the system boot configuration (using the **set boot system** command) has no affect on a pending firmware upgrade.

Use the **set boot high-availability default-mode** command in any command mode to set the HAU default mode.

Configuring HAU Groups

The HAU group feature determines which slot or slots will be simultaneously upgraded. All system slots within the same HAU group are simultaneously upgraded. Each system slot belongs

to an HAU group. HAU occurs one HAU group at a time. By default, there is one slot per group. Therefore, the default HAU behavior is to upgrade each system slot one at a time.

Because HAU groups are upgraded one at a time, the total upgrade time increases with the number of HAU groups configured. In a large chassis it could take a significant amount of time to complete the upgrade and have all physical links back in operation. Upgrade time can be reduced by assigning multiple slots to the same HAU group. When planning system connections, the overall upgrade time will be reduced to the degree that multiple slots can be configured into a single group and still retain sufficient resources in non-upgrading HAU groups to assure system operation. All essential system capabilities on the device should be configured across multiple groups. For example, all LAGs configured on the device should provide sufficient redundancy between HAU groups for packets to continue forwarding on the LAG using slots belonging to HAU groups that are not upgrading.

Use the **set boot high-availability group** command in any command mode to configure an HAU group, specifying the group ID and the system slots that will be members of the HAU group.

Configuring a Delay Between HAU Group Upgrades

When the firmware upgrade of an HAU group completes, depending upon the applications that are configured on the module, it is possible for the next HAU group to begin a firmware upgrade prior to protocols or applications on the just completed HAU module becoming fully operational. Under normal operation there is an approximately 5 second delay between the completion of one HAU group upgrade and the start of the next group upgrade. You can configure an additional delay of up to 600 seconds between the upgrade completion of one HAU group and the beginning of a high availability upgrade for the next HAU group.

Use the **set boot high-availability delay** command in any command mode to set a delay in seconds between the upgrade completion of any HAU group and the beginning of the next HAU group upgrade.

Disabling a Configured HAU

You can disable a pending high availability upgrade by:

- Setting the boot image back to the active image using the **set boot system** *active-image* command
- Deleting the boot image using the **delete** *target-image* command
- Converting the pending high availability upgrade to a standard upgrade by re-issuing the boot command, specifying the target image and the standard system boot mode

After performing one of the methods for disabling an HAU configuration, verify that the HAU status is disabled by using the **show boot high-availability** command.

You cannot disable a high availability upgrade that is running. You can however force the simultaneous upgrade of all remaining non-upgraded HAU groups. See “Forcing Early Completion of a Running HAU” on page 4-6 for details.

Forcing Early Completion of a Running HAU

You cannot disable a high availability upgrade or revert an image back to the original system image on a high availability upgrade that is running. You can however accelerate the upgrade process, by forcing the simultaneous upgrade of all remaining non-upgraded HAU groups.

This should not be considered a normal HAU procedure. It should be assumed that forcing the early completion of HAU will degrade the operational capabilities of the system depending upon the system resources taken out of service.

Use the **set boot high-availability force-complete** command in any command mode to force the simultaneous upgrade of all non-upgraded HAU groups in the system.

High Availability Firmware Upgrade in a Virtual Switch Bonded System

HAU in a Virtual Switch Bonding context assumes that the VSB system is properly configured and fully operational. See "Virtual Switch Bonding (VSB) Configuration" on page 5-1 for VSB configuration details.

There are two restrictions when performing a high availability upgrade to a VSB system:

- There must always be an active interconnect link between VSB chassis during the upgrade
- In the case of an S4, S6, or S8 chassis, there must always be a fabric card that is a member of an HAU group that is not being upgraded

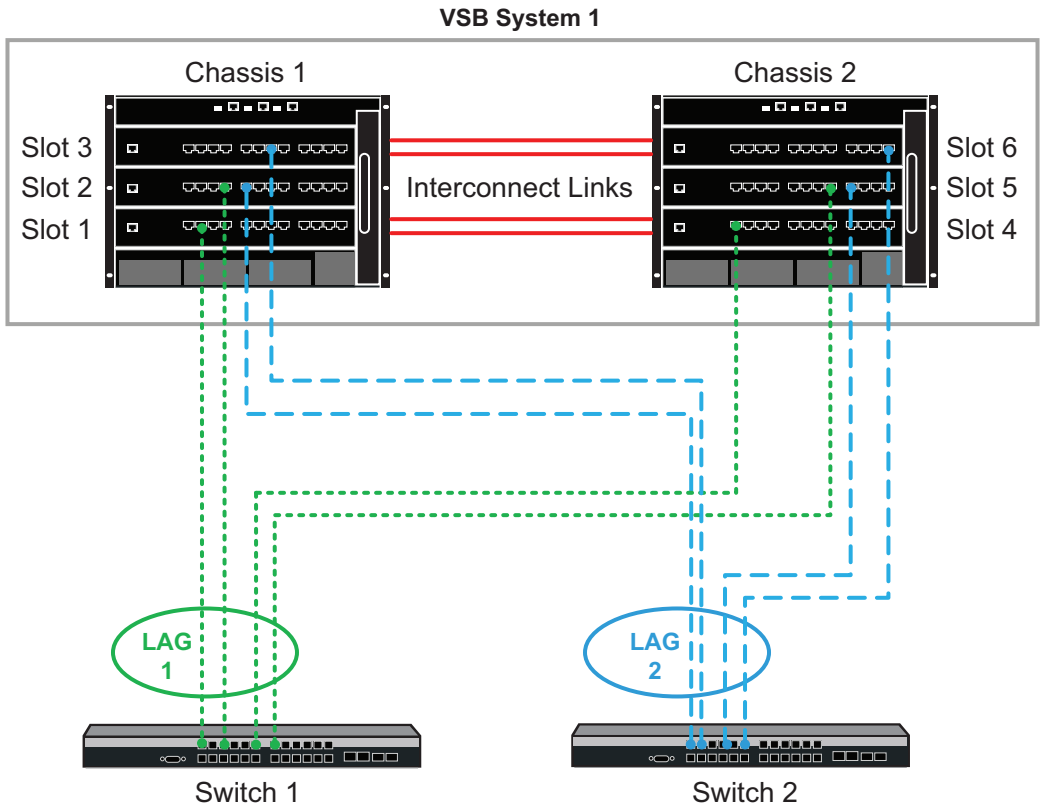
There is an exception to these two restrictions. If a single HAU group is configured per chassis and all chassis slots are members of that HAU group, these restrictions do not apply. All other HAU preconditions apply, the same as they would in an non-VSB system. See "High Availability Upgrade Preconditions" on page 4-3 for details.

Just as in a non-VSB chassis, high availability upgrade takes place one HAU group at a time. By default, each module in a system is in its own group. For example, a VSB system of two S3 chassis (see Figure 4-2) would have six HA groups (groups 1 — 6) by default. As a general rule, in a non-VSB system, the default HAU group configuration should not pose any problems. In a VSB system, updating interconnected modules one at a time, though supported, is not efficient because the interconnect link will be down for two upgrade sessions instead of one. This issue can be avoided by assigning both VSB interconnect link slots to the same HAU group.

Note: It is recommended that you configure a VSB system for high availability upgrade after the VSB system has been globally enabled. You are not prevented from configuring HAU groups prior to globally enabling VSB, but you can not configure slots from both chassis in the same HAU group prior to globally enabling VSB. When VSB interconnected slots are not in the same group, the down time for interconnect links is doubled.

Figure 4-2 displays an example S3 based VSB HA firmware upgrade configuration.

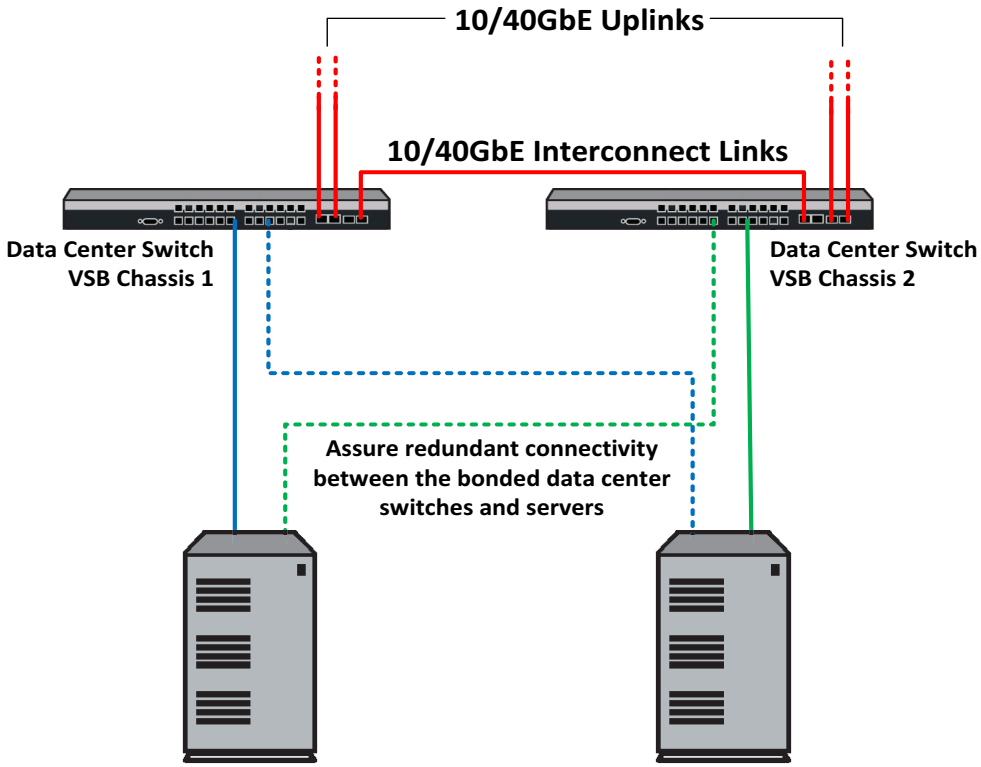
Figure 4-2 VSB System High Availability Firmware Upgrade



We address the restriction that an interconnect link must be up between two non-upgrading modules by assuring that interconnect links are assigned to multiple HAU groups. As each group is upgraded, an active interconnect link remains available. By assuring that the two configured LAGs are distributed between multiple groups in both chassis, traffic continues to be forwarded during the upgrade process.

Figure 4-3 on page 4-9 presents an SSA S-Series HAU VSB configuration in a data center setting. When connecting the standalone bonded data center switches to data center servers, assure there is connectivity from each S-Series switch to each connected server for all configured LAGs. When the S-Series device is taken down for the upgrade, all connectivity from that device is taken out of service for the duration of the upgrade. If redundant connectivity is not present between the servers and the operational bonded switch, any service associated with the missing connectivity will not be available to one or more servers.

Figure 4-3 VSB System High Availability Firmware Upgrade



Configuring HAU

This section provides a table of HAU default values and a procedure for configuring an HAU system.

Table 4-1 lists HAU default values.

Table 4-1 Default HAU Parameters

Parameter	Description	Default Value
system boot mode	Determines firmware load or upgrade behavior when the system is powered on or reset.	standard
HAU delay	A configurable delay in seconds between the upgrade completion of one HAU group and the beginning of the next HAU group upgrade.	0 seconds
HAU default mode	The boot mode that is used when no boot mode is specified when setting the system boot image.	never

Procedure 4-1 describes HAU configuration on the Extreme Networks S-Series devices. All commands used to configure HAU can be entered in any command mode.

Procedure 4-1 Configuring HAU

Step	Task	Command(s)
1.	Optionally, modify the default HAU mode. By default a standard (non-high availability) upgrade is performed, unless high availability is set when specifying the boot image.	**set boot high-availability default-mode** {**never** | **if-possible** | **always**}
2.	Optionally, modify the default HAU group configuration by placing multiple slots into some or all HAU groups.	**set boot high-availability group** *group-id slot(s)*
3.	Optionally, set a delay between the end of an HAU group upgrade and the beginning of the next HAU group upgrade.	**set boot high-availability delay** *delay*
4.	Specify the system image that will boot the next time the system resets, and optionally specify whether the image will load using the standard or high availability method.	**set boot system** *filename* [standard | high-availability]

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

The following is an example of the HAU aspect of a high availability upgrade configuration, based upon the system setup presented in Figure 4-1 on page 4-2. This configuration example:

- Assumes that VSB and LAGs have been correctly configured
- Leaves the high availability default mode unchanged because the high availability method was specified in the **set boot system** command
- Configures HAU groups:
  - HA group 1, slots 1 and 4
  - HA group 2, slots 2 and 5
  - HA group 3, slots 3 and 6
- Sets the delay between the upgrade completion of one HAU group and the beginning of another to 15 seconds
- Sets the boot system mode to high-availability for image **S-80101-0003**

```
S-Series(rw)->set boot high-availability group 1 1,4
S-Series(rw)->set boot high-availability group 2 2,5
S-Series(rw)->set boot high-availability group 3 3,6
S-Series(rw)->set boot high-availability delay 15
S-Series(rw)->set boot system S-80101-0003 high-availability
```

Terms and Definitions

Table 4-2 lists terms and definitions used in this security mode configuration discussion.

Table 4-2 HAU Configuration Terms and Definitions

Term	Definition
active or current image	The image that is currently running or configured to load on the system
HAU compatibility key	A key associated with each image on the system that must be the same for both the target and active images for a high availability upgrade to take place.
HAU default mode	The boot mode that is used when no boot mode is specified when setting the system boot image.
HAU delay	A configurable delay in seconds between the upgrade completion of one HAU group and the beginning of the next HAU group upgrade.
HAU force complete	A high availability upgrade feature that forces all HAU groups that have not yet upgraded to immediately and simultaneously upgrade.
HAU group	A configuration of one or more slots into a group that results in group members being simultaneously upgraded during a high availability upgrade, leaving slots that are members of all non-upgrading groups operational.
high availability boot mode	A system boot mode option that upgrades HAU groups one at a time, if all HAU preconditions are met.
High Availability Firmware Upgrade (HAU)	An Extreme Networks feature that allows a system to retain high availability while being upgraded to an HAU compatible release. By contrast, all system connectivity and features are temporarily lost during a standard upgrade.
interconnect port	10GbE port that plays the same role as the backplane fabric in a non-VSB chassis by providing distribution between VSB system chassis.
standard boot mode	A system boot mode option that simultaneously upgrades all system slots, taking the system out of operation for the duration of the upgrade.
system boot mode	Determines firmware load or upgrade behavior when the system is powered on or reset.
target image	The image that the system will upgrade to on the next system reset.
Virtual Switch Bonding (VSB)	An Extreme Networks S-Series feature that aggregates two like chassis into a single virtual network device.
VSB chassis	One of two chassis configured for VSB that make up a VSB system.
VSB system	The aggregation of two chassis configured for VSB and connected by one or more 10GbE interconnect ports.

5

Virtual Switch Bonding (VSB) Configuration

This chapter provides information about configuring and monitoring Virtual Switch Bonding (VSB) on S-Series devices.

Using Virtual Switch Bonding in Your Network

Virtual Switch Bonding (VSB) is an S-Series feature that allows for the aggregation of links on two physical chassis, providing redundancy, while at the same time allowing ports on both chassis to pass data concurrently, effectively doubling the available bandwidth. VSB aggregates two like chassis into a single virtual network device. VSB joins two chassis into a single system by extending each chassis' distribution to the other chassis using one or more 10 or 40GbE uplink ports as bonding interconnect links, depending upon your platform.

–

There are two types of VSB interconnect ports depending upon the module and option cards installed:

- Dedicated VSB hardware fabric extended GbE port. Dedicated VSB hardware interconnect ports are not standard Ethernet data ports. They provide a line rate direct connection to the fabric and automatically provide VSB entitlement to the module (no license is required). Dedicated VSB hardware interconnect ports can only be:
  - Used for VSB chassis interconnection
  - Linked to another VSB hardware interconnect port
  - Used in configurations where both chassis are in v2 compatibility mode (Refer to "Chassis Compatibility Mode" on page 9-1 for v2 compatibility mode details.
- Standard software assisted Ethernet data 10GbE port

The following VSB chassis restrictions only apply to VSB systems using standard software assisted Ethernet data interconnect ports, and do not apply to a VSB system using VSB hardware interconnect ports:

- LAG capacities are reduced to 126 on a multi-slot chassis and 61 on an SSA

- Tunneling is not supported
- Remote port mirrors are not supported
- Port Mirroring is limited to 5 mirrors
  - IDS mirroring is not supported
  - Frames can be the subject of one mirror only

Some modules have dedicated hardware VSB interconnect ports that are either fixed or have the ability to install an option module that contains hardware VSB interconnect ports. These dedicated hardware VSB interconnect ports are not compatible with standard software assisted Ethernet data interconnect ports. An interconnect bonding mode specifies whether the system interconnection is hardware or software based. Each physical chassis in the VSB system must be of the same chassis type, which implies the same number of slots. For example, two S4 chassis become a single system with 8 slots; two SSAs become a single system with 2 slots. The interconnect ports connecting the physical chassis are designated as bonding ports on each chassis and create the virtual backplane that ties the two physical chassis together.

The Link Failure Response (LFR) protocol provides for the configuration of one or more monitor links. In the unlikely event that all interconnect links should go down or otherwise fail, the LFR monitor link determines whether both chassis are still operational and places the chassis with the lowest LFR priority in a dormant state until at least one interconnect link is restored. These links do not carry user traffic. The sole purpose of a an LFR link is to resolve which physical chassis should remain operational when all bonding ports are non-functional.

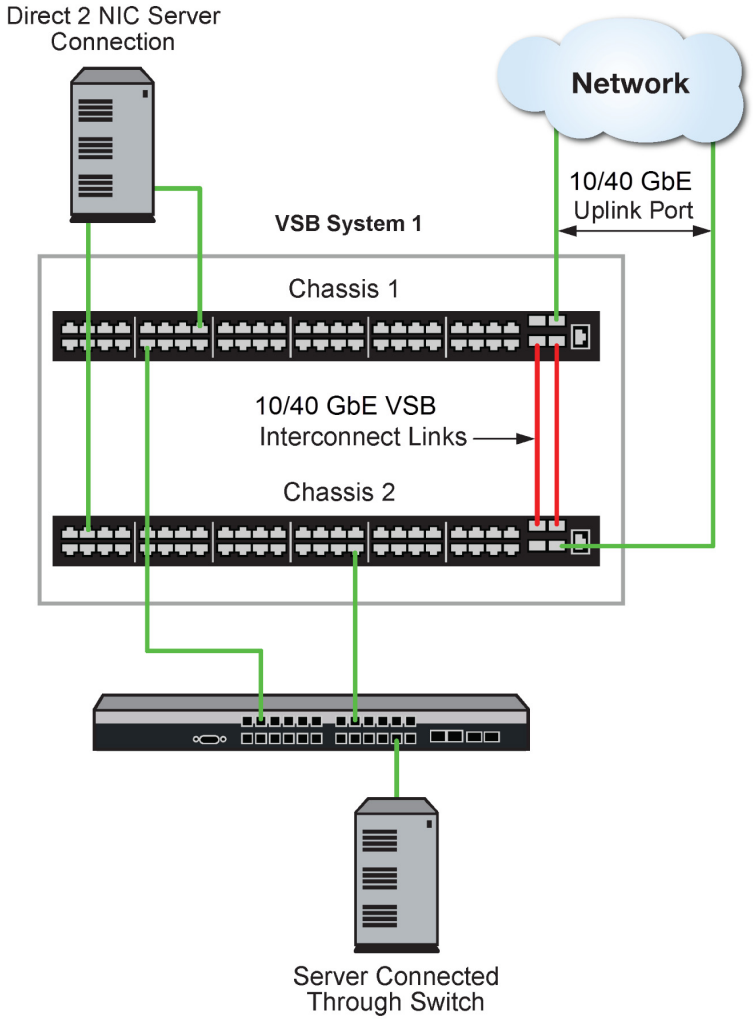
If VSB hardware ports are present on the system, only VSB hardware ports can be used as VSB bond ports. On platforms that do not contain VSB hardware ports, 10GbE software ports enabled for VSB are used as VSB bond ports. All 1GbE software ports enabled for VSB are LFR ports.

VSB is typically used in a data center between two switches, of the same type, LAGed to a server on one side and to network devices on the other.

Figure 5-1 presents an overview of a two SSA chassis VSB system in a data center context. The upper chassis is configured as VSB chassis 1. The lower chassis is configured as VSB chassis 2. These chassis are members of VSB system 1. On each chassis, uplink ports are used to create the interconnect links that aggregate the two chassis into a single virtual network device. Non-VSB 10GbE ports provide the uplink connections to the network.

In Figure 5-1, a server with two NICs installed achieves redundancy through a direct connection, distributing the two NIC connections between the two available VSB system slots. A single NIC server achieves redundancy through an intermediate switch that distributes multiple connections between the two VSB system slots.

Figure 5-1 VSB Data center Configuration Overview



A VSB system is managed by a single IP address and behaves as if it is one chassis with double the slots. Once globally enabled, VSB system IP address configuration is the same as for a non-VSB system.

The VSB system provides gains in network resiliency, performance, and management through:

- A greater number of ports in a single system
- Location redundancy
- Added bandwidth

Though bandwidth and port capacities scale in a VSB system, feature capacities such as route, MAC address tables and user capacities remain the same as in a single chassis system.

VSB licensing is required per chassis unless the chassis contains at least one module that is automatically entitled to VSB. See the release notes that come with the firmware for module VSB entitlement details.

Using the High Availability feature, slots can be grouped such that forwarding is maintained while individual slot groups are firmware upgraded, so long as a group that is not being upgraded continues to provide the interconnect between VSB chassis.

A non-default MAC address can be manually set prior to globally enabling the VSB system.

An outport local preference can be configured for the local chassis, setting a likelihood that a packet will egress the system using a LAG port on the local chassis and not utilize the VSB interconnect link.

A preference for local chassis LAG ports can be set to none, weak, strong or forced.

Implementing VSB

To implement VSB:

1. Select two chassis of the same platform and model that will be the physical chassis members of the VSB system.
2. Configure each chassis' VSB chassis ID and the ID of the VSB system the two chassis belong to.
3. Assure that the appropriate chassis compatibility mode is set for each VSB chassis. Chassis compatibility mode need not be the same for each chassis.
4. Determine the slots that will be used for the VSB connection and identify the 10GbE ports that will be used to interconnect the chassis. One VSB connection is required between the two chassis; a minimum of two VSB interconnections is highly recommended. VSB interconnect ports should be selected taking into consideration the optimization of bandwidth usage and redundancy.
5. Set the bonding mode appropriate to the type of interconnect port used (hardware or software).
6. Enable bonding on the selected GbE interconnect ports.
7. Optionally, enable bonding on GbE monitor ports.
8. Validate VSB feature entitlement by activating licenses on any modules not automatically entitled to VSB. See the release notes that come with the firmware for VSB entitlement information.
9. Optionally change the VSB MAC address from the default value. The VSB MAC address defaults to an internal MAC address associated with chassis 1.
10. If LFR monitor ports were enabled, enable LFR on the system.
11. After completing steps 1 — 8, globally enable the VSB system. The system is reset. If the following two optional steps are configured prior to enabling VSB, the configuration is lost.
12. Optionally configure the VSB system for LACP outport local preference.
13. Optionally configure the VSB system for High Availability firmware upgrades.

VSB Configuration Overview

For information about...	Refer to page...
Globally Enabling and Disabling the VSB System	5-9
LACP Local Preference Configuration	5-9
High Availability Firmware Upgrade	5-11
Applying a VSB Configuration File to a Replacement VSB Chassis	5-12

VSB Chassis Configuration

VSB chassis configuration must take place before the system is VSB globally enabled. Prior to a VSB system being globally enabled, you can modify the VSB chassis configuration as needed. Once a VSB system is globally enabled using the **set bonding** command (See "Globally Enabling and Disabling the VSB System" on page 5-9), VSB specific chassis configuration can not be modified. VSB must be disabled globally, reverting each chassis to a non-VSB state, in order to make modifications to the VSB chassis configuration.

There are two values associated with a VSB chassis configuration:

- Chassis ID – Identifies the physical chassis member of the VSB system.
- System ID – Identifies the VSB system made up of two physical chassis

VSB chassis ID valid values are **1** and **2**. The VSB chassis ID both identifies the VSB chassis and determines chassis slot assignments within a globally enabled VSB system. After the chassis ID and system ID have been configured, but prior to globally enabling VSB, slot numbering does not differ from standard slot numbering.

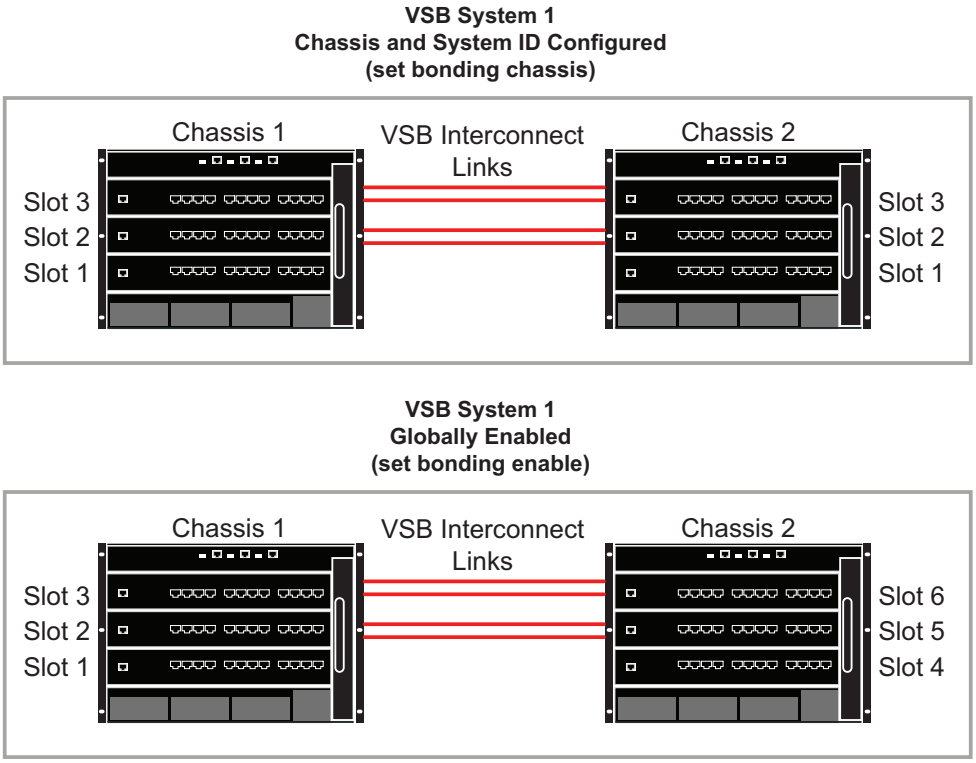
Once the system has been globally enabled for VSB :

- Chassis 1 slots are numbered the same as a non-VSB physical chassis.
- Chassis 2 slots are numbered starting with the maximum number of supported chassis slots plus 1.

For example, when bonding two three slot chassis, chassis 1 slots are numbered 1, 2 and 3, and chassis 2 slots are numbered 4, 5 and 6.

Figure 5-2 displays a three slot chassis based VSB system with slot numbering for both a pre-VSB globally enabled system when only the chassis and system ID are configured and a VSB globally enabled VSB system.

Figure 5-2 VSB Slot Numbering



The VSB system ID is configured with the same value on each chassis in a given VSB system. The VSB system ID is used to identify the bonded system as a whole. VSB does not enforce unique VSB system ID values between systems, but it is highly recommended that each system be configured with a unique value for management purposes. Valid VSB system ID values are **1 — 18446744073709551615**.

A VSB secret can be configured to add security to the VSB links. The VSB secret can be up to 32 printable characters long. If a space is used, the secret must be enclosed in double quotes (“”). The secret can be created or modified at any time. The secret can be overwritten without first clearing it. If a secret only exists on a single chassis, the VSB system will segment until the same secret is present on both physical chassis in the system. A VSB secret is configured using the **secret** option of the **set bonding chassis** command.

Use the **set bonding chassis** command to configure a VSB chassis.

Use the **clear bonding chassis** command to clear the VSB chassis configuration, only if VSB is not globally enabled on the chassis.

VSB Interconnect Link Configuration

In a non-VSB single chassis system, the chassis backplane provides distribution between chassis slots. In a VSB enabled system, GbE port interconnections provide distribution between chassis.

There are two types of VSB interconnect ports depending upon the module and option cards installed:

- Dedicated VSB hardware fabric extended 10GbE port.

- Standard software assisted Ethernet data 10GbE port

Dedicated VSB hardware interconnect ports have their own naming convention: **vsb**.*x*.*y* where *x* specifies the slot number and *y* specifies the port number. VSB hardware interconnect ports are designated by a blue outline and labelled "Bonding Port". If a VSB hardware interconnect port is present, the module does not require a VSB license.

Configure the interconnect port type being used by setting the interconnect port bonding mode to **hard** for VSB hardware interconnect ports or **soft** for standard Ethernet data interconnect ports using the **set bonding mode** command. VSB interconnect port bonding mode defaults to **soft**.

Interconnect link support depends upon whether the platform has dedicated VSB hardware ports present. If present, only VSB hardware ports are supported for VSB interconnect links. If not present, 10GbE ports enabled for VSB are supported for VSB interconnect links.

It is recommended that you consider optimization of interconnect redundancy and optimization of bandwidth when determining the number and location of VSB interconnect links configured between bonded chassis.

Configure the VSB chassis using the **set bonding chassis** command (see "VSB Chassis Configuration" on page 5-5 for details) before enabling VSB interconnect ports.

Use the **set bonding port enable** command, specifying the interconnect port, to enable the VSB interconnect port. The VSB interconnect port must be enabled on both chassis. Port designation is based upon the standard physical chassis slot and port designation until VSB is globally enabled for the VSB system (See "Globally Enabling and Disabling the VSB System" on page 5-9). A minimum of one VSB interconnect link must be configured between bonded chassis before globally enabling VSB. Once VSB is globally enabled, VSB interconnect links can be added or modified.

See "VSB Chassis Configuration" on page 5-5 for details concerning chassis slot numbering before and after a VSB system is globally enabled.

In a VSB system of two three slot chassis that is globally enabled:

- Chassis 1 ports for the bottom slot are specified as **fg.1.***x* where *x* is the port number (followed by slots 2 and 3)
- Chassis 2 ports for the bottom slot are specified as **fg.4.***x* where *x* is the port number (followed by slots 5 and 6)

When modifying interconnect ports in a globally enabled VSB system, use the globally enabled port designation to specify ports.

Link Failure Response (LFR) Configuration

The LFR protocol determines which chassis will be brought down should all VSB interconnect links between the VSB chassis go down, and it is determined that both VSB chassis are operational. Both chassis in an operational VSB system use the same IP address and function as a single system with the GbE interconnect links acting as a virtual backplane for the system. Should all VSB interconnect links go down and both chassis remain operational, the two physical chassis would function as independent network devices with the same IP address.

The LFR protocol allows GbE ports to be designated as VSB monitor links that operate in a standby mode to the primary GbE VSB bond ports. The VSB monitor link provides dedicated redundant control plane connectivity and is used only as a backup communication path between two bonded chassis in the unlikely event that all of the primary VSB interconnect links fail or become unavailable. When the primary GbE VSB bond ports are down, the VSB monitor links facilitate a communications path to allow the physical chassis with highest LFR priority in the bonded pair to remain active while placing the chassis with the lower priority into a dormant state, except for all bonded links which maintain current state.

The LFR protocol must be globally enabled on each VSB chassis in the VSB system for LFR monitoring to occur. Use the **set bonding lfr enable** command to globally enable LFR on each physical chassis.

The LFR monitor port is configured using the **set bonding port enable** command, the same as a VSB interconnect port. What distinguishes the port types in a VSB context is the port speed. All 1GbE software ports enabled for VSB are LFR ports.

The VSB feature supports a combined total of 32 VSB GbE interconnect and LFR GbE monitor links on a VSB system (32 VSB ports per chassis).

The physical chassis to be placed in dormant state is determined by the LFR priority. A chassis' LFR priority defaults to 10 times the VSB chassis ID. For example, if the VSB chassis ID is **1**, the LFR priority is **10**. The LFR priority can be manually set using the **set bonding chassis** command **lfr-priority** parameter with a valid range of **1** - **255**. Setting a duplicate LFR priority is not allowed.

Use the **show bonding** command to review LFR state and port configuration details.

VSB System MAC address Configuration

By default, the VSB system MAC address is set to an internal MAC address associated with VSB chassis 1. You can manually set a MAC address for the VSB system using the **set bonding mac** command.

Note: The VSB system MAC address can not be changed while VSB is globally enabled on the system. You must disable VSB using the set bonding disable command before you can modify a globally enabled VSB system MAC address. Disabling VSB on a globally enabled system clears the configuration on both system chassis.

Prior to globally enabling VSB on the system, you can reset a manually configured VSB system MAC address to the default value using the **clear bonding mac** command.

Licensing

VSB entitlement is by chassis. If one module in the chassis is entitled to VSB, the chassis is VSB entitled. VSB is supported by default on the S155 and S180 module, and any module that has a VSB hardware interconnect port present. A license is required when globally enabling VSB on chassis containing only S130 and S150 modules. When a VSB enabled S155 module is hot swapped with an S130 or S150 module or a license expires, VSB is not disabled on the new or license expired module. If no other module in the chassis is entitled to VSB, an error message will display every five minutes until a valid license is configured on the module.

The displayed error message will be similar to:

```
System[2]Virtual Switch Bonding (VSB) is enabled on chassis-2 without a valid
license. This message will continue until the license is configured or VSB is
disabled.
System[1]Virtual Switch Bonding (VSB) is enabled on chassis-1 without a valid
license. This message will continue until the license is configured or VSB is
disabled.
```

A feature entitlement verification check occurs when attempting to enable VSB globally. If only modules not entitled to VSB are installed in the system, you must activate a VSB license on each chassis in the system before attempting to globally enable VSB.

Use the **set license vsb** command, specifying the license key and VSB chassis ID, to configure the module with a valid license. The **show license** command displays license status per chassis, before the VSB system is globally enabled, or for both chassis once the VSB system is globally enabled.

Globally Enabling and Disabling the VSB System

Note: Do not attempt to globally enable the VSB system before:

- Chassis IDs and the VSB system ID are assigned
- At least one interconnect port is VSB enabled
- Any required licenses are activated
- MAC address is assigned, if a non-default MAC address will be used

Chassis configured as bonded chassis with a chassis ID, system ID, and configured with one or more interconnect ports, maintain a status as individual physical chassis until globally enabled for VSB.

Once a chassis is globally enabled for VSB, you can no longer modify VSB configuration for:

- Chassis ID
- System ID
- Mac Address

Fan, PoE and power supply resources are chassis bound. In a VSB globally enabled system, hardware system statistics are displayed per VSB chassis using the **show system hardware** command. PoE capacities are not shared between VSB chassis. PoE resources are assigned to modules using the **set inlinepower assigned** command and viewed using the **show inlinepower** command, just as you would in a non-VSB system.

Use the **set bonding enable** command to globally enable the VSB system.

Use the **set bonding disable** command to disable a VSB system.

Note: Disabling a VSB system clears all non-VSB configuration on both chassis in the system. VSB licensing and a non-enabled VSB configuration persists. If you wish to clear the VSB chassis and system ID configuration after disabling VSB, use the **clear bonding chassis** command.

LACP Local Preference Configuration

LACP Outport local preference is a VSB only feature that increases the likelihood that the packet flow will egress the system using a LAG port on the local chassis and not utilize the VSB interconnect link. When ports for the same LAG are configured on both VSB system chassis, the hash that determines the egress port does not take into consideration interconnect link utilization when choosing the egress port. This inability to determine interconnect link utilization can cause unnecessary use of the interconnect link and in the worse case scenario can lead to interconnect link saturation. The outport local preference for a chassis can be configured. Local chassis outport local preference determines the likelihood that a packet will egress the VSB system using a LAG port on the local chassis. To the degree an ingressing packet egresses the local chassis, packet utilization of the VSB interconnect links is minimized.

There are four outport local preference levels. If set to **none**, the outport local preference feature has no impact on packet egress for the VSB system. The **weak**, **strong**, and **all-local** outport local preference settings provide an increasing likelihood that the packet will egress the VSB system using a LAG port on the local chassis if resources are available. The degree of likelihood is from a somewhat greater likelihood in the case of **weak**, to an attempt to force all packets to use a local chassis LAG port in the case of **all-local**.

Figure 5-3 and Figure 5-4 on page 5-11 present a likely packet flow example between two users within an LACP outport local preference configuration context. User A is directly attached to switch 1. User B is directly attached to chassis 1 of VSB system 1. Both chassis 1, chassis 2, and switch 1 have ports configured with LAG 1.

Figure 5-3 presents likely packet flows between User A and User B if LACP outport local preference is set to **none**. Because Switch 1 has access to both chassis 1 and chassis 2 on LAG 1, Packet flows from User A to User B may use LAG 1 ports on either VSB chassis 1 or 2. It is not possible to prevent packet flows in this direction from utilizing VSB system 1 interconnect links. Because the outport local preference is set to none on VSB system 1, the same is true for any flows from User B to User A.

Figure 5-3 Outport Local Preference Set to None

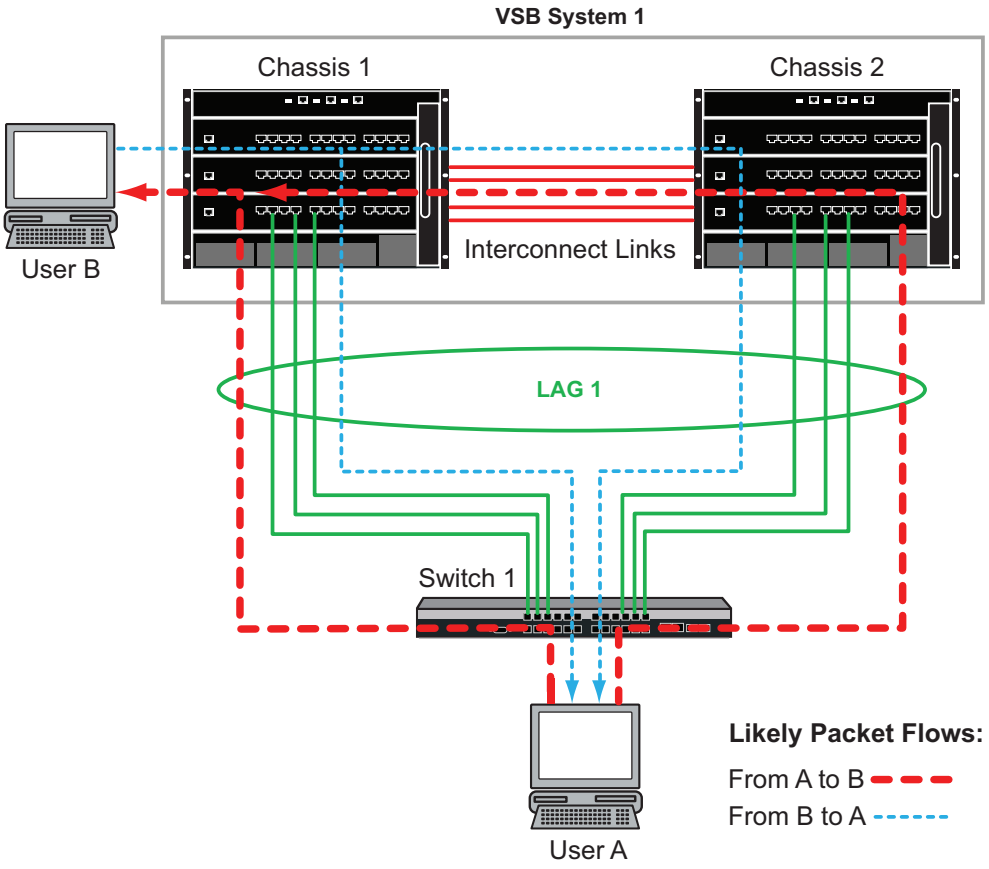
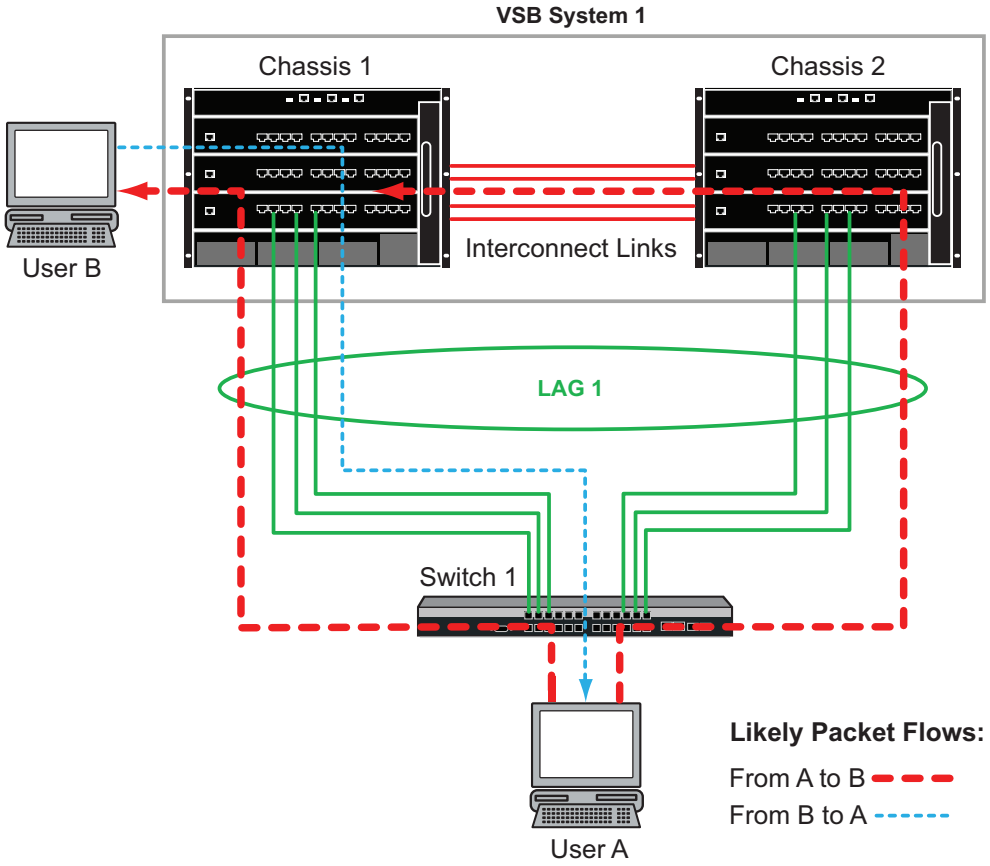


Figure 5-4 presents likely packet flows between User A and User B if LACP outport local preference is set to **all-local**. Once again, because Switch 1 has access to both chassis 1 and chassis 2 on LAG 1, Packet flows from User A to User B may use LAG 1 ports on either VSB chassis 1 or 2. Because the outport local preference is set to **all-local** on VSB system 1 and User B is directly connected to chassis 1, chassis 1 will always be the local chassis. The **all-local** preference setting will always attempt to force packet flows to use chassis 1 LAG 1 ports for User B to User A packet flows.

Figure 5-4 Outport Local Preference Set to All-Local



Note: Local chassis outport local preference is a VSB only feature. The outport local preference setting is only in affect when when VSB is globally enabled on the system.

LACP outport local preference can be configured at any time. If configured before globally enabling VSB, you must configure it on both chassis with the same option. If you configure it after VSB is globally enabled, it is configured globally for the VSB system.

Use the **set lacp outportLocalPreference** command to configure outport local preference for the VSB system.

Use the **clear lacp outportLocalPreference** command to reset outport local preference to the default value of **none** for the VSB system.

High Availability Firmware Upgrade

High Availability firmware Upgrade (HAU) is supported for S-Series VSB configurations whether the chassis is single or multiple slot. Refer to Chapter 4, **High Availability Firmware Upgrade (HAU) Configuration** for HAU configuration details. Refer to "High Availability Firmware Upgrade in a Virtual Switch Bonded System" on page 4-7 for HAU configuration details specific to a VSB configuration.

Applying a VSB Configuration File to a Replacement VSB Chassis

A VSB configuration file contains chassis specific information such as the chassis serial-number. When configuring a replacement VSB chassis with an already existing VSB configuration file, you must specify the chassis ID of the replacement chassis when entering the **configure** command so that chassis specific information on the specified chassis will be ignored and replaced in the configuration file with the correct chassis settings. The chassis ID is specified using the **chassis-id** option.

Configuring VSB

This section provides a table of VSB default values and a procedure for configuring a VSB system.

Table 5-1 lists VSB default values.

Table 5-1 Default VSB Parameters

Parameter	Description	Default Value
Chassis ID	Identifies a VSB system chassis.	None.
System ID	Identifies a VSB system made up of two VSB chassis.	None.
LFR priority	Used to determine the physical chassis to be put in dormant state should all VSB interconnect links go down.	10 times the chassis ID.
LFR state	Specifies whether LFR is globally enabled or disabled on the physical chassis.	Disabled.
MAC address	VSB system MAC address	An internal MAC address associated with VSB chassis 1.
outport local preference	LACP VSB only feature that determines the likelihood of a packet that ingresses the local chassis egressing the VSB system using a local chassis LAG port.	None.
VSB state	Specifies whether VSB is globally enabled or disabled on the physical chassis.	Disabled.

Procedure 5-1 describes VSB configuration on the Extreme Networks S-Series devices. All commands used to configure VSB can be entered in any command mode.

Procedure 5-1 Configuring VSB

Step	Task	Command(s)
1.	On each chassis, configure each chassis' VSB chassis ID and the ID of the VSB system the two chassis belong to. Optionally: • Configure an encrypted secret used on the VSB links • Administratively change the LFR priority for the physical chassis	**set bonding chassis** *chassis-id* {**system-id** *system-id* | **secret** *secret* | **lfr-priority** *priority*}

Procedure 5-1 Configuring VSB (continued)

Step	Task	Command(s)
2.	Validate VSB feature entitlement by activating licenses on any S150 or S130 modules (S155 modules support VSB by default). Enclose the license key in double quotes (“”).	**set license vsb** *license-key*
3.	Optionally change the VSB system MAC address from the default value.	**set bonding mac** *mac-address*
4.	Set the VSB interconnect port bonding mode.	**set bonding mode** {**hard** | **soft**}
5.	Enable bonding on the: • 10GbE ports to be used as VSB interconnect ports. • 1GbE ports to be used as LFR monitor ports	**set bonding port** [*port-string*] **enable**
6.	Globally enable LFR on the physical chassis.	**set bonding lfr enable**
7.	Globally enable the VSB system.	**set bonding enable**
8.	Optionally configure LACP outport local preference on the VSB system.	**set lacp outportLocalPreference** {**none** | **weak** | **strong** | **all-local**}
9.	Optionally configure the VSB system for High Availability firmware upgrades	See “High Availability Firmware Upgrade” on page 5-11.

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

Terms and Definitions

Table 5-2 lists terms and definitions used in this VSB configuration discussion.

Table 5-2 VSB Configuration Terms and Definitions

Term	Definition
Virtual Switch Bonding (VSB)	An S-Series feature that aggregates two like chassis into a single virtual network device.
Link Failure Response (LFR)	A VSB protocol that monitors the VSB interconnect links and determines the physical chassis to be put in a dormant state should all interconnect links go down.
interconnect port	GbE port that plays the same role as the backplane fabric in a non-VSB chassis by providing distribution between VSB system chassis. Interconnect ports can be dedicated VSB hardware or standard Ethernet data ports.
monitor port	GbE port used by the LFR protocol to place the lower priority chassis in a dormant state should all VSB interconnect links go down.
VSB chassis	One of two chassis configured for VSB that make up a VSB system.
VSB system	The aggregation of two chassis configured for VSB and connected by one or more GbE interconnect ports.
VSB MAC address	The MAC address for the VSB system that is either manually assigned or defaults to a VSB chassis 1 internal MAC address.
outport local preference	An LACP VSB only feature that configures a likelihood that a packet will egress the system using a LAG port on the local chassis and not utilize the VSB interconnect link.

6

Port Configuration

This document describes port configuration on Extreme Networks S-Series devices.

Port Configuration Overview

The Extreme Networks S-Series modules and standalone devices have fixed front panel switch ports and, depending on the model, optional expansion module slots. The numbering scheme used to identify the switch ports on the front panel and the expansion module(s) installed is interface-type dependent and is also dependent upon the chassis slots in which the module(s) are installed. Port numbering proceeds from 1 to the maximum number of that port type on the module. If there are multiple port types, each port type numbering starts at 1. Port numbering is displayed next to each port.

When configuring a port, the port string associated with a port is made up of the port type, the slot location of the module in the chassis and the port number delineated by a period as explained in Port String Syntax Used in the CLI.

The following topics are covered in this section:

Port String Syntax Used in the CLI

Commands requiring a *port-string* parameter use the following syntax to designate the type of port being configured, slot location the module containing the port is inserted into the chassis, and port number on the module containing the port:

port type.slot location.port number

Where **port type** can be:

- **ge** - 1-Gbps Ethernet
- **tg** - 10-Gbps Ethernet
- **fg** - 40-Gbps Ethernet
- **com** - COM (console) port
- **host** - the host port
- **vlan** - VLAN interfaces
- **tun** - Layer 3 tunnel
- **lag** - IEEE802.3 link aggregation ports
- **lpbk** - loopback interfaces, or
- **lo** - the local (software loopback) interface
- **vtap** - a MIB-II interface for VLANs, used as the data source input of a port mirror or SMON statistics collection on that particular VLAN
- **vsb** - a Virtual Switch Bonding (VSB) hardware interconnect port

Slot location for modules installed in a S-Series **chassis can be:**
0 through the maximum number of slots in the chassis, with **0** designating virtual system ports (lag, vlan, host, loopback), and **1** designating the lowest module slot in the chassis.

Port number can be:
Any port number on the module. The highest valid port number is dependent on the number of ports in a slot location and the port type.

For example:
If a module in slot 1 has 48, 1GbE front panel ports, and an uplink interface with 6 Mini GBICs, the range of port number designations used in the CLI command would be:

ge.**1.1** through ge.**1.48** for the 48 1GbE front panel ports, and
tg.1.1 through **tg.1.6** for the 6 10GbE uplink ports.

If the uplink has the same type (**ge**) ports as the front panel, the numbering continues with the port number ge.**1.49**.

Examples

Note: You can use a wildcard (*) to indicate all of an item. For example, ge.1.* would represent all 1GbE ports in the module in slot 1.

This example shows the *port-string* syntax for specifying the 1GbE port 14 in the module in chassis slot 3:

```
ge.3.14
```

This example shows the *port-string* syntax for specifying ports 1, 3 and 11 in the module in chassis slot 1:

```
ge.1.1;ge.1.3;ge.1.11
```

This example shows the *port-string* syntax for specifying ports 1, 3, 7, 8, 9 and 10 in the module in chassis slot 1:

```
ge.1.1,ge.1.3,ge.1.7-10
```

This example shows the *port-string* syntax for specifying the 10-GbE port 2 of the module in chassis slot 3:

```
tg.3.2
```

This example shows the *port-string* syntax for specifying the Virtual Switch Bonding hardware interconnect port 2 of the module in chassis slot 1:

```
vsb.1.2
```

This example shows the *port-string* syntax for specifying all 1GbE ports in the module in chassis slot 1:

```
ge.1.*
```

This example shows the *port-string* syntax for specifying all 10-Gbps Ethernet ports in the chassis:

```
tg.*.*
```

This example shows the *port-string* syntax for specifying all 40-Gbps Ethernet ports in the chassis:

```
fg.*.*
```

This example shows the *port-string* syntax for specifying all ports (of any interface type) in all modules in the chassis:

```
*.*.*
```

Console Port Parameters

Each Extreme Networks S-Series module or standalone device includes a console port through which local management of the device can be accessed using a terminal or modem. The CLI provides for:

- The display of console port configurations using the **show console** command in any command mode
- The setting of console port parameters, including the baud rate, flow control, number of bits, number of stop bits and parity, using the **set console** command in any command mode
- The clearing of console port parameters to default values using the **clear console** command in any command mode

If C2 security mode is enabled, You can not create, modify, or clear a console configuration while in Read-Write user mode.

When specifying a console port string, use the **com** keyword for the port type, as specified in the Port String Syntax Used in the CLI discussion.

The following example shows how to set the baud rate to 19200 on console port com.1.1:

```
S Chassis(rw)->set console baud 19200 com.1.1
```

The following example shows how to set the bits property value to 8 on all console ports:

```
S Chassis(rw)->set console bits 8
```

The following example shows how to set the flowcontrol property value to none on console port com.1.1:

```
S Chassis(rw)->set console flowcontrol none com.1.1
```

The following example shows how to set the parity property value to even on all ports:

```
S Chassis(rw)->set console parity even
```

The following example shows how to set the stopbits property value to one on console ports com.1.1 and com.1.2:

```
S Chassis(rw)->set console stopbits one com.1.1-2
```

Administratively Enabling a Port

Ports are administratively disabled by default.

Use the **set port enable** command to administratively enable the specified ports.

Use the **set port disable** command to administratively disable the specified ports.

The following example administratively enables port ge.1.1:

```
S Chassis(rw)->set port enable ge.1.1
S Chassis(rw)->show port ge.1.1
Port ge.1.1 enabled
S Chassis(rw)->
```

Ingress Filtering

The ingress filtering feature provides for a means of limiting the forwarding of received frames on the ingress port based on the VLAN egress list for that port. VLAN IDs of a port's incoming frames are compared to the port's egress list. If the received VLAN ID does not match a VLAN ID on the port's egress list, the frame is dropped. See Chapter 24, **VLAN Configuration** for VLAN egress list information. Ingress filtering is disabled by default.

Use the **set port ingress-filter** command in any command mode to enable ingress filtering on the specified ports.

The following example enables ingress filtering on port ge.1.1

```
S Chassis(rw)->set port ingress-filter ge.1.1 enable
S Chassis(rw)->>show port ingress-filter ge.1.1
Port          State
------------  --------
ge.1.1        enabled
```

Port Alias

The alias feature allows a string name to be associated with a port.

Use the **set port alias** command to configure an alias for the specified ports.

The following example sets the alias on port ge.1.1 to **documentation**

```
S Chassis(rw)->set port alias ge.1.1 documentation
S Chassis(rw)->>show port alias ge.1.1
Alias on port ge.1.1 set to:  Documentation.
```

Force Linkdown

When the force linkdown feature is disabled, disabling a port using **set port disable** will disable the ability to forward traffic, but the link stays up. When force linkdown is enabled, disabling a port using **set port disable** will disable the link completely.

When force linkdown is enabled, disabling a port using the **set port disable** command will not disable PoE on that port.

Force linkdown is disabled by default.

Use the **set forcelinkdown** command in any command mode to enable the force linkdown feature on this device.

The following example enables the force linkdown feature on this device:

```
S Chassis(rw)->set forcelinkdown enable
S Chassis(rw)->show forcelinkdown
ForceLinkDown feature is globally enabled.
S Chassis(rw)->
```

Default Port Speed

When auto-negotiation is enabled, the port speed used is determined by the fastest compatible speed between linked ports. On ports capable of multiple speeds, if auto-negotiation is not enabled, the default port speed setting provides for the configuring of a default speed for this port. Use the **set port speed** command to specify the default speed for the specified ports. Valid values are 10, 100, 1000, 10000, and 40000 Mbps.

Auto-negotiation is enabled by default.

The following example sets the default speed on port ge.1.1 to 100 Mbps:

```
S Chassis(rw)->set port speed ge.1.1 100
S Chassis(rw)->show port speed ge.1.1
default speed is 100 on port ge.1.1.
S Chassis(rw)->
```

The QSFP Port

QSFP ports support operation as a 1x40G port or 4x10G ports.

QSFP port speeds operate in port speed group pairs. For a six QSFP module, there are three port speed group pairs:

- Port Speed Group 1 – ports 1 and 2

- Port Speed Group 2 – ports 3 and 4
- Port Speed Group 3 – ports 5 and 6

All ports in a speed group must be set to the same speed.

Port speed groups can operate in either 40Gbps or 10Gbps mode. A 40Gbps port is identified using the **fg.***x*.*y* format where x identifies the chassis slot and y identifies the module port. A 10Gbps port is identified using the **tg.***x*.*y* format.

When displaying ports on the QSFP card, all possible ports are displayed in show command output. For a six QSFP module, ports **fg.***x***.1-6** and **tg.***x***.1-24** (four 10Gbps ports for each of six 40Gbps ports) are displayed. The ports not associated with the active operating speed are displayed with an oper-status of **not-present**. For example, if port speed group one made up of fg.1.1-2 and tg.1.1-8 is running in 40Gbps mode, the display will show ports tg.1.1-8 as not-present.

Note: A module must be reset when modifying the port speed of a QSFP port for the new speed to become operational.

Many QSFP devices support operation in both 10Gbps and 40Gbps.These include QSFP assemblies with fixed cable assemblies that have QSFP terminations at both ends of the assembly.

At this time of this writing, only the QSFP to 4x SFPP “hydra” cable assemblies, which terminate one end with a SFPP, and QSFP to single SFPP adapters must operate in 10Gbps.

If optical QSFP transceivers with detachable fiber cable are to be used, they must be Extreme Networks brand. See the release notes that come with your firmware version for QSFP transceiver support details.

Changing the QSFP Port Speed

There are three ways in which a QSFP port speed can be changed:

- Using the set port speed command in any command mode
- Setting a new value for the corresponding MIB object
- Inserting a QSFP that supports a single operating speed that does not conflict with an already installed QSFP in the port speed group

Using the set port speed Command

When using the **set port speed** command to change the QSFP port speed, all ports in the port speed group are configured for the new port speed mode. This command only affects ports that have an operational status of present (any status not not-pres). If the **show port status** command displays the operational status as **not-pres**, using this command will have no affect.

To change the operational speed for the QSFP ports in the port speed group with fg.4.1 in the present state from 40Gbps to 10Gbps, first verify that ports fg.1.1-2 are in a present state using the show port status command:

```
S Chassis(su)->show port status *.4.1-2

Port         Alias            Oper     Admin   Speed  Duplex  Type
             (truncated)      Status   Status  (bps)
------------ ---------------- -------- ------- ------ ------- ------------------
tg.4.1                        not-pres up                     unknown
tg.4.2                        not-pres up                     unknown
```

```
tg.4.3                          not-pres up                     unknown
tg.4.4                          not-pres up                     unknown
tg.4.5                          not-pres up                     unknown
tg.4.6                          not-pres up                     unknown
tg.4.7                          not-pres up                     unknown
tg.4.8                          not-pres up                     unknown
com.4.1                         down     up         9.6K        rs232        usb
fg.4.1                          down     up        40.0G full   unknown
fg.4.2                          down     up        40.0G full   unknown
11 of 11 ports displayed, 0 port(s) with oper status 'up' or 'dormant'.
S Chassis(su)->
```

The output shows that the 10Gbps members of the port speed group are not present and that the 40Gpbs members are present and in an operational status down state. The port speed is changed for both members of the port speed group by specifying either port or both ports in the command entry. To change the speed to 10Gbps for the port speed group containing ports fg.4.1-2 enter either port or both along with the new speed, enter:

```
S Chassis(su)->set port speed fg.4.1 10000
S Chassis(su)->
```

You must reset the module for the new speed to take affect only after the reset. Use the show port status command again to confirm the speed change. Ports fg.4.1-2 will now display as not-pres and ports tg.4.1-8 will have a present status of either up or down.

To change the operational speed for the QSFP ports in the port speed group with tg.4.1 in the present state from 10Gbps to 40Gbps, enter:

```
S Chassis(su)->set port speed tg.4.1 40000
S Chassis(su)->
```

The operational status for ports tg.4.1-8 will be set to not-pres and the operational status for ports fg.4.1-2 will be set to present (up or down) upon resetting the module.

Inserting a Single Speed QSFP

The port speed of a speed group member can be changed by inserting a QSFP that supports a single operating speed that does not conflict with the operating speed of the other member of the port speed group. A conflict is defined as a QSFP in the port for the speed you wish to change requiring a different speed than is currently operating and there is a QSFP installed in the other member of the port speed group that is compatible with the current operating speed you wish to change. If there is conflict Syslog reports the conflict.

Port Speed Change Prior to Reset

After a speed change has been set, but prior to the reset for the module containing the port:

- The system reports a Syslog message indicating the module containing the port must be reset for the new speed change to take affect
- All ports, in the port speed group associated with the new operating speed, remain in the **not-pres** state until module resets
- The ports, in the port speed group not associated with the new desired operating speed, go into an oper-status down state with oper-status cause specified as **self** (system initiated)

Use the **show port operstatus** command to display the current port oper-status cause.

The module containing the modified ports must be reset to complete the speed transition.

Retracting a Requested Speed Change

To retract a requested speed change, insert or remove and reinsert a QSFP in the port speed group that can operate in the original speed. Upon inserting or reinserting the QSFP:

- Ports will no longer be held in the oper-status **down** state with the **self** cause and will return to normal operation immediately
- The speed change scheduled for the next reset is cancelled
- The show port commands revert to the status prior to the speed change request

Port Duplex

Duplex between two communicating devices specifies whether communication will be one way at a time (half-duplex) or in both directions simultaneously (full-duplex). When auto-negotiation is enabled, auto-negotiation determines port duplex.

Use the **set port duplex** command to specify whether the specified ports will operate at half or full duplex when auto-negotiation is not enabled.

The following example sets the port duplex on port ge.1.1 to full:

```
S Chassis(rw)->set port duplex ge.1.1 full
S Chassis(rw)->show port duplex ge.1.1
default duplex mode is full on port ge.1.1.
S Chassis(rw)->
```

Jumbo Frames

The jumbo frames feature supports Ethernet frames greater than 1500 bytes of payload on a port. By default, jumbo frame support is disabled on all ports and path MTU discovery is enabled. When jumbo frame support is enabled, path MTU discovery should also be enabled. Path MTU discovery is set using the **set mtu** command.

It is possible for jumbo administrative status to be enabled and jumbo operational status to be deferred. Jumbo frame support is supported on all module ports, but some modules can only handle 12 jumbo enabled ports at one time. If on a module that only supports 12 jumbo frame ports, you enable jumbo frames, without specifying a port, by default, the first 12 ports are enabled. Should you then enable a port number higher than 12, it will immediately take back the jumbo frame resources for ports 1 - 12 and initially show an operational status of deferred. Resetting the module will enable deferred ports.

When enabling jumbo frame support, the maximum frame size defaults to 10239 bytes for untagged packets and supports 10243 bytes for tagged packets. Jumbo frame size can be set to any value between 1000 – 10239 bytes. The standard Ethernet frame MTU for untagged packets is 1518 bytes including the 18 Ethernet header bytes. The standard Ethernet frame MTU for tagged packets is 1522 bytes.

When setting the jumbo frame size, keep in mind that large frames require more packet buffers than standard frames, reducing the total available packet buffers. This limitation can be improved by selecting a smaller MTU jumbo frame size.

Some applications require extra header bytes beyond the standard tagged packet size. For example, provider bridging (Q-in-Q) requires an extra header of 4 additional bytes. The S-Series supports the dynamic MTU frame which when enabled will automatically add the extra 4 bytes to the currently configured jumbo MTU frame size, allowing you to set the jumbo MTU frame size to the standard 1518 bytes and the 4 additional tagged header bytes will automatically be added for Q-in-Q frames. The dynamic MTU frame feature currently only supports Q-in-Q. Other

applications requiring additional header bytes will be added in future releases. Dynamic MTU frame is disabled by default.

Use the **show port jumbo** command to verify the operational status of a jumbo enabled port including display of supported jumbo MTU frame size and the state of the dynamic MTU frame feature. A jumbo administratively disabled port will always have a jumbo operational status of disabled.

If you have manually enabled jumbo frames support on the maximum number of ports allowed on the module, and you attempt to enable additional ports, the additional jumbo frame configurations will fail. You must free up resources by disabling jumbo frames on a port for each additional port you are trying to add before continuing.

Use the **set port jumbo** command in any command mode to enable or disable jumbo frame support on the specified ports.

Use the **set port jumbo mtu** command in any command mode to change the size of the jumbo MTU frame on the port. Jumbo MTU frame must be enabled for the frame size change to take affect.

Use the **set port jumbo mtu dynamic** command to enable or disable dynamic MTU frame on the port.

Use the **show port jumbo** command to verify the operational status of a jumbo enabled port.

The following example enables the port jumbo frame feature, sets the jumbo MTU frame size to 1522 and enables dynamic MTU frame on port ge.1.1 (Q-in-Q must be enabled for the MTU delta and Application name fields to display as presented here):

```
S Chassis(rw)->set port jumbo enable ge.1.1
S Chassis(rw)->set port jumbo mtu 1522 ge.1.1
S Chassis(rw)->set port jumbo mtu dynamic enable ge.1.1
S Chassis(rw)->show port jumbo tg.1.20

* Applicable only if port jumbo is enabled

    Port     Oper     Admin     MTU    MTU   MTU   MTU   MTU   MTU   Application
            Status   Status  Dynamic   Min   Max  Oper* Admin Delta   Name(size)
---------- -------- -------- -------- ----- ----- ----- ----- ----- ------------
        ge.1.1 Enabled  Enabled  Enabled  1000 10239  1522  1522      4  Q-in-Q
S Chassis(rw)->
```

Auto-Negotiation and Port Advertised Ability

Auto-negotiation is an Ethernet feature that facilitates the selection of port speed, duplex, and flow control between the two members of a link, by first sharing these capabilities and then selecting the fastest transmission mode that both ends of the link support. Auto-negotiation is enabled by default.

The advertised ability feature allows for the port to share its port capabilities with the other end of the link. Advertised capabilities will be used during the auto-negotiation process. Actual port capabilities, advertised port capability and remote end advertised port capabilities can be displayed using the **show port advertise** command in any command mode. The following port capabilities can be advertised:

- **10t** - 10BASE-T half duplex mode
- **10tfd** - 10BASE-T full duplex mode

- **100tx** - 100BASE-TX half duplex mode
- **100txfd** - 100BASE-TX full duplex mode
- **1000x** - 1000BASE-X, -LX, -SX, -CX half duplex mode
- **1000xfd** - 1000BASE-X, -LX, -SX, -CX full duplex mode
- **1000t** - 1000BASE-T half duplex mode
- **1000tfd** - 1000BASE-T full duplex mode
- **pause** - PAUSE for full-duplex links
- **apause** - Asymmetric PAUSE for full-duplex links
- **spause** - Symmetric PAUSE for full-duplex links
- **bpause** - Asymmetric and Symmetric PAUSE for full-duplex links

Note: Advertised ability can be activated only on ports that have auto-negotiation enabled.

During auto-negotiation, making use of information gained from the advertised ability feature, the port "tells" the device at the other end of the segment what its capabilities and mode of operation are. If auto-negotiation is disabled, the port reverts to the values specified by the default speed, default duplex, and the port flow control commands.

Use the **set port negotiation** command to enable auto-negotiation on the specified ports.

Use the **set port advertise** command to specify the capabilities to be advertised on the specified ports.

The following example enables auto-negotiation on port ge.1.1 and sets the advertise utility to advertise 10BASE-T half duplex mode, 10BASE-T full duplex mode, 100BASE-TX half duplex mode, 100BASE-TX full duplex mode, and Asymmetric and Symmetric PAUSE for full-duplex links:

```
S Chassis(rw)->set port negotiation ge.1.1 enable
S Chassis(rw)->set port advertise ge.1.1 10t 10tfd 100tx 100txfd bpause
S Chassis(rw)->show port advertise ge.1.1
ge.1.1       capability   advertised    remote
---------------------------------------------
10BASE-T         yes         yes         yes
10BASE-TFD       yes         yes         yes
100BASE-TX       yes         yes         yes
100BASE-TXFD     yes         yes         yes
1000BASE-X       no          no          no
1000BASE-XFD     no          no          no
1000BASE-T       no          no          no
1000BASE-TFD     no          no          no
other            no          no          no
pause            yes         no          yes
Apause           yes         no          no
Spause           yes         no          yes
Bpause           yes         yes         no
S Chassis(rw)->
```

Port MDI/MDIX

The Port MDI/MDIX feature detects and adapts to straight through (MDI) or cross-over (MDIX) Ethernet cabling on switch ports. Ports can be set to auto detect, force MDI or force MDIX. The default is for auto-detection of the cabling type.

Use the **set port mdix** command in any command mode to set the MDI/MDIX feature for the specified ports on this device.

The following example sets the MDI/MDIX feature to cross-over for all ports on this device.

```
S Chassis(rw)->set port mdix mdix
S Chassis(rw)->
```

Port Flow Control

Flow control is the process of managing the rate of data transmission between two nodes to prevent a fast sender from overrunning a slow receiver. It provides a mechanism for the receiver to control the transmission speed. Flow control helps prevent congestion. Flow control should be distinguished from congestion control, which is used for controlling the flow of data when congestion has actually occurred. Flow control on a port is configured for whether the port sends, receives or both sends and receives flow control packets.

When auto-negotiation is enabled the port flow control settings have no bearing on flow control. Pause is negotiated through the predefined advertised settings. The port flow control settings take effect when auto-negotiation is disabled.

Use the **set port flowcontrol** command to both enable flow control and configure the flow control setting for all or the specified ports.

The following example sets flow control on port ge.1.1 to both send and receive flow control packets:

```
S Chassis(rw)->set port flowcontrol ge.1.1 both enable
S Chassis(rw)->
```

Configuring Link Traps and Link Flap Detection

The link traps and link flap detection features provide for the disabling or re-enabling of link traps and to configure the link flapping detection function. By default, all ports are enabled to send SNMP trap messages indicating changes in their link status (up or down).

Use the set port trap command in any command mode to enable the sending of SNMP trap messages when link status changes.

The following example enables SNMP traps on port ge.1.1:

```
S Chassis(rw)->set port trap ge.1.1 enable
S Chassis(rw)->show port trap ge.1.1
Link traps enabled on port ge.1.1.
S Chassis(rw)->
```

The link flap function detects when a link is going up and down rapidly (also called “link flapping”) on a physical port, and takes the configured actions (disable port, and eventually send notification trap) to stop such a condition. If left unresolved, the “link flapping” condition can be detrimental to network stability because it can trigger Spanning Tree and routing table recalculation.

The link flap utility is disabled both globally and on ports by default. The link flap utility must be enabled globally and on the ports for which link flap detection is to occur.

Use the **set linkflap globalstate** command in any command mode to globally enable the link flap utility on this device.

Use the **set linkflap portstate** command in any command mode to enable the link flap utility on the specified ports.

There are three link flap actions that can be configured as a response to link flapping:

- Disable the interface
- Generate a SYSLOG message
- Generate an SNMP trap

You can also set the action to all three. A link flap action will occur if the number of link flaps exceeds the configured link flap threshold (number of times the link flaps) setting within the period configured by link flap interval.

Use the **set linkflap action** command in any command mode to set the link flap action for the specified ports.

Use the **set linkflap threshold** command in any command mode to set the number of link flaps that will trigger a link flap action for the specified ports.

Use the **set linkflap interval** command in any command mode to set the period of time within which the link flap threshold must be exceeded to cause the link flap action to trigger.

If the link flap action is to disable the interface, a port downtime period in seconds can be configured to specify how long the disabled interface will remain down. A value of 0 indicates forever.

Use the **set linkflap downtime** command in any command mode to configure the downtime period for the specified ports.

The following example configures the link flap utility on port `ge.1.1` to:

- Set the link flap action to all three actions
- Set the link flap threshold to 12 link flaps
- Sets the link flap interval to 6 seconds
- Sets the downtime period to 600 seconds

```
S Chassis(rw)->set linkflap action ge.1.1 all
S Chassis(rw)->set linkflap threshold ge.1.1 12
S Chassis(rw)->set linkflap interval ge.1.1 6
S Chassis(rw)->set linkflap downtime ge.1.1 600
S Chassis(rw)->show linkflap parameters ge.1.1
Linkflap Port Settable Parameter Table (X means error occurred)
Port      LF Status  Actions  Threshold   Interval    Downtime
--------  ---------  -------  ----------  ----------  ----------
ge.1.1    disabled   D..S..T  12          6           600
1 port(s) found.
S Chassis(rw)->
```

Port Broadcast Suppression

Broadcast suppression sets a threshold on the broadcast traffic that is received and switched out to other ports. The maximum value in packets per second is 1488100. If the maximum value is configured, broadcast suppression is disabled. Broadcast suppression is disabled by default.

Use the **set port broadcast** command in any command mode to set the broadcast suppression limit, in packets per second, on the specified ports.

The following example sets the broadcast suppression threshold to 10000 packets per second for port ge.1.1:

```
S Chassis(rw)->set port broadcast ge.1.1 10000
S Chassis(rw)->show port broadcast ge.1.1
Port          Total BC       Threshold       Peak Rate   Peak Rate Time
              Packets        (pkts/s)        (pkts/s)    (ddd:hh:mm:ss)
-------------------------------------------------------------------------
ge.1.1        784628         10000           2400         000:00:02:11
```

Port Priority

The Extreme Networks S-Series device supports Class of Service (CoS), which allows you to assign mission-critical data to higher priority through the device by delaying less critical traffic during periods of congestion. The higher priority traffic through the device is serviced first before lower priority traffic. The Class of Service capability of the device is implemented by a priority queueing mechanism. Class of Service is based on the IEEE 802.1D (802.1p) standard specification, and allows you to define eight priorities (0 through 7) and, depending on port type, up to 16 transmit queues (0-15) of traffic for each port.

A priority 0 through 7 can be set on each port, with 0 being the lowest priority. A port receiving a frame without priority information in its tag header is assigned a priority according to the default priority setting on the port. For example, if the priority of a port is set to 4, the frames received through that port without a priority indicated in their tag header are classified as a priority 4 and transmitted according to that priority.

In addition, the device's rate limiting capabilities allow you to further prioritize traffic by limiting the rate of inbound or outbound traffic on a per port/priority basis.

Note: When CoS override is enabled using the **set policy profile** command as described in the "Policy Profile Commands" section of the *Extreme Networks S-Series CLI Reference*, CoS-based classification rules will take precedence over priority settings configured with the **set port priority** command described in this section.

Use the set port priority command in any command mode to set the port priority for the specified ports.

The following example sets the port priority for port ge.1.1 to 4:

```
S Chassis(rw)->set port priority ge.1.1 4
S Chassis(rw)->show port priority ge.1.1
ge.1.1 is set to 4
```

Port Priority to Transmit Queue Mapping

S-Series module ports support up to 16 transmit queues per port depending upon the port type. Use the **show cos port-type txq** command in any command mode to determine the port types and number of transmit queues supported on your module. Packets entering a port are either set for an

802.1p priority value or take on the default priority value for this port. The behavior of a packet as it exits the port is dependent upon the priority value assigned to the packet and the transmit queue it exits the port on.

802.1p priority values can be mapped directly to transmit queues on a per port basis. Regardless of the 802.1p priority mapped to a queue, the queue itself has a priority from low to high where queue 0 has the lowest priority and the highest queue value has the highest priority. For example, in a strict queuing configuration, the highest queue number would empty first before moving on to the next highest queue number. See "Preferential Queue Treatment for Packet Forwarding" on page 53-5 for a detailed discussion of preferential queue treatment.

Use the **set port priority-queue** command to map 802.1p priorities to transmit queues on a per port basis.

The following example sets priority 5 packets to transmit queue 1 on port `ge.1.1`

```
S Chassis(rw)->set port priority-queue ge.1.1 5 1
S Chassis(rw)->show port priority-queue ge.1.1
Port           P0 P1 P2 P3 P4 P5 P6 P7
------------   -- -- -- -- -- -- -- --
ge.1.1         1  0  0  1  2  1  3  3
S Chassis(rw)->
```

Energy Efficient Ethernet (EEE)

Enabling EEE on a link reduces the power consumption on the Ethernet link during low data activity. EEE must be enabled on both sides of the link to operate. Auto negotiation is restarted when EEE is enabled or disabled, causing the link to bounce. Link state does not change as a result of an EEE transition to and from a lower level of power. Frames that are in transit are neither dropped nor corrupted during EEE transition to and from a lower level of power.

The wakeup time is the period between the reception of an IDLE signal and the reception of the first data permitted on the interface. It is recommended that you only modify wakeup and fallback values if a longer wakeup time is required. The negotiation of wakeup times ia accomplished using the LLDP EEE TLV which must be enabled on both sides of the link using the set lldp port tx-tlv energy-eff-eth command. If the configured wakeup time is not acceptable, a fallback wakeup time is used.

Use the **set port energy-eff-eth** command to enable EEE and change wakeup and fallback times.

Configuring Ports

This section provides details for the configuration of ports on the S-Series products.

Table 6-1 lists port parameters and their default values.

Table 6-1 Default Port Parameters

Parameter	Description	Default Value
broadcast suppression	Specifies a limit for the number of broadcast packets per second that can be received and switched on a port.	disabled (set to max value of 1488100)
console baud rate	Specifies the baud rate for the console port.	9600
console bits	Specifies the number of bits per character on the console port.	8 bits

Table 6-1 Default Port Parameters (continued)

Parameter	Description	Default Value
console flow control	Specifies the flow control mechanism for the console port.	ctsrts (Clear to Send/Request to Send)
global link flap state	Specifies whether link flap is enabled globally on this device	disabled
jumbo frame support	Specifies whether Ethernet frame with a payload greater than 1500 is supported on this port.	disabled
port ingress filter	Specifies that frame forwarding is limited to members of the port's VLAN egress list.	disabled
port negotiation	Specifies whether auto-negotiation is enabled on this port.	enabled
port priority	Specifies the 802.1D priority for this port.	0
port state	Specifies the port state.	disabled
port traps	Specifies whether the sending of port traps is enabled on this port.	enabled

Procedure 6-1 describes how to configure ports.

Procedure 6-1 Configuring Ports

Step	Task	Command(s)
1.	Administratively enable one or more ports on the system.	**set port enable** *port-string*
2.	Optionally, change the properties for one or more console ports.	**set console** {[**baud** *rate*] | [**bits** *num-bits*] | [**cts-link** {**enable** | **disable**}] | [**flowcontrol** {**none** | **ctsrts** | **dsrdtr**}] | [**parity** {**none** | **odd** | **even** | **mark** | **space**}] | [**stopbits** {**one** | **oneandhalf** | **two**}] [**vt100 dsr** {**enable** | **disable** | **timeout** *timeout*]} [*port-string*]
3.	Optionally, limit the forwarding of received frames based on port VLAN egress lists.	**set port ingress-filter** *port-string* **enable**
4.	Optionally, assign an alias name to a port.	**set port alias** *port-string* [*string*]
5.	Optionally, enable the forcing of ports in the "operstatus down" state to become disabled.	**set forcelinkdown enable**
6.	Optionally, set the default speed of one or more ports.	**set port speed** *port-string* {**10** | **100** | **1000** | **10000** | **40000**}
7.	Optionally, set the default duplex type for one or more ports.	**set port duplex** *port-string* {**full** | **half**}
8.	Optionally, enable jumbo frame support on one or more ports.	**set port jumbo enable** [*port-string*]
9.	Optionally, enable auto-negotiation on one or more ports.	**set port negotiation** *port-string* **enable**
10.	Optionally, set MDI/MDIX mode on one or more ports.	**set port mdix** [*port-string*] {**auto** | **mdi** | **mdix**}

Procedure 6-1 Configuring Ports (continued)

Step	Task	Command(s)
11.	Optionally, configure the auto-negotiation advertised capabilities on one or more ports.	**set port advertise** *port-string* {[**10t**] [**10tfd**] [**100tx**] [**100txfd**] [**1000x**] [**1000xfd**] [**1000t**] [**1000tfd**] [**pause**] [**apause**] [**spause**] [**bpause**])
12.	Optionally, enable flow control settings for one or more ports.	**set port flowcontrol** *port-string* {**receive** | **send** | **both**} **enable**
13.	Optionally, set the broadcast suppression limit on one or more ports.	**set port broadcast** *port-string threshold-val*
14.	Optionally, set a default port priority for one or more ports.	**set port priority** *port-string priority*
15.	Optionally, map 802.1D (802.1p) priorities to transmit queues for one or more ports.	**set port priority-queue** *port-string priority queue*

Procedure 6-2 describes how to configure link trap and link flap detection.

Procedure 6-2 Configuring Link Trap and Link Flap Detection

Step	Task	Command(s)
1.	Optionally, enable one or more ports for sending SNMP trap messages when link status changes occur.	**set port trap** *port-string* **enable**
2.	Optionally, globally enable the link flap detection function for this device. Defaults to disabled.	**set linkflap globalstate enable**
3.	Optionally, enable the link flap detection function on one or more ports. Defaults to disabled.	**set linkflap portstate enable** [*port-string*]
4.	Optionally, change the period of time within which the link flap threshold must be exceeded to cause the link flap action to trigger.	**set linkflap interval** *port-string interval_value*
5.	Optionally, set the action that will occur when a link flap violation threshold is met.	**set linkflap action** *port-string* {**disableInterface** | **gensyslogentry** | **gentrap** | **all**}
6.	Optionally, change the link flap action trigger threshold.	**set linkflap threshold** *port-string threshold_value*
7.	Optionally, set the length of time one or more ports will be held down after a link flap violation threshold is met and the action is set to disable the interface.	**set linkflap downtime** *port-string downtime_value*

Table 6-2 describes how to manage port configuration.

Table 6-2 Managing Port Configuration

Task	Command
To clear the properties set for one or more console ports to its default values:	**clear console** [**baud**] [**bits**] [**cts-link**] [**flowcontrol**] [**parity**] [**stopbits**] [**vt100**] [*port-string*]
To override the causes configured to place operating status to a down or dormant state for one or more ports:	**clear port operstatuscause** [*port-string*] [**admin**] [**all**] [**cos**] [**flowlimit**] [**linkflap**] [**policy**]

Table 6-2 Managing Port Configuration (continued)

Task	Command
To reset the force link down function to the default state of disabled:	**clear forcelinkdown**
To reset jumbo frame support status to enabled on one or more ports:	**clear port jumbo** [*port-string*]
To reset MDIX mode to the default setting of auto on one or more ports:	**clear port mdix** [*port-string*]
To reset auto-negotiation advertised capabilities to the default setting on one or more ports:	**clear port advertise** *port-string* [**10t** | **10tfd** | **100tx** | **100txfd** | **1000x** | **1000txfd** | **1000t** | **1000tfd** | **pause** | **apause** | **spause** | **bpause**]
To clear the configured actions to a link flap violation:	**clear linkflap action** {*port-string*} {**disableInterface** | **gensyslogentry** | **gentrap** | **all**}
To toggle link flap disabled ports to operational:	**clear linkflap down** [*port-string*]
To clear all link flap options or statistics on one or more ports:	**clear linkflap** {**all** | **stats** [*port-string*] | **parameter** *port-string* {**threshold** | **interval** | **downtime** | **all**}
To reset the broadcast threshold or clear the peak rate and peak time values on one or more ports:	**clear port broadcast** *port-string* {[**threshold**] [**peak**]}
To reset the current default port priority setting to the default value of 0 on one or more ports:	**clear port priority** *port-string*
To reset port priority queue settings back to defaults for one or more ports.	**clear port priority-queue** *port-string*

Table 6-3 describes how to display port configuration information and statistics.

Table 6-3 Displaying Port Configuration Information and Statistics

Task	Command
To display properties set for one or more console ports:	**show console** [**baud**] [**bits**] [**flowcontrol**] [**parity**] [**stopbits**] [*port-string*]
To display whether or not one or more ports are enabled for switching:	**show port** [*port-string*]
To display operating and admin status, speed, duplex mode and port type for one or more ports on the device:	**show port status** [*port-string*] [**-interesting**]
To display port counter statistics detailing traffic through the device and through all MIB2 network devices:	**show port counters** [*port-string*] [**switch** | **mib2** | **brief** | **packets** | **detail** | **errors**] [**nonzero**]
To display the causes configured to place operating status to a down or dormant state for one or more ports:	**show port operstatuscause** [**admin** | **any** | **cos** | **dot1x** | **flowlimit** | **init** | **lag** | **linkflap** | **linkloss** | **modifiable** | **policy** | **self**] [*port-string*]
To display all ingress-filter enabled ports or the ingress-filter state of the specified ports:	**show port ingress-filter** *port-string*
To display alias name(s) assigned to one or more ports:	**show port alias** [*port-string*]
To display the status of the force link down function:	**show forcelinkdown**

Table 6-3 Displaying Port Configuration Information and Statistics (continued)

Task	Command
To display port transceiver information:	**show port transceiver** [*port-string*] [**basic-only**] [**sensor-only**] [**all**]
To display the default speed setting on one or more ports:	**show port speed** [*port-string*]
To display the default duplex setting for one or more ports:	**show port duplex** [*port-string*]
To display the status of jumbo frame support and MTUs on one or more ports:	**show port jumbo** [*port-string*]
To display the status of auto-negotiation for one or more ports:	**show port negotiation** [*port-string*]
To display MDIX mode on one or more ports:	**show port mdix** [*port-string*] {**all** | **auto** | **mdi** | **mdix**}
To display the advertised abilities on one or more ports:	**show port advertise** [*port-string*]
To display the flow control state for one or more ports:	**show port flowcontrol** [*port-string*]
To display the default 802.1D priority for one or more ports:	**show port priority** [*port-string*]
To display port broadcast suppression information on one or more ports:	**show port broadcast** [*port-string*]

Terms and Definitions

Table 6-4 lists terms and definitions used in this port configuration discussion.

Table 6-4 Port Configuration Terms and Definitions

Term	Definition
auto-negotiation	An Ethernet feature that facilitates the selection of port speed, duplex, and flow control between the link segments by first advertising these capabilities and then selecting the fastest transmission mode common to both segments.
baud rate	The speed the console port operates at.
broadcast suppression	A port feature that sets a threshold on the broadcast traffic that is received and switched out to other ports.
console port	A port through which local management of the device can be accessed using a terminal or modem.
default priority	A default 802.1p priority that will be applied to a packet when no priority is set in the packet as it transits the port.
duplex	The specification of whether the communications between two devices is one way at a time or both ways simultaneously.
flow control	A port feature that manages the rate of data transmission between two nodes to prevent a fast sender from overrunning a slow receiver.
force linkdown	A port feature that allows for the forcing of a port in the “operstatus down” state be become disabled.
ingress filtering	A port feature that provides a means of limiting the forwarding of received frames on the ingress port based on the VLAN egress list for that port.
jumbo frame	A port feature that supports Ethernet frames greater than 1500 bytes of payload on the port.

Table 6-4 Port Configuration Terms and Definitions (continued)

Term	Definition
link flap detection	A port feature that detects when a link is rapidly going up and down and provides for a port behavior when a threshold is crossed during a configured interval.
MDI/MDIX	A port feature that detects and adapts to straight through (MDI) or cross-over (MDIX) Ethernet cabling on the switch ports.
port advertised ability	The aspect of auto-negotiation that allows a port to share its capabilities with the other end of the link.
port alias	The association of a string name with a port.
port string	A port identifier made up of port type, chassis slot the module containing the port is installed into, and the port number, delineated by a period (**.**).

7

Ethernet Operations, Administration, and Maintenance (OAM) Configuration

This document provides the following information about configuring Ethernet Operations, Administration, and Maintenance (OAM) on the S-Series platforms.

Using Ethernet OAM in Your Network

Ethernet Operations, Administration, and Maintenance (OAM) is a collection of standards provided by multiple standards bodies to enable network operators a means to effectively monitor and troubleshoot individual Ethernet links. The Extreme Networks modular switch OAM implementation supports the IEEE 802.3-2008 Clause 57 standard.

The IEEE 802.3-2008 Clause 57 standard allows network operators to monitor and exercise an individual Ethernet link. It provides a set of diagnostics and monitoring functions at a data link level, allowing operators to make a determination of a link's relative health and operational status, and to take administrative action against degraded or faulty links.

A network operator may use SNMP to periodically poll devices for statistics in an attempt to determine when faults occur. However, when a fault occurs, the network operator does not have any means of detecting which links are impacted, and which customers are affected. By implementing OAM across the network, the network operator can proactively determine link degradation or failure, and indicate which customer services are down.

An operator may choose to define an administrative action that will take effect when an error condition occurs or a specified error threshold is crossed. This action may include the generation of Syslog events to bring the state of the suspect link to the operator's attention or to operationally disable the link to avoid network service interruptions caused by degraded or faulty links.

Once corrective action has taken place, link monitoring and remote loopback can be used to verify that the remedial action has succeeded.

The OAM implementation includes a Unidirectional Link Detection (ULD) feature capable of determining when an otherwise bidirectional link is only operational in one direction. This

problem is often caused by faulty wiring or a hardware failure. ULD is not explicitly defined in the OAM standard. The OAM standard provides the means for a ULD solution through a combination of its discovery protocol, administrative actions, and organization specific information LLDP Type-Length-Value (TLV)s. ULD provides for disabling the offending port and sending a Syslog message or only sending a Syslog message.

Implementing Ethernet OAM

To implement Ethernet OAM on your network:

1. Set the port OAM status to enabled for ports to be monitored.
2. Optionally, set the port OAM mode to passive for any ports that should not initiate OAM contact with their neighbors or have the ability to put the neighbor in remote loopback.
3. Optionally, modify the default OAM link monitor threshold configuration and actions to be taken when link monitor thresholds are crossed, for each OAM enabled port.
4. Optionally, configure an OAM enabled port to process remote loopback requests from its peer.
5. Optionally, configure the number of event notification OAM PDUs retransmitted by the remote peer.
6. If a remote port requires troubleshooting, optionally, set the remote port in remote loopback mode to aid in diagnosing the problem.
7. If you need to bring up an interface that has been taken down due to exceeding OAM monitoring thresholds, clear the OAM operstatus cause to bring the interface back up.
8. Optionally, configure ports for ULD.

Ethernet OAM Overview

OAM Client

The OAM client contains the essential control operations and state information concerning OAM operations on a specific port. It is responsible for the handling of received OAM PDUs from remote clients, and based upon the state of local and remote settings, allows OAM to operate upon a link.

Link events are transmitted via OAM PDUs between OAM client entities. The OAM Client is also responsible for maintaining statistics concerning transmitted and received OAM PDUs.

You must enable the OAM client on the device for OAM operations to take place. The OAM client is disabled by default.

Use the **set port oam status** command to enable the OAM client on the device.

OAM Discovery

Periodic Information OAM PDU messages are exchanged between OAM clients to both initiate OAM discovery on the link and, once initiated, assure that remote client information is correct from the perspective of the local client. Information OAM PDUs can contain the remote OAM client's information, as well as a copy of the local client's information. Both clients must accept the exchanged information to complete OAM discovery. Clients can reject received information that is incorrect, outdated, or incomplete.

OAM clients must accept two sets of information to complete OAM discovery:

- The remote client's information
- A copy of the local client's information that has been reproduced by the remote client

Once discovery is completed, any change to configuration or state information, on either OAM client, forces the OAM discovery session to be torn down and re-established with the new information.

OAM Client Mode

OAM clients may operate in either active or passive mode. Clients configured for active mode may initiate contact with remote peers. Once the discovery process has completed with the remote peer, active clients are allowed to send remote loopback control OAM PDUs to that peer. Clients configured for active mode should ignore requests from a passive remote peer to enter remote loopback mode.

OAM clients configured for passive mode may not initiate any contact with a remote peer. Passive OAM clients are only allowed to respond to requests received from a remote peer. When OAM is disabled on the port, all OAM frames are discarded.

You may want to require that a remote peer outside of your administrative control be set to OAM passive mode to disable that port's ability to initiate OAM discovery and processing, as well as to prevent the remote port from setting the local port in remote loopback. If you need to keep the remote port in active mode, you can also protect against the local port being placed in remote loopback by configuring the local port loopback receive behavior to ignore remote loopback requests, using the **set port oam loopback-rx** command. See "OAM Client Remote Loopback Request Behavior" on page 7-9 for details about setting the port loopback receive behavior.

OAM client mode is configured using the **set port oam mode** command.

OAM Datalink Layer Monitoring

OAM includes datalink layer monitoring, which can be used to proactively take action upon links which are exhibiting faulty behavior.

The OAM link monitor checks for symbol and frame errors.

- A symbol is a fixed length electrical or optical waveform on the wire that represents a binary value and is a subcomponent of a frame. A symbol error is a symbol with an invalid data value.
- A frame is made up of a variable number of symbols formatted with a preamble, header, data blocks, and checksum. A frame error can have a number of causes such as MTU exceeded, invalid frame check sequence, an alignment error (length is not an integer number of octets) or length error.

Network administrators may define threshold values for symbol or frame errors. If the threshold is exceeded within a configured link monitor window, a threshold crossing event has transpired and an action will be taken. Supported link monitor actions include:

- Transmitting a Link Event Notification message to the remote OAM client
- Generating a Syslog message
- Taking the link operationally off-line

The administrator has the option of keeping a suspect link operationally disabled until manual intervention has been taken or for the link to be operationally re-enabled once the error condition has resolved.

Configure OAM datalink layer monitoring using the **set port oam link-monitor** command.

Configuring OAM datalink layer monitoring includes setting several options, described in the following sections.

Frame Option

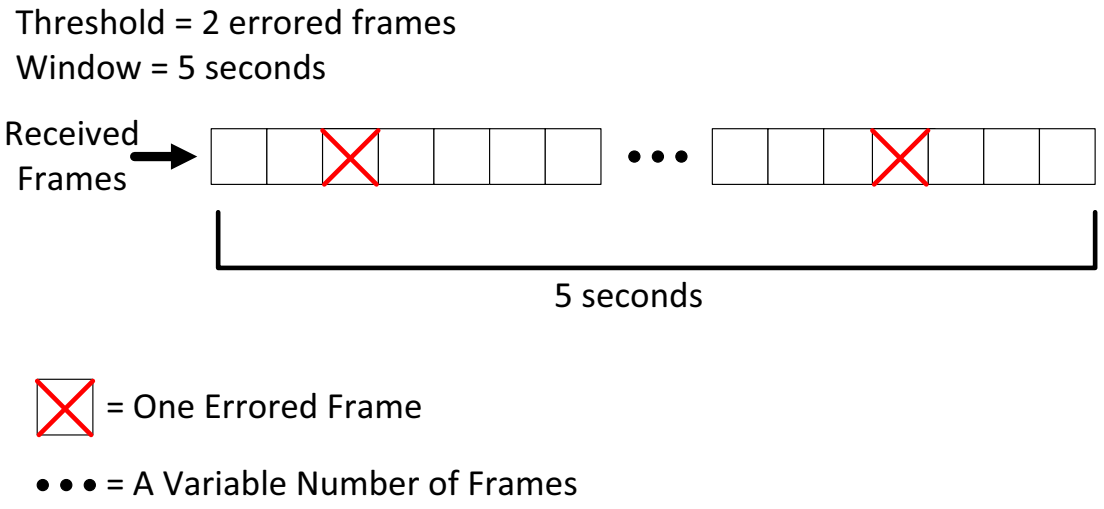
The **frame** option monitors frame errors occurring during a period of time. The default threshold for the frame option is **1** errored-frame. The default window for the frame option is **1** second, and the maximum window is **60** seconds.

As presented in Figure 7-1:

- The configured threshold is **2** errored-frames that occur within the configured window.
- The window within which the threshold must be exceeded for a configured action to occur is **5** seconds.

If more than **2** errored-frames are received on the monitored port within any **5** second window, the configured action occurs.

Figure 7-1 Frame Link Monitor Option



Frame-Seconds Option

The **frame-seconds** option monitors frames within one or more one-second windows for errors. The link monitor window is specified as a number of one-second windows to monitor. The default threshold is **1** errored-second window. The default window is **60** one-second windows. The minimum window is **10** one-second windows, and the maximum window is **900** one-second windows.

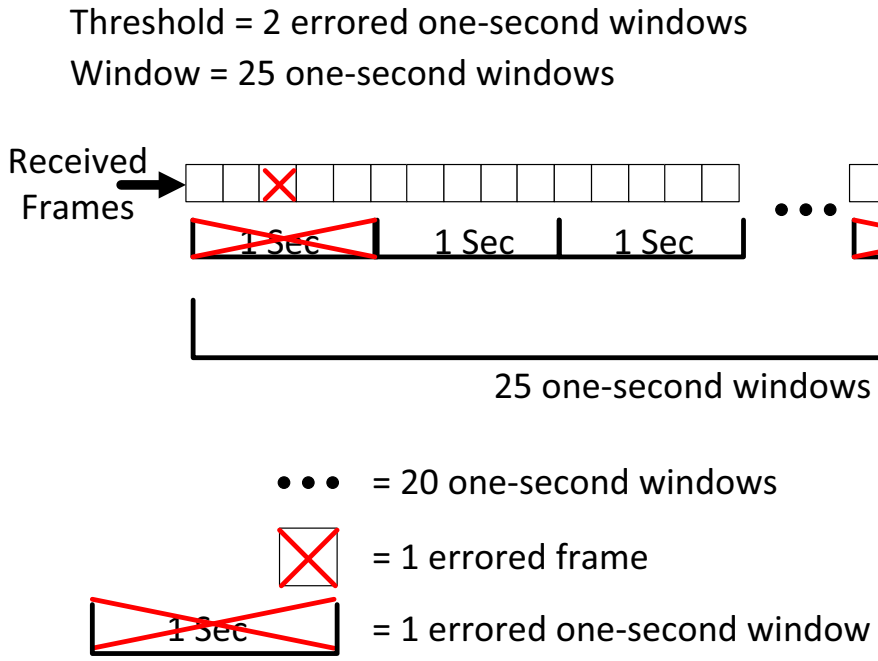
For example, if defaults are being used and a single frame error occurs within one of sixty one-second windows, the threshold has been met. If the threshold was raised to three errored-seconds, three one-second windows would have to register a frame error, within sixty one-second windows, for the threshold to be met.

As presented in Figure 7-2:

- The configured threshold is **2** one-second windows in which one or more errored-frames occur.
- The window is **25** one-second windows to monitor.

If there is one or more frame errors in **3** or more one-second windows within the link monitor window of **25** one-second windows, the configured action occurs.

Figure 7-2 Frame-Seconds Link Monitor Option



Frame-Period Option

The **frame-period** option monitors frame errors that occur during the reception of a given number of frames. The default threshold is **1** errored-frame. The default window is equivalent to the maximum number of minimum sized frames that may be transmitted over the link during a **1** second interval, and the upper bound is the maximum number of minimum sized frames that may be transmitted over the link during a **1** minute interval.

The frame-period option defines its window values based upon the line rate of the port being configured. As such, option values may not be determined until the port has achieved a valid link state. The frame-period window default and range values, based upon link speed, are displayed in Table 7-1.

Table 7-1 Frame-Period Window Values

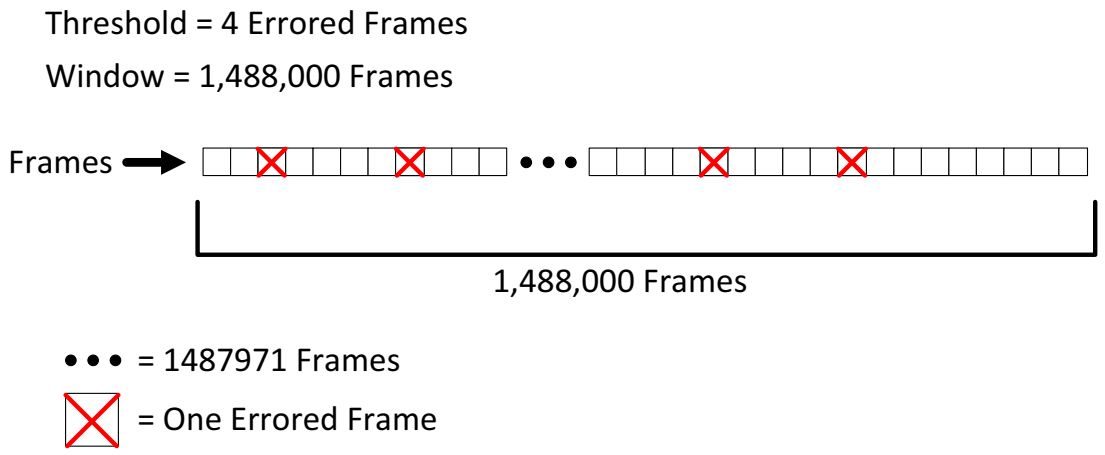
Port Line Rate	Default Window Value	Window Value Range
100 Mbps	148,800 frames	14880 - 8928000 frames
1Gbps	1,488,000 frames	148,800 to 89,280,000 frames
10 Gbps	14,880,000 frames	1,488,000 to 892,800,000 frames

As presented in Figure 7-3:

- The configured threshold is **4** errored-frames that occur within the configured window.
- The window is the default window for a 1Gbps port: **1,488,000** frames.

If there is more than **4** errored-frames within the reception of **1,488,000** frames, the configured action occurs.

Figure 7-3 Frame-Period Link Monitor Option



Symbol-Period Option

The **symbol-period** option monitors symbol errors that occur during the reception of a given number of symbols. The default threshold is **1** errored-symbol. The default window is equivalent to the maximum number of symbols that may be transmitted over the link during a one second interval, and the upper bound is the number of symbols that may be transmitted over the link during a one minute interval.

The symbol-period option defines its default window value based upon the line rate of the port being configured. As such, the default window value may not be determined until the port has achieved a valid link state. The symbol-period window default and range values, based upon link speed, are displayed in Table 7-2 on page 7-6.

Table 7-2 Symbol-Period Window Values

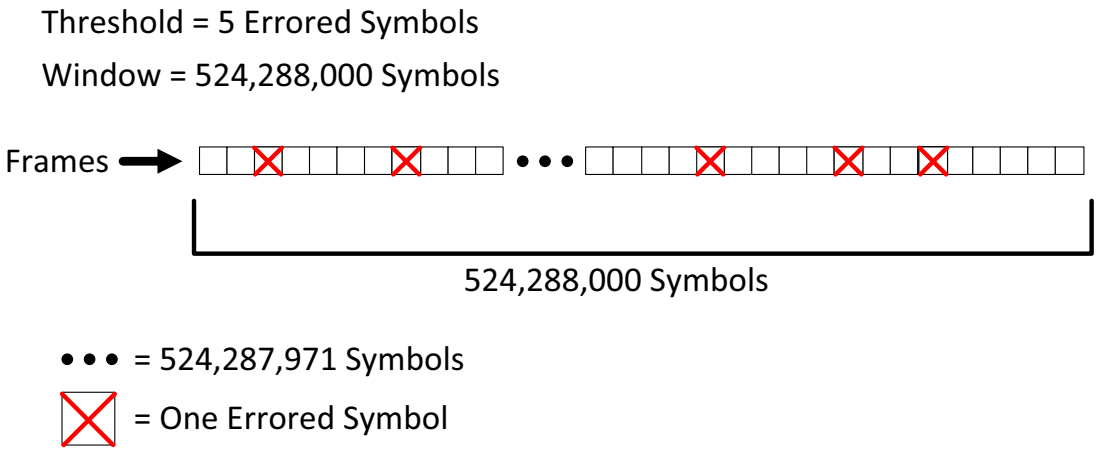
Port Line Rate	Default Window Value	Window Value Range
100 Mbps	131,072,000 symbols	52,428,800 - 7,864,320,000
1Gbps	524,288,000 symbols	524,288,000 - 31,457,280,000
10 Gbps	5,242,880,000 symbols	5,242,880,000 - 314,572,800,000

As presented in Figure 7-4:

- The configured threshold is **5** errored-symbols.
- The window is the default window for a 1Gbps port: **524,288,000** symbols.

If there is more than **5** errored-symbols within the reception of **524,288,000** symbols, the configured action occurs.

Figure 7-4 Symbol-Period Link Monitor Option



Actions

The administrator may configure one or more of three actions to be taken upon the detection of a link event:

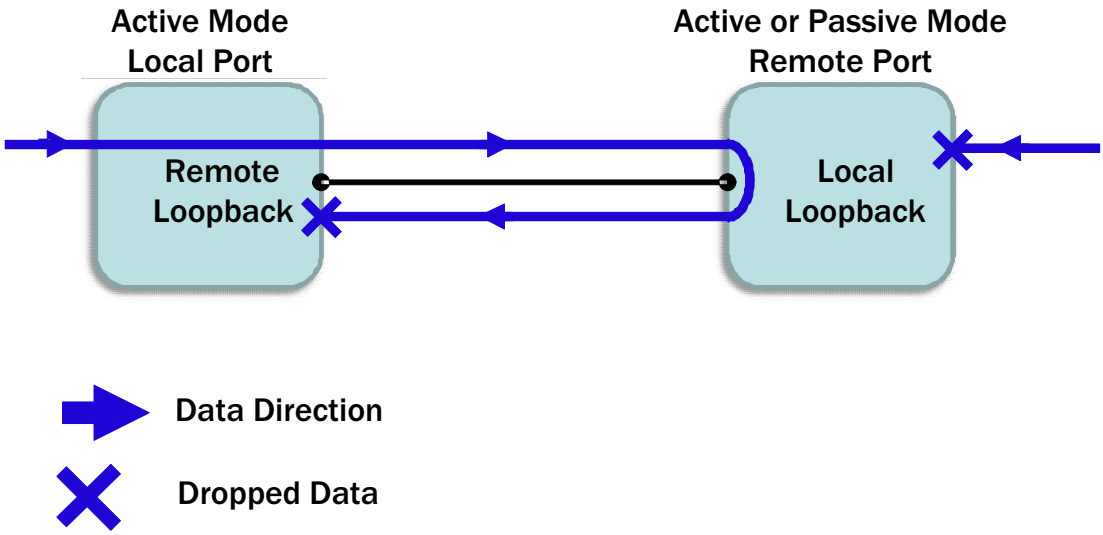
- The **syslog** option triggers a syslog message to be generated, which confers information related to the event.
- The **notify** option triggers the transmission of a link event notification OAM PDU to the remote client.
- The **disable-interface** option operationally disables the port in question, and the port remains in that state until the administrator restores the port to operational status.

OAM Remote Loopback Mode

OAM provides an optional datalink layer loopback mode, which can be used for fault detection and performance analysis. An active mode OAM client may command its remote peer to go into loopback mode. When the link examination has been completed by the network operator, the OAM client on the remote port can be taken out of loopback mode resulting in the restoration of its previous operational mode. A client which has been put into loopback mode will retransmit all traffic that has been received on that port (with the exception of OAM PDUs) back towards the sender.

Figure 7-5 on page 7-8 displays an operational OAM remote loopback configuration. Data egressing the local port is reflected back to the local port by the remote port. All reflected data is dropped by the local port. All data ingressing the non-loopback side of the remote port is dropped. Use a combination of RMON and port counters to determine port performance.

Figure 7-5 Remote Loopback



Note: OAM remote loopback is a disruptive test state intended to aid in the diagnosis of network issues. OAM remote loopback interferes with the normal operation of other network protocols and data flows over that link.

Use the **set port oam remote-loopback** command to enable the OAM remote loopback mode on the local OAM client.

This command:

- Instructs the remote OAM client to initiate, if enabled, or terminate, if disabled, the remote loopback process.
- Is a volatile configuration option that does not persist across reboots, and is not displayed in the **show config port** output.

Port OAM must be configured for active mode, the OAM client must have completed the discovery process with the remote OAM client, and that client must indicate that it supports loopback in order to initiate the loopback process. A client which has been put into loopback mode will re-transmit all traffic that has been received on that port (with the exception of OAM PDUs) back towards the sender.

A client which has put its peer into loopback mode will discard all received traffic (with the exception of OAM PDUs) on that port. Be aware that OAM remote loopback is a disruptive test state intended to aid in the diagnosis of network issues, and will interfere with the normal operation of other network protocols and data flows over that link.

Caution must be used when placing an OAM enabled port in remote loopback. When requesting the remote OAM enabled port to be placed in loopback, the loopback mode of the remote port must be set to **process** using "set port oam loopback-rx" on page 20-28. OAM remote loopback's behavior is "fire and forget". If for any reason, including OAM mode set to **ignore** or the loopback request or remote response should be lost in transit, remote loopback will remain in either an initiating or terminating loopback state. The potential for harm to the network exists where one end of the link believes remote loopback is in effect, and the other does not. The port will be prevented from both sending and receiving data until OAM is administratively disabled and then enabled on both ends of the link.

The default value for OAM remote loopback is **disable**.

OAM Client Remote Loopback Request Behavior

The behavior of the OAM client when receiving a remote loopback request from a remote OAM client can be set to either **process** or **ignore** the remote loopback request. Setting this value to process will allow the OAM client to receive and operate upon a remote loopback OAM PDU request. Setting this value to ignore will force the OAM client to discard any received remote loopback OAM PDUs. The default value is **ignore**.

Use the **set port oam loopback-rx** command to configure the OAM local client remote loopback request behavior.

OAM Event Notification Retries

The local OAM client can be configured to retransmit event notification OAM PDUs to its remote peer up to the configured number of retries.

If an OAM monitor link threshold is crossed, generating an event, and the notification action is configured, a single event notification OAM PDU is always generated. By default no further notifications are sent unless a value greater than zero is configured for the number of OAM event retries. This retransmission process will halt if the link monitoring process determines that additional events have transpired upon the link. A notification of the new event is then sent by the remote OAM client. The default value is **0**, and the maximum value is **10**.

Use the **set port oam notify-retry** command to configure the number of event notification retries sent by the remote OAM client when an OAM event occurs.

Unidirectional Link Detection (ULD)

When a link is in a unidirectional state, the ability to pass traffic over the link is broken in one direction. This causes problems for protocols that depend upon reliable bidirectional communications to operate. For example, the Spanning Tree Protocol (STP) relies on the bidirectional exchange of bridge protocol data units to create a loop free topology. By implementing link layer OAM with Unidirectional Link Detection (ULD) across the network, OAM ULD can be configured to take a port out of service when a unidirectional link is detected. In the case of a unidirectional link in a STP context, the creation of a loop within the network is prevented.

ULD Mode

By default ULD mode is disabled, and the detection of unidirectional links does not occur. Extreme Networks supports two means of detecting a unidirectional link on a port using ULD modes:

- **Standard** – Uses the existing OAM discovery protocol to perform the configured ULD action, if more than 5 seconds elapses between reception of standard information OAMPDUs on the port.
- **Fast** – Establishes a second tier of OAM discovery by transmitting information OAMPDUs with the Fast ULD information TLV. The configured ULD action is performed when up to 3 times the interval defined by the fast timer setting (a configurable range of between 600 milliseconds and 3 seconds) elapses between reception of a Fast ULD information TLV on the port.

You can set the ULD mode on a per port basis using the **set port oam uld mode** command.

ULD Fast Timer

The ULD fast timer determines the interval between transmission of Fast ULD information TLVs used by the ULD fast mode to detect unidirectional links.

When setting the fast timer you specify a fast timer multiplier between **2** and **10**. ULD sets the fast timer value by multiplying the specified fast timer multiplier by 100 milliseconds. The fast timer interval defaults to 200 milliseconds (2 x 100 milliseconds). When using the default timer interval of 200 milliseconds, up to 600 milliseconds (3 times the configured timer interval) will elapse between the reception of the Fast ULD information TLV and ULD performing the configured ULD action.

You configure the fast timer setting by specifying the fast timer multiplier using the **set port oam uld fast-timer** command.

ULD Action

When ULD detects a unidirectional link in either standard or fast mode, a Syslog message is sent or a Syslog message is sent and the port is disabled. The default action is sending a Syslog message without disabling the port.

You configure the ULD action on a port when a unidirectional link is detected using the **set port oam uld action** command.

Configuring Ethernet OAM

This section provides:

- A listing of OAM default values
- An OAM configuration procedure

Table 7-3 lists the S-Series device default Ethernet OAM configuration settings.

Table 7-3 Default Ethernet OAM Configuration Settings

Parameter	Description	Default Value
OAM client mode	Specifies the operating mode of the local OAM client for the port.	active
OAM event notification retries	Sets the number of event notification retries to send for the port.	0
OAM loopback request receive behavior	Determines the behavior of the OAM client when receiving a remote loopback request from a remote OAM client.	ignore
OAM remote loopback	Instructs the remote OAM client to initiate, if enabled, or terminate, if disabled, the remote loopback process.	disabled
OAM status	Specifies the state of the local port OAM client.	disabled
OAM monitor link frame window	Specifies the period for single errored-frame threshold.	1 second

Table 7-3 Default Ethernet OAM Configuration Settings (continued)

Parameter	Description	Default Value
OAM monitor link frame-period frame threshold	Specifies the number of errors received based upon the number of frames per second by port linked rate.	100 Mbps – 1 error per 148,800 frames per second 1Gbps – 1 error per 1,488,000 frames per second 10 Gbps – 1 error per 14,880,000 frames per second
OAM monitor link frame-seconds threshold	Specifies the number of one-second intervals in which one or more frame errors occurs.	1 errored-second
OAM monitor link symbol-period threshold	Specifies the number of symbol errors that occur during the reception of a given number of symbols based upon the port line rate.	100 Mbps – 1 error per 131,072,000 symbols per second 1Gbps – 1 error per 524,288,000 symbols per second 10 Gbps – 1 error per 5,242,880,000 symbols per second
OAM monitor link action	Specifies the action that occurs should a link monitor event occur.	notify
OAM ULD mode	Specifies whether ULD is using standard or fast mode to detect unidirectional links or whether ULD is disabled.	disabled
OAM ULD action	Specifies whether ULD will only send a Syslog message or both disable the port and send a Syslog message when a unidirectional link is detected.	send Syslog message only
OAM ULD fast timer multiplier	Specifies an integer value, multiplied by 100 milliseconds, to determine the fast timer interval used by ULD fast mode for sending OAMPDUs with Fast ULD information TLVs.	2

Procedure 7-1 provides an example of an OAM configuration.

Procedure 7-1 Configuring OAM

Step	Task	Command(s)
1.	Administratively enable Ethernet OAM on the specified port(s).	**set port oam** *port-string* **status enable**
2.	Optionally, set the operating mode for the OAM client on the specified port(s).	**set port oam** *port-string* **mode** {**active** | **passive**}
3.	Optionally, configure OAM link monitor functionality for the specified port.	**set port oam** *port-string* **link-monitor** {**frame** | **frame-period** | **frame-seconds** | **symbol-period**} {**threshold** *threshold* | **window** *window* | **action** {[**syslog**] [**disable-interface**] [**notify**]}
4.	Optionally, set the OAM loopback request behavior for the specified port(s).	**set port oam** *port-string* **loopback-rx** {**ignore** | **process**}

Procedure 7-1 Configuring OAM

Step	Task	Command(s)
5.	Optionally, set the number of notify retries to send for the specified port.	**set port oam** *port-string* **notify-retry** *retries*
6.	Optionally, enable OAM remote loopback for the specified port(s).	**set port oam** *port-string* **remote-loopback enable**
7.	Optionally, bring up an interface that has been taken down due to exceeding OAM monitoring thresholds.	**clear port operstatuscause** [*port-string*] [**oam**] [**oamlb**]
8.	Optionally, enable ULD standard or fast mode to detect unidirectional links on a port.	**set port oam** *port-string* **uld mode** {**standard** | **fast**}
9.	Optionally, set the ULD action to both disable the unidirectional port and send a Syslog message or only send a Syslog message.	**set port oam** *port-string* **uld action** {**disable-port** | **syslog-only**}
10.	Optionally, change the ULD fast timer multiplier for a port.	**set port oam** *port-string* **uld fast-timer** *multiplier*

Ethernet OAM Configuration Example

The following CLI example provides a local and remote port OAM configuration that:

- Enables port OAM status on both the local (ge.3.14) and remote (ge.1.10) ports.
- Retains the default active OAM mode on the local port and sets the remote port OAM mode to passive. This configuration allows the local port to both initiate OAM on the link and place the remote port into remote loopback for troubleshooting and diagnostic purposes. This configuration prevents the remote port from initiating any OAM operations.
- Sets the remote loopback receive behavior on the remote port to process a remote loopback request. The local port retains the default setting of ignore.
- Sets the number of notify retries sent by the remote port to 3. The local port retains the default value of 0 retries.
- Sets the local and remote port's OAM monitor link frame window to 100 seconds, the threshold to 2 errors and the action to Syslog, disable the interface, and notify, should the frame event threshold be reached.

Switch 1, Local Port: ge.3.14

```
S Chassis(rw)->set port oam ge.3.14 status enable
S Chassis(rw)->set port oam ge.3.14 notify-retry 3
S Chassis(rw)->set port oam ge.3.14 link-monitor frame threshold 2
S Chassis(rw)->set port oam ge.3.14 link-monitor frame window 100
S Chassis(rw)->set port oam ge.3.14 link-monitor frame action syslog
disable-interface notify
```

Switch 2, Remote Port ge.1.10

```
S Chassis(rw)->set port oam ge.1.10 status enable
S Chassis(rw)->set port oam ge.1.10 mode passive
S Chassis(rw)->set port oam ge.1.10 loopback-rx process
S Chassis(rw)->set port oam ge.1.10 link-monitor frame threshold 2
S Chassis(rw)->set port oam ge.1.10 link-monitor frame window 100
```

```
S Chassis(rw)->set port oam ge.1.10 link-monitor frame action syslog
disable-interface notify
```

Terms and Definitions

Table 7-4 lists terms and definitions used in this OAM configuration discussion.

Table 7-4 OAM Configuration Terms and Definitions

Term	Definition
frame	A frame is made up of a variable number of symbols formatted with a preamble, header, data blocks, and checksum.
frame monitoring	Monitors link frame errors that occur during a single specified window in seconds.
frame-period monitoring	Monitors frame errors that occur during the reception of a given number of frames relative to the port line rate.
frame-seconds monitoring	Monitors a threshold made up of the number of one second intervals in which one or more frame errors occurs over a specified window of one second intervals.
IEEE 802.3-2008 Clause 57	An OAM standard that allows network operators to monitor and exercise an individual Ethernet link.
OAM	Ethernet Operations, Administration, and Maintenance (OAM) is a collection of standards provided by multiple standards bodies to enable network operators a means to effectively monitor and troubleshoot individual Ethernet links.
OAM client	Contains the essential control operation and state information concerning OAM operations on a specific port, including the handling of received OAM PDUs from the remote client.
OAM client mode	Specifies whether an OAM client can initiate contact with remote peers, and whether it is allowed to send remote loopback control OAM PDUs to the remote peer.
OAM datalink layer monitoring	Monitors Ethernet links for faulty link behavior and is capable of proactive actions when error notifications are received from a remote OAM client.
OAM remote loopback mode	An optional datalink layer loopback mode used for fault detection and performance analysis.
OAM remote loopback request behavior	Provides for the ability to set a local OAM port to either process or ignore a received OAM PDU requesting the port be placed in remote loopback.
symbol	A fixed length electrical or optical waveform on the wire that represents a binary value and is a subcomponent of a frame.
symbol-period monitoring	Monitors the number of symbol errors that occur during the reception of a given number of symbols relative to the port line rate.

8

Port Mirroring Configuration

This chapter provides the following information about configuring and monitoring port mirroring on S-Series devices.

How to Use Port Mirroring in Your Network

Port mirroring, also known as port redirect, is a network traffic monitoring method. It forwards a copy of each received or transmitted frame (or both) from one or more switch ports (source ports) to another port or ports (destination ports) where the data can be studied. Once the bit stream from one or more source ports is mirrored to one or more destination ports, you can further analyze the captured data using an RMON probe, a network sniffer, or an Intrusion Detection System (IDS), without affecting the original port's normal switch operation. You can also mirror, to a policy mirror destination, specific received traffic types for source ports associated with a policy.

Port mirroring is an integrated diagnostic tool for tracking network performance and security that is especially useful for fending off network intrusion and attacks. It is a low-cost alternative to network taps and other solutions that may require additional hardware, may disrupt normal network operation, may affect client applications, and may even introduce a new point of failure into your network. Port mirroring scales better than some alternatives and is easier to monitor. It is convenient to use in networks where ports are scarce.

Enhanced port mirroring provides for following benefits that non-enhanced port mirrors do not:

- L2/L3 multicast egress frames are mirrored
- CNM (Congestion Notification Message) frames that the switch generates are mirrored
- Mirrored egress frames accurately reflect all reframing actions

A maximum of 4 ports can be enabled for enhanced port mirroring.

The S-Series device supports port mirroring for Outbound Rate Limited (ORL) frames.

You can set up the following types of port mirroring relationships on received or transmitted traffic (or both):

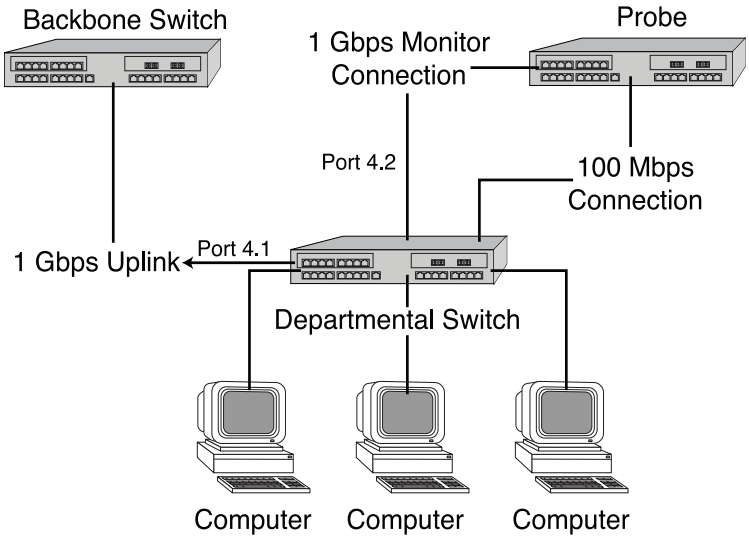
- One-to-one (source port to destination port)
- Many-to-one
- One-to-many

Policy mirroring allows for the same mirror relationships, though policy mirroring applies only to received traffic.

Depending on your network, ports that you can configure to participate in mirroring include physical ports, virtual ports—including Link Aggregation Group (LAG) and host ports—VLAN interfaces, and intrusion detection ports that are members of a LAG. For more information, refer to "Overview of Port Mirroring Configurations" on page 8-4.

You can use port mirroring for analyzing bi-directional traffic and ensuring connectivity between, for example, a departmental switch and its high speed uplink to your backbone switch as shown in Figure 8-1.

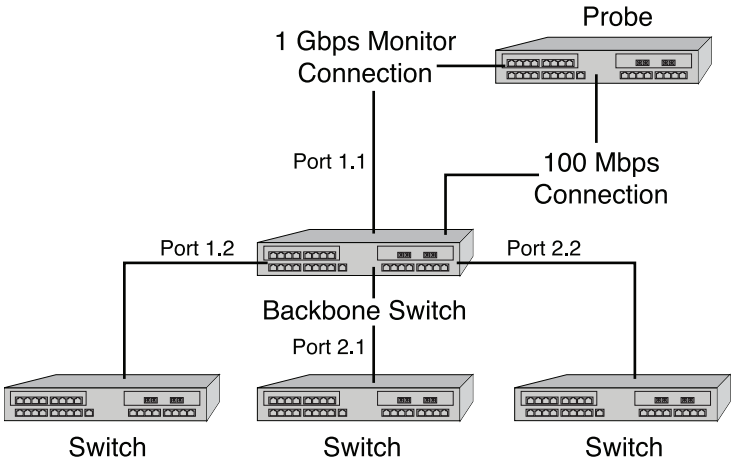
Figure 8-1 Using Port Mirroring to Monitor a Departmental Switch



This one-to-one configuration would allow you to capture traffic in both directions to the backbone uplink port. In this example, you would set a port mirror between departmental switch port 4.1 (source) and the destination port 4.2 connected to the traffic probe.

You can also use port mirroring, for example, to monitor all received traffic or a specific type of received traffic to your backbone switch as shown in Figure 8-2.

Figure 8-2 Using Port Mirroring to Monitor Incoming Traffic to a Backbone Switch



The many-to-one configuration in this example would be possible by setting a port mirror on the backbone between source ports 1.2, 2.2 and 2.1 to destination port 1.1. To monitor a specific type of received traffic (for example, Web traffic—TCP port 80) on the source ports, you would associate the source ports with a policy for that traffic type and associate the policy with a policy mirror destination (the destination port). Destination ports can be ports or LAGs.

The Standalone device and S-Series module supports 15 port-mirrors. These mirrors can be a mixed variety of port, VLAN, and IDS combinations. Any or all mirrors can be configured in a many-to-one mirroring configuration (that is, many sources mirrored to one destination). The LAG that is the destination of an IDS mirror can consist of up to 10 ports.

Note: Standalone devices and S-Series modules that are part of a Virtual Switch Bond (VSB) system support:

- 5 port-mirrors
- 0 IDS mirrors

Examples of port mirroring combinations on an S-Series module include:

- 15 port mirrors
- 15 VLAN mirrors
- 8 port and 7 VLAN mirrors
- 12 port and 3 VLAN mirrors
- 14 port and 1 IDS mirror (where the device mirrors to 10 ports)
- 14 VLAN and 1 IDS mirror (where the device mirrors to 10 ports)

Implementing Port Mirroring

You can implement port mirroring on S-Series devices using simple CLI commands. The source port of a VLAN mirror is a VTAP interface created using the **set vlan interface** command. A VTAP interface provides the data source input of a VLAN mirror and must exist before attempting to create a VLAN port mirror. Once the specific device ports are operationally linked, use the **set port mirroring** command to create a mirroring relationship between your intended source and your destination ports. For policy-based mirroring, use the **set mirror create** and **set mirror ports** commands to create the policy mirror destination. To associate a source port with the policy mirror destination, use the **set policy rule** or the **set policy profile** command to specify both the source port and the policy mirror destination for the policy.

Use the **set port mirroring enhanced** command to enable up to 4 ports to use enhanced port mirroring providing mirroring of L2/L3 egress multicast frames.

Use the **set port mirroring orl** command to enable port mirroring of outbound rate limited frames.

You can also use CLI to operationally disable mirroring, if necessary, and to specify whether to mirror received traffic, transmitted traffic, or both. You can also monitor multicast traffic by enabling IGMP mirroring on specific ports.

Note: It is important to not oversubscribe ports in a mirroring configuration. This can cause bottlenecks and will result in discarded traffic.

Once configured, all packets (network, data, control, and so on) received by the switch will be mirrored. Errored packets will not be mirrored. Unless you disable Spanning Tree on destination ports, they will continue to function as active bridge ports, in accordance with the SMON (Switch Monitoring) standard.

Overview of Port Mirroring Configurations

One or more source ports can be mirrored locally to another physical port within the same S-Series device. In addition, virtual ports and other types of port configurations can also participate in mirroring on Extreme Networks switching devices as described in the following sections:

- LAG Mirrors
- IDS Mirrors
- VLAN Mirrors
- Policy Mirrors

LAG Mirrors

Each S-Series module designates a specific number of virtual link aggregation ports which the Link Aggregation Control Protocol (LACP) can use to dynamically group multiple physical ports into one logical link. Once underlying physical ports (such as ge.x.x) are associated with an aggregator port, the resulting aggregation is represented as one Link Aggregation Group (LAG) with a lag.x.x port designation.

Refer to the Chapter 25, **Link Aggregation Control Protocol (LACP) Configuration** for more information.

When used as a source port in a mirror, LAG ports act identically to a single physical port. Either dynamic or static LAGs can be used as source ports. When used as a destination port in a mirror, the mirror is configured as an IDS mirror as described in the next section.

IDS Mirrors

Since IDS devices are normally bandwidth limited, they benefit from distribution of mirrored data across multiple ports (for example, a 10 Gigabit port mirrored to multiple Gigabit Ethernet ports).

An IDS mirror is a one-to-many port mirror that has been designed for use with an Intrusion Detection System. The target (destination) port of an IDS mirror must be a virtual LAG port that you administratively set, called a static LAG. Once configured, an IDS mirror load-shares traffic among all destination ports in the LAG you set as the port mirror.

An S-Series module hashes the source port conversation based on source and destination IP (SIP/DIP) address pairs and sends the same pairs out the same physical port in the destination mirror. This way, each IDS device will see all of the conversations between a DIP/SIP and will not duplicate the same information out multiple destination ports. When IDS mirroring is enabled, the system performs a Layer 3 lookup for all frames. All non-IP traffic (including control frames) is sent to an arbitrary, “designated” physical out-port. This port is included in the DIP/SIP hash list. If the S-Series module detects a failure of any of the physical ports in the LAG, it will automatically redistribute the DIP/SIP conversations among the remaining ports in the LAG. With IDS mirroring, source traffic is load-shared among all destination ports to ensure no packet loss.

When configuring IDS mirroring on your S-Series device, you must take into consideration the following:

- Only one IDS mirror is allowed per S-Series chassis.
- Ten destination ports must be reserved for an IDS mirror.
- All DIP/SIP pairs will be transmitted out the same physical port.
- All non-IP traffic will be mirrored out the first physical port in a LAG. This port will also be used for IP traffic.

- Port failure or link recovery in a LAG will cause an automatic re-distribution of the DIP/SIP conversations.

Refer to "Example: Configuring an IDS Mirror" on page 8-14 for more information.

VLAN Mirrors

Creating a VLAN and setting a mirror for the VLAN allows you to monitor all traffic to your specified VLAN interface. For example, you could track all data traveling in and out of a confidential group of workstations, such as a Finance VLAN, by analyzing only one connection point. Considerations when configuring VLAN mirrors include:

- A one-to-many or many-to-one VLAN mirror is considered a single destination port.
- Many-to-one mapping allows multiple VLANs to be sent to one specific destination port.
- Oversubscribed traffic will be dropped.

A VTAP interface provides the data source input of a VLAN mirror. VTAP creation is the mechanism for adding a MIB-II interface table entry for a VLAN. A VLAN will not have a MIB-II ifIndex if a VTAP interface does not exist for it. Use the **set vlan interface** command to create a VTAP interface.

Avoiding Bottlenecks

It is especially important to not oversubscribe ports in a mirroring configuration because this can cause bottlenecks and will result in discarded traffic.

If, for example, there are 10 users in VLAN 1, each attached to a 10 Mbps port, when you mirrored VLAN 1 to another 10 Mbps port to which your sniffer is attached, the probe switch would probably have to drop packets at the destination port. Since your purpose in configuring mirroring is to see all of the traffic for VLAN 1, it would be better in this scenario to attach the sniffer to a 100 Mbps port.

Policy Mirrors

The mirror destination mirrors only the received traffic specified in an associated policy. If a source port is associated with both a port mirror and a policy mirror destination, the policy mirror destination takes precedence over the port mirror: the source port traffic specified in the associated policy is mirrored only at the policy mirror destination port, not at the port mirror.

For example, a port mirror is created to mirror, on the destination port ge.1.2, the traffic received at source port ge.1.1. Port ge.1.1 is also associated with a policy for Web traffic. That policy has a policy mirror destination with ge.1.3 as the destination port. Because the policy mirror destination takes precedence over the port mirror, the Web traffic for port ge.1.1 is mirrored to port ge.1.3 only. Port ge.1.2 mirrors all other traffic with the exception of the Web traffic.

Note: The S-Series hardware does not support both port mirroring and outbound rate limiting of a frame. Port mirroring of an outbound rate limited frame is disabled by default. Use the **set port mirroring orl enable** command to enable port mirroring and disable outbound rate limiting of outbound rate limited frames.

Configuring Port Mirrors

Note: When a port mirror or policy mirror destination is created, It is automatically enabled.

Reviewing Port Mirroring

Use this command to display the status of port mirroring and information about any mirrors configured:

```
show port mirroring
```

Use this command to display the status of enhanced port mirroring on the device:

```
show port mirroring enhanced
```

Examples

This example shows that no port mirrors are configured on the device:

```
S Chassis(rw)->show port mirroring
No Port Mirrors configured.
IGMP Multicast Mirror status Disabled
Mirror Outbound Rate Limited Frames : Disabled
```

This example shows that a port mirror is configured between source port vtap.0.5 and ge.1.1 and that both received (Rx) and transmitted (Tx) frames will be monitored. It also shows that mirroring status is currently administratively and operationally enabled. A mirror must be administratively enabled (as described in the next section) and its source and destination ports must have an active link for operational status to be enabled.

```
S Chassis(rw)->show port mirroring
Port Mirroring
==============

 Source Port        = vtap.0.5
 Target Port        = ge.1.1
 Frames Mirrored    = Rx and Tx
 Admin Status       = enabled
 Operational Status = enabled

Mirror Outbound Rate Limited Frames : Disabled
```

This example shows how to enable ports ge.3.1 and ge.3.4 for enhanced port mirroring and to display enhanced port mirroring status for this device:

```
S Chassis(rw)->set port mirroring enhanced enable ge.3.1,4

S Chassis(rw)->show port mirroring enhanced
Enhanced Port Mirroring
=======================

Max Enhanced Mirror Source Ports: 4

Port         Admin Status Oper Status
------------ ------------ ------------
ge.3.1       enabled      enabled
ge.3.4       enabled      disable
S Chassis(rw)->
```

Reviewing Policy Mirror Destinations

Use this command to display the status of policy mirror destinations and information about any mirror destinations configured:

show mirror *control-index-list*

Setting Port or VLAN Mirroring

Use this command to create a new mirroring relationship, or to enable or disable an existing mirroring relationship. Optionally, you can specify whether to mirror received frames, transmitted frames, or both:

set port mirroring {**create** | **disable** | **enable**} *source destination* [**both** | **rx** | **tx**]

If not specified, **both** received and transmitted frames will be mirrored.

The S-Series hardware does not support both port mirroring and outbound rate limiting of a frame. Outbound rate limiting is enabled and port mirroring is disabled by default for outbound rate limited frames. Use this command to set port mirroring behavior for outbound rate limited frames:

set port mirroring orl {**enable** | **disable**}

Note: By default, when you create a port mirror, the port mirror is enabled.

Examples

This example shows how to create a port mirror to mirror frames sourced on port ge.1.4 and received on port ge.1.11:

```
S Chassis(rw)->set port mirroring create ge.1.4 ge.1.11 rx
```

This example shows how to create a many-to-one mirroring configuration between source ports ge.1.2, ge.1.3 and ge.1.4, and target port ge.1.10. By default, frames in both directions will be monitored:

```
S Chassis(rw)->set port mirroring create ge.1.2-4 ge.1.10
```

This example enables port mirroring and enables outbound rate limiting of outbound rate limited frames:

```
S Chassis(rw)->set port mirroring orl enable
```

This example shows how to configure mirroring from source port 5 to destination port 1 in slot 1 (ge.1.1):

```
S Chassis(rw)->set vlan interface 5 create
S Chassis(rw)->set port mirroring create vtap.0.5 ge.1.1
S Chassis(rw)->show port mirror
Port Mirroring
==============

 Source Port        = vtap.0.5
 Target Port        = ge.1.1
 Frames Mirrored    = Rx and Tx
 Admin Status       = enabled
 Operational Status = Unavailable resources

Mirror Outbound Rate Limited Frames : Disabled
```

Note: If you configure a port mirror on an uplink (tagged) port, make sure the port is assigned to egress frames with that VLAN tag. Refer to Chapter 24, **VLAN Configuration** for more information about configuring VLANs.

Setting Enhanced Port Mirroring

Up to 4 port mirrors created using the set port mirror command can be enabled for enhanced port mirroring using the set port mirroring enhanced command.

Example

This example enables enhanced port mirroring on ports ge.3.1 and ge.3.4:

```
S Chassis(rw)->set port mirroring enhanced ge.3.1,4
```

Setting Policy Mirror Destinations

Use these commands to create a policy mirror destination and to associate a destination port.

- **set mirror create** *control-index-list*
- **set mirror ports** *port-string control-index-list* [**append**]

You must also associate the policy mirror destination with either a policy role or a policy rule, which you then must associate with a policy role, by setting the mirror-index for the **mirror-destination** parameter in the following commands:

- **set policy profile**
- **set policy rule admin-profile**

The mirror-index value in the **set policy** commands is the same as the control-index-list value in the **set mirror** commands.

For more information about the policy commands, see Chapter 26, **Policy Configuration**.

You can also specify the number of packets at the beginning of a flow to mirror by using the **set mirror mirrorN** command.

set mirror *control-index* **mirrorN** *mirrorN-packets*

Deleting Mirrors

Use this command to clear a port mirroring configuration:

```
clear port mirroring source destination
```

Use this command to clear a policy mirror destination:

```
clear mirror ports port-string control-index-list
```

Remote Mirroring Using a Layer 2 GRE Tunnel

The S-Series supports remote mirroring using a Layer 2 (L2) GRE tunnel. Any L2 traffic (unicast, multicast, or broadcast) that can be mirrored with a non-L2 GRE remote mirror can be mirrored with an L2 GRE remote mirror. The mirror source port is the source of the mirrored packets found on the local router of interest. The mirror encapsulates the L2 traffic seen by the mirrored source port in an IP GRE header and delivers it to the tunnel destination address. The tunnel destination address is the ultimate destination port of the tunnel where packets are decapsulated and delivered to the port local to the remote router.

The tunnel destination port resides on a remote router with the appropriate L2 GRE tunnel configuration back to the ultimate mirror destination. The intermediate mirror destination port is the actual mirror destination of the local SMON or policy configuration. Once the L2 tunnel is enabled and fully configured, packets sent to an intermediate destination are encapsulated and forwarded for any SMON or policy port mirrors that use the mirror destination port. Once the tunnel is enabled, the tunnel destination port is in internal loopback mode and can no longer be used as a normal switch port.

Configuration of a remote mirror using a L2 GRE tunnel consists of:

- Creating a tunnel interface in global configuration mode, with a loopback source address and a route to the destination
- Configuring the mirrored port:
  - Configure an SMON port mirror by creating the port mirror, specifying the L2 GRE tunnel destination as the destination port, or
  - Configure a policy port mirror by creating a mirror destination and specifying a policy to be associated with the mirror

The following CLI input sets up the L2 GRE tunnel for mirrored port **ge.1.1** by:

- Configuring **loop.0.1** as the loopback source address (**88.88.88.1/32**) for the mirrored port
- Configuring VLAN 20 as the VLAN interface the tunnel resides on
- Entering configuration mode for tunnel **5**
  - Configuring the tunnel **5** destination address (**99.99.99.1**)
  - Setting the tunnel mode to GRE L2 and specifying **ge.1.8** as the mirror destination
  - Enabling the mirrored tunnel which allocates the necessary resources to support mirrored packets

- Configuring a static route to the mirror destination

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface loop.0.1
S Chassis(rw-config-intf-loop.0.1)->ip address 88.88.88.1/32
S Chassis(rw-config-intf-loop.0.1)->no shutdown
S Chassis(rw-config-intf-loop.0.1)->exit
S Chassis(rw-config)->interface vlan.0.20
S Chassis(rw-config-intf-vlan.0.20)->ip address 6.1.1.1 255.255.255.0 primary
S Chassis(rw-config-intf-vlan.0.20)->no shutdown
S Chassis(rw-config-intf-vlan.0.20)->exit
S Chassis(rw-config)->interface tunnel 5
S Chassis(rw-config-intf-tun.0.5)->tunnel destination 99.99.99.1
S Chassis(rw-config-intf-tun.0.5)->tunnel mode gre l2 ge.1.8
S Chassis(rw-config-intf-tun.0.5)->tunnel mirror enable
S Chassis(rw-config-intf-tun.0.5)->tunnel source 88.88.88.1
S Chassis(rw-config-intf-tun.0.5)->no shutdown
S Chassis(rw-config-intf-tun.0.5)->exit
S Chassis(rw-config)->ip route 99.99.99.1/32 6.1.1.2 interface vlan.0.20 1
S Chassis(rw-config)->exit
```

The following CLI input creates an SMON port mirror specifying the L2 GRE mirror-destination (ge.1.8) as the destination:

```
S Chassis(rw)->set port mirror create ge.1.1 ge.1.8 both
S Chassis(rw)->
```

The following CLI input sets up the L2 GRE tunnel on the mirror destination router by:

- Configuring **loop.0.1** as the loopback source address (**99.99.99.1/32**) for the mirrored port
- Configuring VLAN 33 as the VLAN interface the tunnel resides on
- Entering configuration mode for tunnel **12**
  - Configuring the tunnel **12** destination address (**88.88.88.1**)
  - Setting the tunnel mode to GRE L2 and specifying **ge.2.4** as the mirror destination

  You do not enable the mirrored tunnel on the mirror destination router. The L2 GRE-encapsulated packets:

  - Arrive with source **88.88.88.1** and destination **99.99.99.1**
  - Are decapsulated and forwarded out physical port **ge.2.4**

  A PC running a packet-monitoring program, such as WireShark, can be attached to this port. The packet-monitoring program displays the L2 traffic that is seen by ge.1.1 on the router on which the mirrored port resides.
- Configure a static route to the router on which the mirror port resides (**88.88.88.1/32**)

```
S-K-Series(rw)->configure
S-K-Series(rw-config)->interface loop.0.1
S-K-Series(rw-config-intf-loop.0.1)->ip address 99.99.99.1/32
S-K-Series(rw-config-intf-loop.0.1)->no shutdown
S-K-Series(rw-config-intf-loop.0.1)->exit
S-K-Series(rw-config)->interface vlan.0.33
```

```
S-K-Series(rw-config-intf-vlan.0.33)->ip address 5.1.1.1 255.255.255.0 primary
S-K-Series(rw-config-intf-vlan.0.33)->no shutdown
S-K-Series(rw-config-intf-vlan.0.33)->exit
S-K-Series(rw-config)->interface tunnel 12
S-K-Series(rw-config-intf-tun.0.12)->tunnel destination 88.88.88.1
S-K-Series(rw-config-intf-tun.0.12)->tunnel mode gre 12 ge.2.4
S-K-Series(rw-config-intf-tun.0.12)->tunnel source 99.99.99.1
S-K-Series(rw-config-intf-tun.0.12)->no shutdown
S-K-Series(rw-config-intf-tun.0.12)->exit
S-K-Series(rw-config)->ip route 88.88.88.1/32 5.1.1.2 interface vlan.0.33 1
S-K-Series(rw-config)->exit
S-K-Series(rw)->
```

The following CLI input:

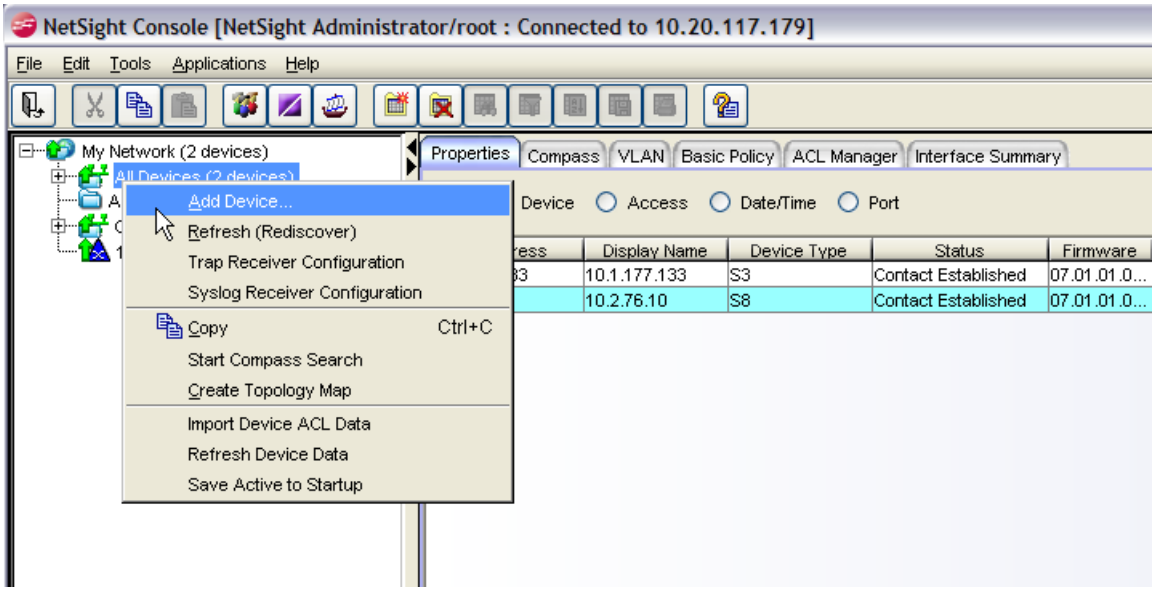
- Creates policy profile **1** and applies mirror-destination index **2** to the profile
- Creates an admin-profile rule for port **ge.1.1** and applies it to policy profile **1**
- Creates policy port mirror index **2**
- Sets port **ge.1.8** as the destination for port mirror index **2**

```
S Chassis(rw)->set policy profile 1 mirror-destination 2
S Chassis(rw)->set policy rule admin-profile port ge.1.1 mask 16 port-string
ge.1.1 admin-pid 1
S Chassis(rw)->set mirror create 2
S Chassis(rw)->set mirror ports ge.1.8 2
```

Example: Configuring and Monitoring Port Mirroring

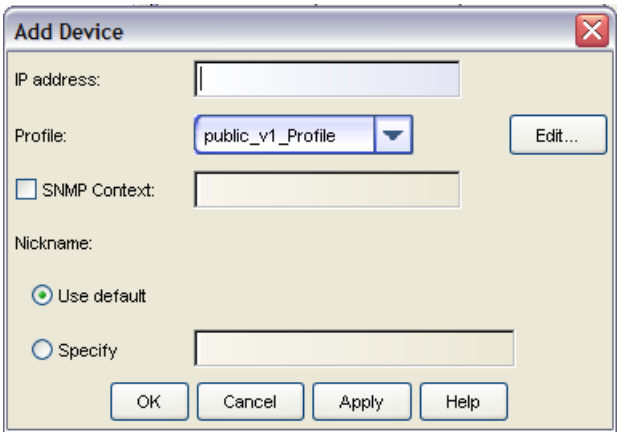
This section describes how to use Extreme Networks NetSight Console from a Network Management Station (NMS) to display RMON statistics for monitoring port mirroring.

1. Log onto Netsight Console.



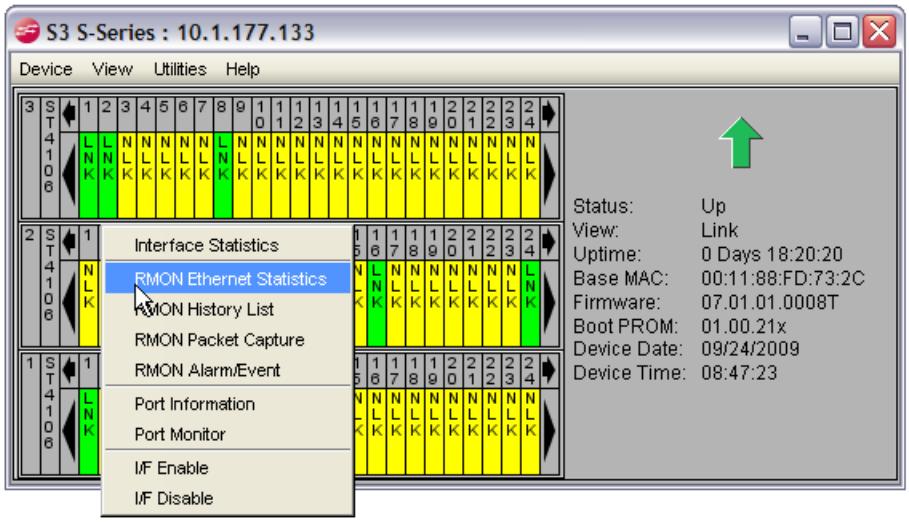
2. On the console main screen, expand **My Network** in the file directory tree, right-click **All Devices**, and select **Add Device**.

   The Add Device screen displays.



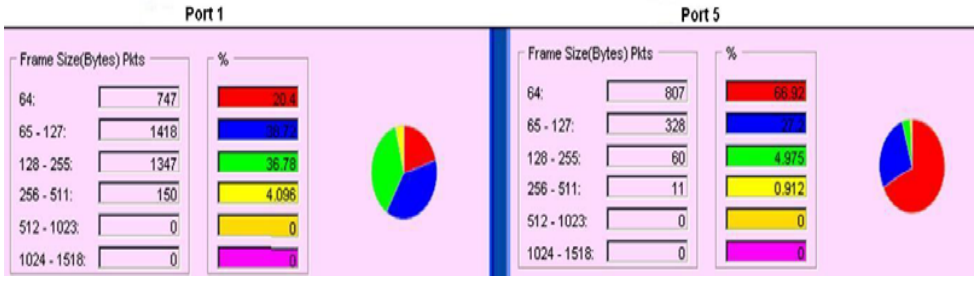
3. Model the S-Series device by entering its **IP address** in the field provided. Click **OK**.
4. On the console main screen, expand **All Devices** in the file directory tree to show the IP address(es) of the device(s) you just modeled.
5. Right click on the IP address of the S-Series device and select **Device Manager**.

   The device manager screen displays for the S-Series device.

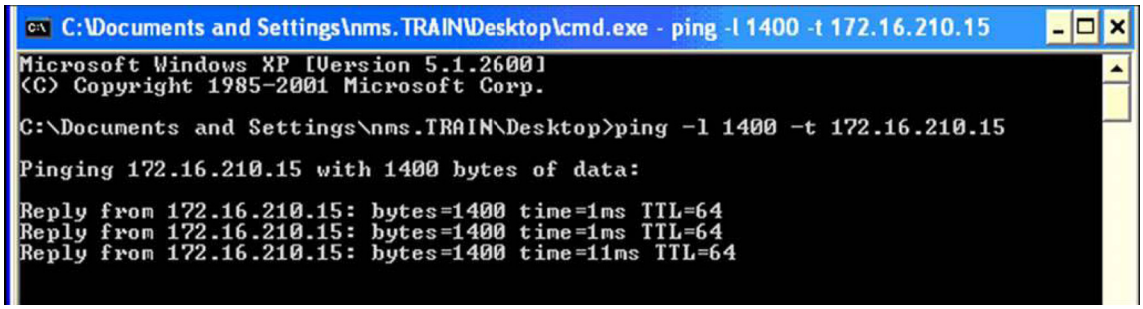


6. Right click on port 1 (ge.1.1) and select **RMON Ethernet Statistics**.
7. Repeat step 9 for port 5 (ge.1.5).

RMON Ethernet statistics charts will display for ports 1 and 5.



8. Note that the section of the two charts that shows the frame count by frame size lists no larger size frames (512-1518 bytes). In the next step, you will create large frames.
9. Open the Command Prompt window and set up a continuous ping to the S-Series device, as shown below. Use -l 1400 to set the size of the ping frame to 1400 bytes and **-t** to set a continuous ping.



10. Refer back to the RMON Ethernet Statistics windows opened in Steps 9 and 10. You should see the number of **1024 - 1518** frames incrementing on Port 1 because the NMS is connected on this port. You should also see that these larger size frames are not incrementing on Port 5.
11. From the terminal session with the S-Series device, create a port mirroring instance with port 1 (ge.1.1) as the source and port 5 (ge.1.5) as the destination port.

```
S Chassis(su)->set port mirroring create ge.1.1 ge.1.5 both
```

12. Verify the mirroring configuration.

```
S Chassis(su)->show port mirroring

Port Mirroring
==============
Source Port = ge.1.1
Target Port = ge.1.5
Frames Mirrored = Rx and Tx
Port Mirroring Admin status = enabled
Port Mirroring Oper status = enabled
```

13. Refer again to the RMON Ethernet Statistics windows and notice that both port 1 and port 5 are now incrementing the larger size frames. If you connected a network analyzer to port 5, you would see these frames being received and transmitted on port 1.

Example: Configuring an IDS Mirror

S-Series devices support IDS mirroring on ports that are members of a Link Aggregation Group (LAG). A maximum of eight ports are allowed per LAG port. Only manually formed (static) LAGs can be used as mirrored destination ports.

Procedure 8-1 shows how to create a static LAG and then create an IDS mirror to that LAG port destination. In this example, ports ge.1.1 through ge.1.5 are administratively set to form lag.0.21, which is then set to mirror traffic from port ge.1.10.

For more information on command parameters used in LAG configuration, refer to the Link Aggregation chapter.

Note: When creating a static LAG for port mirroring, you must assign a unique admin key to aggregating ports. If ports other than the desired underlying physical ports share the same *admin key* value, aggregation will fail or undesired aggregations will form.

Procedure 8-1 Configuring a Static LAG for an IDS Mirror

Step	Task	Command(s)
1.	Create a static LAG aggregating ports ge.1.1 through ge.1.5 into LAG port 21 and assign a unique admin key to that LAG port.	**set lacp static lag.0.21 key 4000 ge.1.1-5**
2.	Create a port mirror between source port ge.1.10 and the static LAG.	**set port mirror create ge.1.10 lag.0.21 both**

Example: Configuring a Policy Mirror Destination

In this example, policy mirror destination 2 is created with ge.1.3 as the destination port for the mirrored traffic. This mirror destination is associated with policy that mirrors all received TCP port 80 traffic on port ge.1.1

```
S Chassis(su)->set mirror create 2
S Chassis(su)->set mirror ports ge.1.3 2
S Chassis(su)->set policy profile 1 name tcp80
S Chassis(su)->set policy rule 1 tcpsourceportip 80 forward mirror-destination 2
S Chassis(su)->set policy port ge.1.1 1
```

9

System Configuration

This document provides the following information about system configuration on the Extreme Networks S-Series platforms.

Chassis Compatibility Mode

There are currently two generations of S-Series fabric module. The S-Series S130, S150, and S155 modules belong to the first generation. For the remainder of this discussion, this module grouping is identified as **S130/150/155**. The second generation consists of the S-Series S140 and S180 modules. This module grouping is identified as **S140/180**. These two fabric generations have capability differences that are not compatible with each other. Where allowed, mixed systems will modify the capabilities of the S140/180 modules. You need to be aware of the supported configurations for these two S-Series fabric generations when installing modules into an S-Series physical chassis. Once a supported module configuration is determined, an appropriate compatibility mode must be set to assure that all module configurations operate at supported levels.

Fabric and I/O Module Restrictions

The two S-Series fabric versions are not compatible. Both fabric module versions can not be installed in a mixed configuration on the same physical chassis. You can mix versions of I/O modules in the same chassis, but if an S140/180 I/O module exists in the chassis, only S180 fabric modules can be installed in that chassis.

There are two exception to mixing I/O version modules in the same chassis:

- You can not mix S130 and S140 I/O modules in an S3 chassis
- S130/150 I/O modules can not be used in VSB configurations using dedicated VSB hardware interconnect ports (Refer to "VSB Interconnect Link Configuration" on page 5-6 for VSB hardware interconnect mode details).

S-Series Module Compatibility Mode

Once S-Series version module restrictions are met, you must assure that an appropriate compatibility mode is configured for the system. There are three compatibility mode settings: **auto**, **v1**, and **v2**.

Auto compatibility mode: When configured, the **auto** fabric compatibility mode actively determines the appropriate V1 or V2 setting for the system only when booting for the first time in a cleared (default) configuration state. Based upon the hardware installed at boot time, the appropriate fabric compatibility mode is operationally set and persists across subsequent system boots. If subsequent hardware changes occur requiring a module compatibility mode change, the module compatibility mode does not get changed, and any new hardware not appropriate to the current operational compatibility mode remains non-operational upon system boot. If changes that are not compatible with the current configuration occur subsequent to an initial auto compatibility mode boot, the appropriate **v1** or **v2** compatibility mode must be administratively entered, unless an action has occurred that causes the auto setting to be reset to its initial state.

The auto compatibility mode is in its initial state under the following conditions:

- The first boot up of 8.11 FW and newer firmware
- Any time the configuration is lost due to a **clear config** command being entered, use of switch 7 on all fabrics, or all fabrics in the device are newly installed
- Issuing the **clear chassis compatibility** or **set chassis compatibility auto** commands

Note: Chassis compatibility mode defaults to **auto**. You do not need to modify this default setting so long as you either do not modify the module configuration in the chassis or the modification of the module configuration is appropriate to the current operational chassis compatibility mode. The current operational chassis compatibility mode is displayed in the **show chassis compatibility-mode** command output.

When displaying chassis compatibility mode information, the display indicates both the admin setting and the operational setting. In auto compatibility mode, the admin display field will display **auto**. The operational setting will display the compatibility mode that auto mode selected during a clear config system boot: either **v1** or **v2**. When in admin **auto** compatibility mode, use the **show chassis compatibility-mode** command to determine the selected operational chassis compatibility mode.

V1 compatibility mode: V1 compatibility mode is specified for chassis that have only S130/150/155 modules installed or for chassis with supported mixed version modules installed. See "Fabric and I/O Module Restrictions" on page 9-2 for restrictions associated with mixed version module configurations. S140/180 modules in a mixed configuration operate at a modified capability level in order to co-exist with the S130/150/155 modules installed in the chassis.

V2 compatibility mode: V2 compatibility mode is specified for chassis that have only S140/180 modules. Should an S130/150/155 module be present when the operational compatibility mode is set to v2, the S130/150/155 modules will not become active.

Note: When administratively changing the compatibility mode to a mode that will change the current operational compatibility mode an appropriate warning displays and the system resets.

S-Series Module Compatibility Mode Capabilities

There are two possible operational modes when installing S-Series modules: **v1** and **v2**. S130/150/155 modules can only operate in v1 mode. S140/180 modules are capable of operating in v1 mode with modified capabilities, allowing compatible operation with S130/150/155 modules, or in v2 mode taking full advantage of the module's capabilities. You can determine the compatibility mode capabilities of an installed S-Series module using the **show chassis compatibility-mode capabilities** command.

Supported VSB Compatibility Mode Configurations

Hardware interconnect ports can only be used if both chassis are set to v2 compatibility mode. Software assisted connectivity ports must be used if either physical chassis compatibility mode is set to v1.

See "VSB Interconnect Link Configuration" on page 5-6 for a bonding mode and interconnect port discussion.

This example shows how to set the chassis compatibility mode for chassis index 1 with a mixed configuration to **auto**:

```
S Chassis(rw)->set chassis compatibility-mode auto
S Chassis(rw)->show chassis compatibility-mode chassis-index 1
Chassis Index:                              1
Current Fabric Compatibility Admin Mode:    auto
Current Fabric Compatibility Oper Mode:     v1
```

This example shows how to set the chassis compatibility mode for chassis index 2 containing all S180 modules to v2:

```
S Chassis(rw)->set chassis compatibility-mode v2 chassis-index 2
S Chassis(rw)->show chassis compatibility-mode chassis-index 2
Chassis Index:                              2
Current Fabric Compatibility Admin Mode:    v2
Current Fabric Compatibility Oper Mode:     v2
```

System Properties Overview

Table 9-1 lists system parameter default values.

Table 9-1 Default System Parameters

Parameter	Description	Default Value
IP Gratuitious ARP	Provides an ARP announcement packet containing valid sender hardware and protocol addresses for the host that sent it.	disabled

Table 9-1 Default System Parameters (continued)

Parameter	Description	Default Value
System Utilization Threshold	Sets the threshold for sending CPU utilization notification messages.	800 (80%)
MTU	Sets the path MTU discovery protocol on the device.	enabled

You must configure your S-Series device with an IP interface and an IP address. You can also configure other system properties on the S-Series device. See Table 9-2.

Table 9-2 System Properties Configuration

Task	Command
Set IP interfaces. You may specify an IP interface as the default management IP interface.	**set ip interface** *interface-name* [**default**]
Set the system IP address, subnet mask and default gateway. If not specified, *ip-mask* will be set to the natural mask of the *ip-address* and *ip-gateway* will be set to the *ip-address*. If not specified, the first IP interface configured on a system becomes the default IP interface.	**set ip address** *ip_address* [**mask** *ip_mask*] [**gateway** *ip_gateway*] [**interface** *interface-name*]
Set the threshold for sending CPU utilization notification messages. A value of **0** will disable utilization notification messages.	**set system utilization threshold** *threshold*
Change the time of day on the system clock.	**set time** [*mm/dd/yyyy*] [*hh:mm:ss*]
Enable or disable the daylight savings time function.	**set summertime** {**enable** | **disable**} [*zone*]
Configure one of the following:	
• Specific dates to start and stop daylight savings time. These settings will be non-recurring and will have to be reset annually.	**set summertime date** *start_month start_date start_year start_hr_min end_month end_date end_year end_hr_min* [*offset_minutes*]
• Recurring daylight savings time settings. These settings will start and stop daylight savings time at the specified day of the month and hour each year and will not have to be reset annually.	**set summertime recurring** *start_week start_day start_month start_hr_min end_week end_day end_month end_hr_min* [*offset_minutes*]
(Optional) Configure a name for the system. A name string containing a space in the text must be enclosed in quotes as shown in the example below. If *string* is not specified, the system name will be cleared.	**set system name** [*string*]
(Optional) Identify the location of the system. A location string containing a space in the text must be enclosed in quotes as shown in the example below. If *string* is not specified, the location name will be cleared.	**set system location** [*string*]

Table 9-2 System Properties Configuration (continued)

Task	Command
(Optional) Identify a contact person for the system. A contact string containing a space in the text must be enclosed in quotes as shown in the example below. If *string* is not specified, the contact name will be cleared.	**set system contact** [*string*]
Set the alias, a text name, for a physical object. If *string* is not specified, the specified alias will be cleared.	**set physical alias** {[**chassis**] | [**backplane** *backplane*] | [**slot** *slot*] | [**module** *module*] | **sub-module** *slot module* | [**powersupply** *powersupply*] | [**powersupply-slot** *powersupply-slot*] | [**poe-powersupply-***poe-powersupply*] | [**fan** *fantray*] | [**fan-slot** *fantray*] | [**port** *port-string*]} [*string*]
Set the asset ID for a physical object.	**set physical assetid** {[**chassis**] | [**module** *module*] | [**powersupply** *powersupply*] | [**poe-powersupply-***poe-powersupply*] | [**fan** fantray]} *string*
Disable or re-enable path MTU discovery protocol on the device.	**set mtu** {**enable** | **disable**}

Table 9-3 lists system properties management and display commands for S-Series devices.

Table 9-3 System Properties Management and Display Commands

Task	Command
Display the gratuitous ARP processing behavior.	**show ip gratuitous-arp**
Display system information, including contact information, power and fan tray status and uptime.	**show system**
Display the system's hardware configuration.	**show system hardware**
Display system resource utilization information.	**show system utilization** [**cpu** | **process** | **storage**] [**slot** *slot*]
Display the current time of day in the system clock.	**show time**
Display daylight savings time settings.	**show summertime**
Display the alias (a text name) for one or more physical objects.	**show physical alias** {[**chassis**] | [**backplane** *backplane*] | [**slot** *slot*] | [**module** *module*] | **sub-module** *slot module* | [**powersupply** *powersupply*] | [**powersupply-slot** *powersupply-slot*] | [**poe-powersupply-***poe-powersupply*] | [**fan** *fantray*] | [**fan-slot** *fantray*] | [**port** *port-string*]}
Display the asset ID for a physical object.	**show physical assetid** {[**chassis**] | [**module** *module*] | [**powersupply** *powersupply*] | [**poe-powersupply-***poe-powersupply*] | [**fan** fan]}
Display the status of the path MTU (maximum transmission transmission) discovery protocol on the device.	**show mtu**
Display information about scheduled device resets.	**show reset**

Table 9-3 System Properties Management and Display Commands (continued)

Task	Command
Display output for technical support-related commands. Optionally, you can write this output to a file.	**show support** *filename*
Clear the IP interface.	**clear ip interface** *interface-name*
Clear an IP address.	**clear ip address** *ip-address*
Stop all gratuitous ARP processing.	**clear ip gratuitous-arp**
Clear the threshold for sending CPU utilization notification messages.	**clear system utilization**
Clear the daylight savings time configuration.	**clear summertime**
Reset the alias for a physical object to a zero-length string.	**clear physical alias** {[**chassis**] | [**backplane** *backplane*] | [**slot** *slot*] | [**module** *module*] | **sub-module** *slot module* | [**powersupply** *powersupply*] | [**powersupply-slot** *powersupply-slot*] | [**poe-powersupply-***poe-powersupply*] | [**fan** *fantray*] | [**fan-slot** *fantray*] | [**port** *port-string*]}
Reset the asset ID for a module to a zero-length string.	**clear physical assetid** {[**chassis**] | [**module** *module*] | [**powersupply** *powersupply*] | [**poe-powersupply-***poe-powersupply*] | [**fan** fan]}
Reset the state of the path MTU discovery protocol back to enabled.	**clear mtu**
Reset the device without losing any user-defined configuration settings or to display information about device resets.	**reset** {[*mod* | **system**] [**cancel**]}
Reset an option module CPU.	**reset nemcpu** *mod.nemcpu*
Schedule a system reset at a specific future time. This feature is useful for loading a new boot image.	**reset at** *hh:mm* [*mm/dd*] [*reason*]
Schedule a system reset after a specific time. This feature is useful for loading a new boot image.	**reset in** *hh:mm* [*reason*]
Clear all user-defined switch and router configuration parameters for one or all modules.	**clear config** *mod_num* | **all**

System Properties Example

```
S Chassis(rw)->set ip interface vlan.0.5 default
S Chassis(rw)->set ip address 10.1.10.1 mask 255.255.128.0 gateway 10.1.10.1


S Chassis(rw)->set system utilization threshold 1000
S Chassis(rw)->set time 7:50:00
S Chassis(rw)->set summertime enable
S Chassis(rw)->set summertime recurring second Sunday March 02:00 first Sunday
November 02:00 60
S Chassis(rw)->set system name "Information Systems"
S Chassis(rw)->set system location "Bldg N32-04 Closet 9"
S Chassis(rw)->set system contact "Joe Smith"
```

```
S Chassis(rw)->set physical alias chassis chassisone
S Chassis(rw)->set physical assetid module 1 blade1
```

User Management Overview

An admin user (super user) can create user accounts, set the system password, and set the system lockout. Users with read-write access can change their own passwords. See Procedure 9-1.

The S-Series device supports up to 16 user accounts, including the admin account, which cannot be disabled or deleted.

The S-Series supports security profiles that determine user access to certain commands and can also limit parameter settings for certain commands. The security profiles supported are normal and C2. The normal security profile provides standard user access based upon the configured user mode: super-user, read-write, and read-only. C2 is defined as Controlled Access Protection mode and is a security rating established by the U.S. National Computer Security Center (NCSC) and granted to products that pass Department of Defense (DoD) Trusted Computer System Evaluation Criteria (TCSEC) tests. A C2 rating ensures the minimum allowable levels of confidence demanded for government agencies and offices and other organizations that process classified or secure information. Use the **set security profile** command to set the security profile to either normal or C2 for the device. C2 mode can affect command availability and parameter value defaults and ranges. If C2 security mode affects a command, it is specified in the command entry found in the *Extreme Networks S-Series CLI Reference*.

Access to the boot menu during startup can be disabled. Access to the boot menu during startup is enabled by default.

The S-Series supports enabling of the Federal Information Processing Standards (FIPS) mode. FIPS mode is a mode where only FIPS approved authentication and encryption algorithms and methods are used. The current implementation supports the SHA1 algorithm in FIPS mode. Use the **set security fips mode** command to enable FIPS mode on the device.

User management configuration also includes the following:

- "Setting the Authentication Login Method" on page 9-9
- "Using WebView" on page 9-10

Procedure 9-1 User Management Configuration

Step	Task	Command(s)
1.	Create a new user login account, or disable or enable an existing account.	**set system login** *username* [**read-write** | **read-only** | **super-user**] [**enable** | **disable**] [**password** {*password* | **aging** {*days* | **disable** | **system**}] [**allowed-interval** {*HH:MM HH:MM*}] [**allowed-days** {[**Sun**] [**Mon**] [**Tue**] [**Wed**] [**Thu**] [**Fri**] [**Sat**]}] [**simultaneous-logins** *num*] [**local-only** {**yes** | **no**}]
2.	Change system default passwords or set a new login password on the CLI. (Only available to users with super-user access.)	**set password** [*username*]

Procedure 9-1 User Management Configuration (continued)

Step	Task	Command(s)
3.	Configure system password parameters. A system password can contain the following special characters: !@#$%^&*()-=[]\;?,./`	**set system password** [**aging** {*days* | **disable**}] [**history** {*size*}] [**length** *characters*] [**min-required-chars** {[**uppercase** *characters*] [**lowercase** *characters*] [**numeric** *characters*] [**special** *characters*]}][**require-at-creation** {**yes** | **no**}] [**allow-duplicates** {**yes** | **no**}] [**allow-user-id** {**yes** | **no**}] [**substring-match-len** *characters*] [**allow-repeating-chars** {*num* | **yes** | **no**}] [**change-first-login** {**yes** | **no**} [**all**]] [**change-frequency** *minutes* [**all**]] [**expire-warning** *days*] [**grace-period** {**logins** *num* | **time** *days*}]
4.	Optionally, disable access to the boot menu during bootup. Access to the boot menu is enabled by default.	**set security boot-access** {**enable** | **disable**}
5.	Set the number of failed login attempts before locking out (disabling) a read-write or read-only user account, the number of minutes to lockout the default admin super user account after maximum login attempts, and the number of inactive days before a non-superuser account is locked out. If you set **inactive** to 0, no accounts will be locked out due to inactivity. Once a user account is locked out, it can only be re-enabled by a super user with the **set system login** command.	**set system lockout** {[**attempts** *attempts*] [**time** *minutes* [**all**]] [**port** {**enable** | **disable**] [**inactive** *days* [**all**]] [**emergency-access**]}
6.	Optionally, enable FIPS mode on the device. Fips mode is disabled by default.	**set security fips mode** {**enable** | **disable**}
7.	Optionally, set the device's security profile. The security profile defaults to normal.	**set security profile** {**c2** | **normal**}

Table 9-4 lists user account management and display commands for S-Series devices.

Table 9-4 User Account Management and Display Commands

Task	Command
To display user login account information.	**show system login [-verbose]**
To display current password configuration settings.	**show system password**
To display settings for locking out users.	**show system lockout**
To display the current boot access state for this device.	**show security boot-access**
To display the current security FIPS mode state for this device.	**show security fips mode**
To display the current security profile for this device.	**show security profile**

Table 9-4 User Account Management and Display Commands (continued)

Task	Command
To remove a local login user account or to reset a specified option to its default value. The account is removed if no optional parameters are entered.	**clear system login** *username* [**allowed-interval**] [**allowed-days**] [**password** [**aging**]] [**simultaneous-logins**] [**local-only**]
To reset system lockout parameters to default values.	**clear system lockout** [**attempts**] [**time**] [**inactive**]
To clear local login password parameters to default values. If no options are specified, all options are reset to default values.	**clear system password** [**aging**] [**history**] [**length**] [**min-required-chars** {[**uppercase**] [**lowercase**] [**numeric**] [**special**]}] [**require-at-creation**] [**allow-duplicate**] [**allow-user-id**] [**substring-match-len**] [**allow-repeating-chars**] [**change-first-login**] [**change-frequency**] [**expire-warning**] [**grace-period**]
To reset access to the boot menu during bootup to the default state of enabled.	**clear security boot-access**
To reset FIPS mode state to the default value of disabled on the device.	**clear security fips mode**
To reset the device security profile to the default value of normal.	**clear security profile**

User Management Example

This example includes the following:

- Configuring system password parameters
- Creating a new user account
- Setting the password for the new user account
- Setting the system lockout parameters

```
S Chassis(su)->set system password age 60 length 6 allow-repeating-chars no
S Chassis(su)->set system login netops read-write enable
S Chassis(su)->set password rw
Please enter new password: ********
Please re-enter new password: ********
Password changed.
S Chassis(su)->set system lockout attempts 5 time 30 inactive 60
```

Setting the Authentication Login Method

By default, the authentication login method is set to any, which uses the following precedence order:

- TACACS+
- RADIUS
- Local

Procedure 9-2 Authentication Configuration

Step	Task	Command(s)
1.	Change the default authentication login method. This command is not available to Read-Write users while in C2 security mode.	**set authentication login** {**any** | **local** | **radius** | **tacacs**}
2.	Display the current authentication login method to verify your changes.	**show authentication login**
3.	If necessary, reset the authentication login method to the default setting (any). This command is not available to Read-Write users while in C2 security mode.	**clear authentication login**
4.	Configure the chosen authentication login method. For more information, see Chapter 49, **Security Configuration** for TACACS+ and Chapter 56, **Authentication Configuration** for RADIUS.	

Using WebView

By default, WebView (Extreme Networks' embedded web server for device configuration and management tasks) is enabled on TCP port number 80 of the S-Series device. You can verify WebView status, enable or disable WebView, and reset the WebView port.

Procedure 9-3 describes how to configure WebView on an S-Series device.

Procedure 9-3 WebView Configuration

Step	Task	Command(s)
1.	Enable WebView	**set webview {enable | disable}**
2.	If necessary, change the TCP port for WebView from the default (port 80).	**set webview port** *port*
3.	Display WebView status to verify your changes.	**show webview**

Management Authentication Notification MIB Overview

You can enable or disable the sending of SNMP notifications when a user login authentication event occurs for various management access types. The types of access currently supported by the MIB include console, telnet, ssh, and web. By default, all Management Authentication Notification types are enabled.

Note: Ensure that SNMP is correctly configured in order to send these notifications. For more information, see Chapter 20, **Simple Network Management Protocol (SNMP) Configuration**.

Configuring Management Authentication Notification MIB

Procedure 9-4 describes how to configure the Management Authentication Notification MIB on an S-Series device. Management Authentication Notification MIB commands can be entered in any command mode.

By default, all Management Authentication Notification types are enabled.

Procedure 9-4 Management Authentication Notification MIB Configuration

Step	Task	Command(s)
1.	Enable or disable the Management Authentication Notification MIB. By selecting the optional Management access type, you can specifically enable or disable a single access type, multiple access types or all of the access types.	**set mgmt-auth-notify** {**enable** | **disable**} [**console**] [**ssh**] [**telnet**] [**web**]
2.	Display the current setting for the Management Authentication Notification MIB.	**show mgmt-auth-notify**
3.	If necessary, set the current setting for the Management Authentication Notification access types to the default setting of enabled.	**clear mgmt-auth-notify**

Management Authentication Notification MIB Configuration Examples

This example shows how to set all the authentication types to be disabled on the Management Authentication Notification MIB. That information is then displayed with the **show** command:

```
S Chassis(su)->set mgmt-auth-notify disable
S Chassis(su)->show mgmt-auth-notify

Management Type  Status
---------------  --------
console          disabled
ssh              disabled
telnet           disabled
web              disabled
```

This example shows how to set only the console and telnet authentication access types to be enabled on the Management Authentication Notification MIB. That information is then displayed with the **show** command:

```
S Chassis(su)->set mgmt-auth-notify enable console telnet
S Chassis(su)->show mgmt-auth-notify

Management Type  Status
---------------  --------
console          enabled
ssh              disabled
telnet           enabled
web              disabled
```

This example displays the state of Management Authentication Notification access types prior to using the **clear** command, then displays the same information after using the **clear** command:

```
S Chassis(su)->show mgmt-auth-notify

Management Type  Status
```

```
---------------  --------
console          enabled
ssh              disabled
telnet           enabled
web              disabled

S Chassis(su)->clear mgmt-auth-notify

S Chassis(su)->show mgmt-auth-notify

Management Type  Status
---------------  --------
console          enabled
ssh              enabled
telnet           enabled
web              enabled
```

License Overview

A license, purchased separately, is available for the following:

- Increased port capacity, to 1024 users per S-Series access module or SSA (S-EOS-PPC license)
- Enhanced routing for S-Series S-130 fabric class (S-EOS-L3-S130 license)
- Advanced routing for S-Series S-150 fabric class (S-EOS-L3-S150 license)

You must activate the purchased license key.

The S-EOS-L3-S130 license is required to run VRF on the S130 class of fabrics or in the S3 chassis with S130 class I/O module installed. In a mixed chassis of S150 and S130 Fabrics, the feature entitlement will revert to the S130 feature set and therefore a license would be required to run VRF in this mixed environment.

The S-EOS-L3-S150 license is not currently available. This license is reserved for future routing enhancements on the S150 class of fabrics.

The license is activated on an S-Series module or chassis, as applicable, by using the **set license** command in any command mode to specify the license type and the ASCII advanced licensing key.

Use the **show license** command in any command mode to display the license key once you have activated the license.

Configuring a License

Procedure 9-5 describes how to configure the license on an S-Series device. License commands can be entered in any command mode.

Procedure 9-5 License Configuration

Step	Task	Command(s)
1.	Activate the license on an S-Series device—module or chassis—as applicable. S-Series license keys can contain white spaces; therefore, you should enclose your license key in double quotation marks.	**set license** {**port-capacity** | **l3-s150** | **l3-s130**}
2.	Display the license key.	**show license**

License Examples

The following example shows how to activate a port capacity license:

```
S Chassis(rw)->set license port-capacity "0001:KS-EOS-PPC:0:12345678:0:Enterprise
Name:0:abcdefgh:abcdefghijklmnopqrstuvwxyz123456" slot 2
```

The following example shows how to display an advanced routing license information:

```
S Chassis(rw)->show license


License Type      Location  Status      Key
----------------  --------  ----------  ----------------------------------------
port-capacity     slot 1    active      0001:S-EOS-PPC:A:BCDEFGHI:0:Enterprise
Name:0:12345678:abcdefghijklmnopqrstuvwxyz123456
port-capacity     slot 2    active      0001:S-EOS-PPC:1:BCDEFGHI:0:Enterprise
Name:0:12345678:abcdefghijklmnopqrstuvwxyz123456
port-capacity     slot 3 active      0001:S-EOS-PPC:0:BCDEFGHI:0:Enterprise
Name:0:12345678:abcdefghijklmnopqrstuvwxyz123456
l3-s130          chassis   active     0001:S-EOS-L3-S130:0:abcdefg:0:Enterprise
Name:0:00000000:abcdefghij+abcdefghijklmnopqrst/abcdefghijklmnopqrstuv
/1234567890abcdefghijklmno/12345==
```

The following example shows how to clear the port capacity license on slot 2:

```
S Chassis(rw)->clear license port-capacity slot 2
```

SNTP Overview

Simple Network Time Protocol (SNTP) provides for the synchronizing of system time for managed devices across a network. The S-Series implementation supports unicast polling and broadcast listening modes of operation to obtain the time from an SNTP server. SNTP is a subset of the Network Time Protocol (NTP) as specified in RFC 1305. The most recent version of SNTP is specified in RFC 2030. Since SNTP is a subset of NTP, all NTP servers are capable of servicing SNTP clients. The SNTP mode is set on the client using the **set sntp client** command.

Unicast Polling Mode

When an SNTP client is operating in unicast mode, SNTP update requests are made directly to a server, configured using the set sntp server command. The client queries these configured SNTP servers at a fixed poll-interval configured using the **set sntp poll-interval** command. The order in which servers are queried is based on a precedence value optionally specified when you configure the server. The lower the configured precedence value, the higher the precedence for that server.

The default is for all servers to have the same precedence. In this case, the server ordering is based upon the indexing of the server table.

The SNTP client makes a request to the SNTP server. The client waits a period of time configured using the **set sntp poll-timeout** command for a response from the server. If the poll timeout timer expires, the client will resend another request, up to the number of retries specified by the **set sntp poll-retry** command. If the retries have been exhausted, the client request is sent to the next server with the lowest configured precedence value or the next server in the server table, if precedence values are the same. If no server responds, the client waits the configured poll-interval time period and the process starts over again.

Broadcast Listening Mode

With SNTP configured for broadcast listening mode, the client is passive and it is the broadcast server that broadcasts the time to the client. Broadcast listening uses the same poll-interval, poll-timeout and poll-retry values as unicast polling but they function differently. To account for the propagation delay between the server and the client, a broadcast delay value in milliseconds is configurable using the **set sntp broadcastdelay** command. The broadcast delay is the time window within which the device can accept a Broadcast SNTP packet from the SNTP server. Once the broadcast delay time window has ended, the poll interval window takes effect where the device will not accept Broadcast SNTP packets. When the poll interval window ends, the broadcast delay window starts again; SNTP packets can once again be accepted. If no Broadcast SNTP packets are seen within that broadcast delay window it is considered a timeout.

SNTP Authentication

SNTP authentication provides the means for the SNTP client to authenticate the SNTP server using symmetric key cryptography. Because SNTP packet data is not sensitive information, the packet itself does not require encryption. Symmetric key cryptography uses a secret password shared between the SNTP client and server to generate an encrypted checksum which is appended to the SNTP packet data. The S-Series SNTP authentication supports 128-bit MD5 symmetric key cryptography.

SNTP authentication is configured by:

- Globally enabling the mode for the SNTP client
- Configuring up to 32 SNTP authentication key instances, by specifying:
  - A numeric key that identifies this SNTP authentication instance
  - The MD5 authentication type
  - A password as either an ASCII string of up to 32 printed characters (no white space) or the Hex formatted cypher produced by the previously entered ASCII string
- Associating an SNTP key instance with the SNTP server
- Enabling the authentication trust flag for the SNTP instance key assigned to the SNTP client

Authentication Mode

SNTP authentication mode must be set to enabled for SNTP authentication to occur between the SNTP client and server. When the mode is set to enable, the SNTP client authenticates with the SNTP server before synchronization occurs. When the mode is set to disable, no authentication is performed on SNTP communications. SNTP authentication is set to disabled by default.

Use the **set sntp authentication mode** command to enable SNTP authentication on the SNTP client.

This example shows how to enable SNTP authentication mode:

```
S Chassis(rw)->set sntp authentication mode enable
```

Authentication Key

The SNTP authentication key specifies the authentication instance to be used by the SNTP client when authenticating with the SNTP server. The SNTP client supports the configuration of up to 32 authentication keys. The authentication key instance ID is a numeric value. Each authentication key instance specifies the authentication type and password. SNTP authentication supports the MD5 authentication algorithm. The password is known to both the SNTP client and server. The password consists of an ASCII string of up to 32 non-white characters or the hexadecimal formatted cypher that was generated from the previously entered ASCII string.

Use the **set sntp authentication key** command to configure an authentication key instance.

This example shows how to create SNTP authentication key instances 1 - 3:

```
S Chassis(rw)->set sntp authentication key 1 md5 foobaraboof
S Chassis(rw)->set sntp authentication key 2 md5 DEADBEAFCAFEBABEDEADBEAFCAFEBAE
S Chassis(rw)->set sntp authentication key 3 md5 01234567890123456789012345678901
```

The SNTP authentication key is associated with an SNTP server using the **set sntp server** command.

This example shows how to set the server at IP address 10.21.1.100 as an SNTP server and to SNTP authenticate using authentication key instance 1:

```
S Chassis(rw)->set sntp server 10.21.1.100 key 1
```

Authentication Trust Flag

The authentication trust flag specifies whether the key associated with it is enabled or disabled. When an authentication key trust flag is enabled, authentication will occur between the client and server the key is assigned to. If an authentication key trust flag is disabled, authentication will not occur between the client and server the key is assigned to.

The authentication trust flag is configured by specifying the instance the trust flag is associated with and whether the trust flag is enabled or disabled.

Use the **set sntp authentication trust** command to configure an SNTP authentication trust flag.

This example shows how to enable trust status for authentication key instance 1 and disable the trust status for authentication key instance 3:

```
S Chassis(rw)->set sntp authentication trust 1 enable
S Chassis(rw)->set sntp authentication trust 3 disable
```

Configuring SNTP

This section provides details for the configuration of SNTP on the S-Series products.

Table 9-5 lists SNTP parameters and their default values.

Table 9-5 Default SNTP Parameters

Parameter	Description	Default Value
SNTP authentication mode	Specifies whether authentication for all SNTP client communications is enabled or disabled.	disabled

Table 9-5 Default SNTP Parameters (continued)

Parameter	Description	Default Value
SNTP authentication trust	Specified whether the trust state of an existing SNTP authentication key is enabled or disabled. Must be enabled for the SNTP authentication to occur.	disabled
SNTP mode	Specifies whether the current SNTP state is broadcast, unicast, or disabled.	disabled
unicast server precedence	Specifies a value that determines the order in which SNTP servers are polled if the precedence values are not the same.	1 (highest precedence)
broadcast delay	Specifies the propagation delay added to the time sent to the client in broadcast listening mode.	3000 milliseconds
poll-interval	Specifies the interval between unicast SNTP requests by the client to the server.	16 seconds
poll-retry	Specifies the number of times the client will resend the SNTP request to the server before moving on to the next server.	1
poll-timeout	Specifies the amount of time a client will wait for a response from the the SNTP server before retrying.	5 seconds
timezone offset	Specifies the offset in hours and minutes from UTC for this device	0 hours, 0 minutes

Procedure 9-6 describes how configure SNTP. SNTP can be configured in any command mode.

Procedure 9-6 Configuring SNTP

Step	Task	Command(s)
1.	Set the SNTP operation mode on the client.	**set sntp client** {**broadcast** | **unicast** | **disable**}
2.	When operating in broadcast mode, optionally change the broadcast delay period in milliseconds to be added to the server time for this client.	**set sntp broadcastdelay** *time*
3.	When operating in unicast mode, set the SNTP server(s) for this client, optionally specifying a precedence value per server.	**set sntp server** *ip-address* [*precedence*][**key** *key-instance*]
4.	When operating in unicast mode, optionally change the poll interval between SNTP unicast requests.	**set sntp poll-interval** *interval*
5.	When operating in unicast mode, optionally change the number of poll retries to a unicast SNTP server.	**set sntp poll-retry** *retry*

Procedure 9-6 Configuring SNTP (continued)

Step	Task	Command(s)
6.	When operating in unicast mode, optionally change the poll timeout for a response to a unicast SNTP request.	**set sntp poll-timeout** *timeout*
7.	Optionally, set the SNTP time zone name and the hours and minutes it is offset from Coordinated Universal Time (UTC). **Note:** The daylight savings time function can be enabled and associated with the timezone set here using the **set summertime** command.	**set timezone** *name* [*hours*] [*minutes*]
8.	Optionally, enable authentication for all SNTP client communications.	**set sntp authentication mode** {**enable** | **disable**}
9.	Optionally, create a new or modify an existing SNTP authentication key.	**set sntp authentication key** *key-instance type password*
10.	Optionally, change the SNTP authentication trust state for an authentication key.	**set sntp authentication trust** *key-instance* {**enable** | **disable**}

Table 9-6 describes how to manage and display SNTP.

Table 9-6 Managing and Displaying SNTP

Task	Command(s)
To display SNTP client settings:	**show sntp**
To set the SNTP client's operational mode to disable:	**clear sntp client**
To remove one or all servers from the SNTP server list:	**clear sntp server** {*ip-address* | **all**}
To reset the delay time for SNTP broadcast frames to its default value:	**clear sntp broadcastdelay**
To reset the poll interval between unicast SNTP requests to its default value:	**clear sntp poll-interval**
To reset the number of poll retries to a unicast SNTP server to its default value:	**clear sntp poll-retry**
To reset the SNTP poll timeout to its default value:	**clear sntp poll-timeout**
To display the current timezone setting:	**show timezone**
To remove the SNTP timezone adjustment values:	**clear timezone**
To clear SNTP authentication key configuration or reset the SNTP authentication mode to the default value:	**clear sntp authentication** {**all** | **key** *key-instance* | **mode**}

SNTP Configuration Examples

The following example configures the client for SNTP broadcast mode:

- Setting the broadcast delay to 3500 milliseconds
- Setting the timezone to Eastern Daylight Time (EDT)
- Displaying the current SNTP configuration

```
S Chassis(rw)->set sntp client broadcast
S Chassis(rw)->set sntp broadcastdelay 3500
```

```
S Chassis(rw)->set timezone EDT -4 0
S Chassis(rw)->show sntp

SNTP Version: 4
Current Time: SAT AUG 01 14:34:53 2009
Timezone: 'EDT', offset from UTC is -4 hours and 0 minutes
Client Mode: broadcast
Broadcast Delay: 3500 microseconds
Broadcast Count: 1
Poll Interval: 512 seconds
Poll Retry: 1
Poll Timeout: 5 seconds
SNTP Poll Requests: 0
Last SNTP Update: SAT AUG 01 14:23:54 2009
Last SNTP Request: SAT AUG 01 14:23:54 2009
Last SNTP Status: Enabled

Status        Precedence        SNTP-Server
------------------------------------------------------------------------
Active             1            10.21.1.300
S Chassis(rw)->
```

The following example configures the client for SNTP unicast mode with SNTP authentication operational:

- Enables SNTP authentication mode
- Creates an SNTP authentication key instance 1 and sets the password to foobar
- Sets the SNTP server to IP address 10.21.1.100 and assigns authentication key instance 1 to it
- Set the SNTP authentication key trust flag to enable for key instance 1
- Sets the SNTP poll interval to 600 seconds
- Sets the UTC timezone to Eastern Daylight Time (EDT)
- Sets the poll retry to 2
- Displays the current SNTP configuration

```
S Chassis(rw)->set sntp client unicast
S Chassis(rw)->set sntp authentication mode enable
S Chassis(rw)->set sntp authentication key 1 md5 foobar
S Chassis(rw)->set sntp authentication trust 1 enable
S Chassis(rw)->set sntp server 10.21.1.100 key 1
S Chassis(rw)->set sntp poll-interval 600
S Chassis(rw)->set timezone EDT -4 0
S Chassis(rw)->set sntp poll-retry 2
S Chassis(rw)->show sntp
SNTP Version: 4
Current Time: FRI MAY 06 15:33:53 2011
Timezone: 'EDT', offset from UTC is -4 hours and 0 minutes
```

```
Client Mode: unicast
Broadcast Delay: 3000 microseconds
Broadcast Count: 0
Poll Interval: 600 seconds
Poll Retry: 2
Poll Timeout: 5 seconds
SNTP Poll Requests: 2
Last SNTP Update: MON MAY 02 14:42:52 2011
Last SNTP Request: MON MAY 02 14:42:52 2011
Last SNTP Status: Enabled

SNTP Servers:

Status        Precedence     Key        SNTP-Server
------------------------------------------------------------------------
Active             1          1         10.21.1.100

SNTP Authentication: Enabled

Status        Key            Type         Trusted
------------------------------------------------------------------------
Active        1             MD5           Enabled
S Chassis(rw)->
```

Telnet Overview

Telnet provides an unsecured communications method between a client and the switch.

Telnet is activated by enabling Telnet on the device, using the **set telnet enable** command in any command mode.

Use the **show telnet** command in any command mode to display whether Telnet is currently enabled or disabled.

Configuring Telnet

Procedure 9-7 describes how to configure and use Telnet on an S-Series device. Telnet commands can be entered in any command mode.

Procedure 9-7 Telnet Configuration

Step	Task	Command(s)
1.	Enable or disable either inbound or outbound or both Telnet services.	**set telnet** {**enable** | **disable**} {**all** | **inbound** | **outbound**}
2.	Verify the Telnet status.	**show telnet**

Procedure 9-7 Telnet Configuration (continued)

Step	Task	Command(s)
3.	Start a Telnet connection. • **-s** - The source IP address to use in the outgoing telnet • **-4** | **-6** - Use only IPv4 or IPv6 addresses but not both • **-vrf** - The name of the router used for this session • **-r** - Bypass the host routing table for this session • *host* - The remote host to Telnet to for this session	**telnet** [**-s** *src-addr*] [**-4** | **-6**] [**-vrf** *router*] [**-r**] {*host* [*port*]}

Telnet Examples

The following example shows how to enable Telnet:

```
S Chassis(rw)->set telnet enable all
```

The following example shows how to verify the Telnet status:

```
S Chassis(rw)->show telnet

Telnet inbound is currently: ENABLED
Telnet outbound is currently: ENABLED
```

The following example telnets to remote host 10.21.42.01:

```
S Chassis(rw)->telnet 10.21.42.01
```

Secure Shell Overview

The Secure Shell (SSH) security feature provides a secure encrypted communications method between a client and the switch providing data privacy and integrity that is an alternative to the unsecure Telnet protocol. Using SSH, the entire session is encrypted, including the transmission of user names and passwords, and negotiated between a client and server both configured with the SSH protocol. Telnet sessions are unsecure. All data is sent unencrypted. Use SSH instead of Telnet when the security of login and data transmission is a concern.

The S-Series SSHv2 implementation includes:

- Data privacy
- Communication integrity

An SSH server resides on the S-Series platform and listens for client connection requests. Once a request is authenticated, a secure connection is formed through which all subsequent traffic is sent. All traffic is encrypted across the secure channel, which ensures data integrity. This prevents someone from seeing clear text passwords or file content, as is possible with the Telnet application.

Once SSH has been enabled and the 7100-Series has at least one valid IP address, you can establish an SSH client session from any TCP/IP based node on the network, by using an application supporting SSH to connect to an IP address and entering your user name and password. Refer to the instructions included with your SSH application for information about establishing a session.

SSH is activated by enabling the SSH server on the device, using the **set ssh enable** command in any command mode.

Enabling the server automatically generates a host key for the server, used during the life of the client to server connection. The host key type can be set to either dsa or rsa. The host key type defaults to rsa.

There is one host key per device; every time an SSH client logs into a device it should see the same host key; if the host key is different, the SSH Client warns you that the host key has changed. The following is a sample warning when an SSH Client detects a new host key:

```
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that the RSA host key has just been changed.
The fingerprint for the RSA key sent by the remote host is
67:c6:71:ff:e8:02:7c:ce:0f:0d:67:67:63:a8:2e:9c.
Please contact your system administrator.
Add correct host key in /home/documentation/doc1/.ssh/known_hosts to get rid of
this message.
Offending key in /home/documentation/doc1/.ssh/known_hosts:24
RSA host key for 10.4.99.4 has changed and you have requested strict checking.
Host key verification failed.
```

The SSH server can be reinitialized. Reinitializing the server clears all current client to server connections. Reinitializing the server does not reinitialize the host key. Should you believe the host key has been compromised, or otherwise wish to change it, the host key can be reinitialized using the **set ssh hostkey reinitialize** command.

An SSH session to a remote host can be started using the **ssh** command.

Use the **show ssh state** command in any command mode to display whether SSH is currently enabled or disabled.

SSH Client Authentication

There are two allowed authentication methods supported for a remote SSH client attempting to login to the S-Series SSH server:

- Password – The SSH client authenticates using a username and password.
- Public key – The SSH client authenticates using public key. This key can either be configured locally (authkey) or provided using an X.509 certificate (PKI).

Password authentication method is enabled by default. The public key authentication method is disabled by default. Allowed authentication methods can be configured using the **set ssh server allowed-auth** command.

Password Authentication

Once SSH has been enabled and the S-Series has at least one valid IP address, you can establish an SSH client session from any TCP/IP based node on the network, by using an application supporting SSH to connect to an IP address and entering your user name and password. Refer to the instructions included with your SSH application for information about establishing a session.

Public Key Authentication

The public key authentication method requires each user to posses a pair of keys, one public and one private. An S-Series device grants access to a specific user by loading the user's public key(s) into a trusted list. Once a public key is configured on a device, any person or device who is in possession of the corresponding private key is "authorized" (authenticated as the owner of the username account).

Authkey

The S-Series supports either the explicit configuration of a single authkey public key or the implicit configuration of public keys using PKI and X.509 Certificates (Refer to Chapter 12, **Public-Key Infrastructure (PKI) Configuration** for PKI configuration details).

The authkey method requires that the public key for each user be explicitly configured on the device using the **set ssh server authkey** command. One key is allowed per user. A drawback of the authkey method is that it is not scalable. Authorization of new users and de-authorization of existing users requires configuration changes on each and every device in the network.

PKI

An alternative and scalable method for obtaining a user's public key is to use Public Key Infrastructure (PKI). With PKI, a user's identity and public key are bound together in an X.509 Certificate. These certificates are digitally signed by a Certificate Authority (CA). A device which trusts a CA implicitly trusts all certificates signed by that CA. This allows the management of users to be moved from the devices to a centralized CA.

The **set ssh server pki trusted-ca-list** command defines the list of CAs which the SSH server will use to verify user certificates. These certificates are provided by the SSH client as part of SSH authentication. This means that once a chain of trusted certificates is configured on the device, any certificate issued by any CA in the chain will also be trusted.

Authorized Certificate List

When a user's certificate is configured on the device that certificate is said to be explicitly trusted. By design, PKI authentication does not require a user's certificate to be configured on the device. However, if desired, you may impose an explicit trust requirement.

Use the **set ssh server pki authorized-cert-list** command to require a user's certificate to be explicitly configured on the device.

If an authorized-cert-list is configured, any certificate presented by a user which is not on this list will be rejected. If the certificate is on the list, then normal PKI authentication will be performed.

If an authorized-cert-list is not configured, then user certificates are only subject to normal PKI verification using the CA certificate trust chain set using the **set ssh server pki trusted-ca-list** command.

The certificate lists specified for both the server PKI trusted and authorized commands are configured using the **set pki certificate** command.

Configuring Secure Shell

Procedure 9-8 describes how to configure Secure Shell on an S-Series device. Secure Shell commands can be entered in any command mode.

Procedure 9-8 SSH Configuration

Step	Task	Command(s)
1.	Enable, disable, or reinitialize the SSH server.	**set ssh** {**enable** | **disable** | **reinitialize**}

Procedure 9-8 SSH Configuration (continued)

Step	Task	Command(s)
2.	Optionally modify the SSH client alive interval.	**set ssh client alive-interval** *interval*
3.	Optionally modify the the maximum number of times a client alive message will be sent before the session times out.	**set ssh client alive-count** *count*
4.	Set or reinitialize the host key on the SSH server.	**set ssh hostkey** [**reinitialize**] [**type** *type*]
5.	Start an SSH session. *hostname* - Specifies the host name or IP address of the remote host this SSH session is connecting to. **-4** | **-6** - Optionally specifies that SSH should use either IPv4 or IPv6 addresses, but not both. **-b** *bind-address* - Optionally specifies the IP address to transmit from when there are multiple interfaces and or addresses. **-c** *cipher-spec* - Optionally specifies a list of the cipher specifications allowed for encrypting this session. **-e** *escape-char* - Optionally sets the escape character for the session. **-l** *login-name* - Optionally specifies the user to login as on the remote host. **-m** *mac-spec* - Optionally specifies the MAC algorithms used for data integrity protection. **-p** *port* - Optionally specifies the host port to connect to on the remote host. **-q** - Optionally specifies that the session will operate in quiet mode, causing all warning and diagnostic messages to be suppressed. **-r** - Optionally specifies that normal routing table lookup should be bypassed and that the session request should be sent directly to a host on an attached network. **-v** - Optionally specifies that the session will operate in verbose mode, causing SSH to print debugging messages about its progress. **-vrf** *router* - Optionally specifies the router on which to source this SSH session.	**ssh** *hostname* [**-4** | **-6**] [**-b** *bind-address*] [**-c** *cipher-spec*] [**-e** *escape-char*] [**-l** *login-name*] [**-m** *mac-spec*] [**-p** *port*] [**-p**] [**-q**] [**-r**] [**-v**] [**-vrf** *router*]
6.	Set the allowed authentication methods when connecting to the SSH server.	**set ssh allowed-auth** {[**password** {**enable** | **disable**}] [**pubkey** {**enable** | **disable**}]}
7.	If the public key authentication method is enabled and you are using the authkey method, explicitly map a public key to each user to be authenticated on the device.	**set ssh server authkey** *username* {**ssh-dss** | **ssh-rsa**} *ssh-key* [**no-confirm**]
8.	If the public key authentication method is enabled and you are using the PKI method, establish the list of trusted CA certificates used during PKI authentication of a user's X.509 certificate.	**set ssh server pki trusted-ca-list** *pki-cert-list*

Procedure 9-8 SSH Configuration (continued)

Step	Task	Command(s)
9.	If the public key authentication method is enabled and you want to require that a user's certificate be explicitly configured on the device, configure the authorized certificate list containing all user certificates required for the device.	**set ssh server pki authorized-cert-list** *pki-cert-list*
10.	Verify the SSH state.	**show ssh state**

Secure Shell Configuration Examples

The following commands enable and verify SSH:

```
S Chassis(rw)->set ssh enable
S Chassis(rw)->show ssh state
SSH Server state: Enabled
S Chassis(rw)->
```

The following command reinitializes the host key on the SSH server:

```
S Chassis(rw)->set ssh hostkey reinitialize
```

Domain Name Server (DNS) Overview

The Domain Name Server (DNS) resolver is a session layer protocol that maps network host names to IP addresses (and vice versa). The client function queries configured servers to provide mapping services for CLI commands (for example, ping, telnet) which allow a hostname to be specified.

The DNS resolver feature is enabled by default. Up to four DNS servers can be configured for DNS resolution. The domain name (Net, Host, Gateway, or Domain name) associated with this device can be configured. A default DNS zone can be specified indicating the initial zone used for DNS lookup. Supported zones are IPv4 and IPv6. The default zone is IPv4. The default zone names are:

- IPv4: - **in-addr.arpa**
- IPv6: - **ip6.int**

The port number the DNS resolver uses for DNS queries can be configured. The default port is **53**. DNS requests will time out and retry the request after a configurable number of seconds. After a configurable amount of retries, if there is more than a single DNS server configure, the request will be sent to the next configured server for up to the number of configured retries.

Configuring DNS

This section provides details for the configuration of DNS resolution on the S-Series products.

Table 9-7 lists DNS parameters and their default values.

Table 9-7 Default DNS Parameters

Parameter	Description	Default Value
DNS resolver state	Specifies whether DNS resolver is enabled or disabled on the device.	enabled

Table 9-7 Default DNS Parameters (continued)

Parameter	Description	Default Value
DNS zone	Specifies the DNS zone for IPv4 and IPv6.	IPv4 - in-addr.arpa IPv6 - ip6.arpa
DNS port	Specifies the port number the DNS resolver uses for DNS queries.	53
timeout	Specifies the number of seconds before a DNS request is retried when the DNS server fails to respond.	10 seconds
query-retries	Specifies the number of times to retry a lookup request to a DNS server that has failed to respond.	2

Procedure 9-9 describes how to configure DNS resolution. DNS can be configured in any CLI command mode.

Procedure 9-9 Configuring DNS Resolution

Step	Task	Command(s)
1.	Enable DNS on the switch if you have manually disabled it. DNS is enabled by default.	**set ip dns enable**
2.	Optionally, set the domain name for this device.	**set ip dns domain** *name*
3.	Configure the DNS servers for this device. Valid server values are: **primary**, **secondary**, **tertiary**, **quaternary**.	**set ip dns server** *ip-address server*
4.	Optionally, configure the DNS zone for IPv4 and IPv6 IP address to name lookups.	**set ip dns zone** {**ipv4** | **ipv6**} *zone-name*
5.	Optionally, configure the port number the DNS resolver uses for DNS queries. The default port is **53**.	**set ip dns port-number** *port-number*
6.	Optionally, change the number of seconds before a DNS request is retried when the DNS server fails to respond.	**set ip dns timeout** *seconds*
7.	Optionally, change the number of times to retry a lookup request to a DNS server that has failed to respond.	**set ip dns query-retries** *retries*

Table 9-8 describes how manage DNS resolution on an S-Series switch. DNS commands can be configured in any CLI command mode.

Table 9-8 Managing DNS Resolution

Task	Command(s)
To clear the DNS domain name configuration.	**clear ip dns domain**
To clear the DNS server configuration.	**clear ip dns server** [*server* | **all**]
To reset the DNS IPv4 or IPv6 zone configuration.	**clear ip dns zone** [**ipv4** | **ipv6**]
To reset the DNS port number used for DNS queries to the default value.	**clear ip dns port-number**
To reset the DNS timeout to the default value.	**clear ip dns timeout**

Table 9-8 Managing DNS Resolution (continued)

Task	Command(s)
To reset the number DNS query retries to the default value.	**clear ip dns query-retries**
To clear all DNS configuration to the default state.	**clear ip dns all**
To reset DNS status for this device to the default value.	**clear dns status**
To display DNS configuration for this device.	**show ip dns**

DNS Configuration Example

The following DNS configuration example:

- Sets the DNS domain name to **Extremenetworks.Documentation**
- Configures two DNS servers:
  - Primary - **123.50.50.10**
  - Secondary - **123.50.50.20**
- Configures the DNS timeout value to **4** seconds
- Configures the number of query retries to **3**

```
S-Series(rw)->set ip dns domain Extremenetworks.Documentation
S-Series(rw)->set ip dns server 153.50.50.10 primary
S-Series(rw)->set ip dns server 153.50.50.20 secondary
S-Series(rw)->set ip dns timeout 4
S-Series(rw)->set ip dns query-retries 3
S-Series(rw)->show ip dns
Current State:             Enabled
Default DNS domain name:   Extremenetworks.Documentation
DNS zones:
  IPv4:                    in-addr.arpa
  IPv6:                    ip6.int

DNS port number:           53
DNS server timeout:        4 seconds
DNS query retries:         3
DNS Name servers                          Status
----------------------------------------  ----------
153.50.50.10                              primary
153.50.50.20                              secondary
S-Series(rw)->
```

DHCP Overview

The Dynamic Host Configuration Protocol (DHCP) provides services for allocating and delivering IPv4 addresses and other IPv4 DHCP server options to Internet hosts. DHCP consists of two components: a protocol for delivering host-specific configuration parameters from a DHCP server to a host, and a mechanism for allocating network addresses to hosts. Optional functionality also

provides services to complete high-availability, authenticated and QoS-dependant host configuration.

The DHCP protocol is based on a client-server model in which a designated DHCP server allocates network addresses and delivers configuration parameters to dynamically configured clients. Throughout the remainder of this section, the term “server” refers to a host providing initialization parameters through DHCP, and the term “client” refers to a host requesting initialization parameters from a DHCP server.

DHCP supports the following mechanisms for IP address allocation:

- Automatic — DHCP assigns an IP address to a client for a limited period of time (or until the client explicitly relinquishes the address).
- Manual — A client's IP address is assigned by the network administrator, and DHCP is used simply to convey the assigned address to the client.

The amount of time that a particular IP address is valid for a system is called a lease. The S-Series device maintains a lease database which contains information about each assigned IP address, the MAC address to which it is assigned, the lease expiration, and whether the address assignment is dynamic or static. The DHCP lease database is stored in flash memory.

Note: The S-Series DHCP server is not designed to work as the primary DHCP server in an enterprise environment with hundreds of clients that are constantly seeking IP address assignment or reassignment. A standalone DHCP server with a redundant backup server may be more suitable for this type of environment.

IPv4 DHCP Supported Server Options

Table 9-9 on page 9-27 lists the IPv4 DHCP server option names and codes supported by the firmware. All options specified in Table 9-9 may be configured using the **option** command. Several commonly-used options may also be configured using dedicated commands: **domain-name**, **dns-server**, **netbios-name-server**, **netbios-node-type**, and **default-router**. These commands are specified in Procedure 9-11 on page 9-36

Except where noted, all options are defined in RFC-2132. In addition, the site-specific option codes designated by RFC-2132 (128-254) may be used to define options for use within a site or an organization. Some vendors have made use of site-specific options to configure their product features.

Table 9-9 IPv4 DHCP Server Codes

Description	Code	Input Methods	RFC
SubnetMask:	1	hex,ip	RFC 2132
TimeOffset:	2	hex	RFC 2132
Router:	3	hex,ip	RFC 2132
TimeServer:	4	hex,ip	RFC 2132
NameServer:	5	hex,ip	RFC 2132
DomainNameServer:	6	hex,ip	RFC 2132
LogServer:	7	hex,ip	RFC 2132
CookieServer:	8	hex,ip	RFC 2132
LPRServer:	9	hex,ip	RFC 2132
ImpressServer:	10	hex,ip	RFC 2132
ResourceLocationServer:	11	hex,ip	RFC 2132

Table 9-9 IPv4 DHCP Server Codes

Description	Code	Input Methods	RFC
HostName:	12	ascii,hex	RFC 2132
BootFileSize:	13	hex	RFC 2132
MeritDumpName:	14	ascii,hex	RFC 2132
DomainName:	15	ascii,hex	RFC 2132
SwapServer:	16	hex,ip	RFC 2132
RootPath:	17	ascii,hex	RFC 2132
ExtensionsPath:	18	ascii,hex	RFC 2132
IpForwarding:	19	hex	RFC 2132
NonLocalSourceRouting:	20	hex	RFC 2132
PolicyFilter:	21	hex,ip	RFC 2132
MaxDatagramReassemblySize:	22	hex	RFC 2132
DefaultIpTTL:	23	hex	RFC 2132
PathMTUAgingTimeout:	24	hex	RFC 2132
PathMTUPlateauTable:	25	hex	RFC 2132
InterfaceMTU:	26	hex	RFC 2132
AllSubnetsLocal:	27	hex	RFC 2132
BroadcastAddress:	28	hex,ip	RFC 2132
PerformMaskDiscovery:	29	hex	RFC 2132
MaskSupplier:	30	hex	RFC 2132
PerformRouterDiscovery:	31	hex	RFC 2132
RouterSolicitationAddress:	32	hex,ip	RFC 2132
StaticRoute:	33	hex,ip	RFC 2132
TrailerEncapsulation:	34	hex	RFC 2132
ARPCacheTimeout:	35	hex	RFC 2132
EthernetEncapsulation:	36	hex	RFC 2132
TCPDefaultTTL:	37	hex	RFC 2132
TCPDefaultKeepaliveInterval:	38	hex	RFC 2132
TCPDefaultKeepaliveGarbage:	39	hex	RFC 2132
NISDomain:	40	ascii,hex	RFC 2132
NISServers:	41	hex,ip	RFC 2132
NTPServers:	42	hex,ip	RFC 2132
VendorSpecificInfo:	43	ascii,hex	RFC 1533 RFC 2132
NetBIOSNameServer:	44	hex,ip	RFC 1533 RFC 2132

Table 9-9 IPv4 DHCP Server Codes

Description	Code	Input Methods	RFC
NetBIOSDatagramDistributionServer:	45	hex,ip	RFC 1533 RFC 2132
NetBIOSNodeType:	46	hex	RFC 1533 RFC 2132
NetBIOSScope:	47	ascii,hex	RFC 1533 RFC 2132
XWindowFontServer:	48	hex,ip	RFC 1533 RFC 2132
XWindowDisplayManager:	49	hex,ip	RFC 1533 RFC 2132
IpAddressLeaseTime:	51	hex	RFC 1533 RFC 2132
RenewalTimeValue:	58	hex	RFC 1533 RFC 2132
RebindingTimeValue:	59	hex	RFC 1533 RFC 2132
NISPlusDomain:	64	ascii,hex	RFC 2132
NISPlusServers:	65	hex,ip	RFC 2132
TFTPServerName:	66	ascii,hex	RFC 2132
BootfileName:	67	ascii,hex	RFC 2132
MobileIpHomeAgent:	68	hex,ip	RFC 2132
SMTPServer:	69	hex,ip	RFC 2132
POP3Server:	70	hex,ip	RFC 2132
NNTPServer:	71	hex,ip	RFC 2132
DefaultWWWServer:	72	hex,ip	RFC 2132
DefaultFingerServer:	73	hex,ip	RFC 2132
DefaultIRCServer:	74	hex,ip	RFC 2132
StreetTalkServer:	75	hex,ip	RFC 2132
STDAServer:	76	hex,ip	RFC 2132
UserClass:	77	ascii,hex	RFC 3004
SLPDirectoryAgents:	78	hex	RFC 2610
SLPServiceScope:	79	hex	RFC 2610

DHCP Server

DHCP provides the following mechanisms for IP address allocation by a DHCP server:

- Automatic—DHCP assigns an IP address, from a range of addresses defined by the **ip local pool** command in configuration mode and configured as a pool of addresses by the **ip dhcp pool** command. The address is assigned to a client for a limited period of time set by the **lease** command (or until the client explicitly relinquishes the address). The **exclude** command is used to exclude one or more IP addresses from a DHCP local address pool.
- Manual—A client's IP address is assigned by the network administrator using the **host** command in DHCP host configuration command mode, and DHCP is used simply to convey the assigned address to the client. Enter DHCP host configuration command mode using the **hardware-address** or **client-identifier** commands in DHCP pool configuration command mode. The **hardware-address** or client-identifier command specifies the client hardware address and client unique identifier, respectively.

The S-Series device maintains a lease database which contains information about each assigned IP address, the MAC address/unique identifier to which it is assigned, the lease expiration, and whether the address assignment is automatic or static.

In addition to assigning IP addresses, the DHCP server can also be configured to assign the following to requesting clients:

- Default router(s), using the **default-router** command in DHCP pool configuration command mode
- DNS server(s), using the **dns-server** command, and domain name, using the **domain-name** command in DHCP pool configuration command mode
- NetBIOS WINS server(s), using the **netbios-name-server** command, and node type, using the **netbios-node-type** command in DHCP pool configuration command mode
- Boot file, using the **bootfile** command mode in DHCP pool configuration command mode
- DHCP options as defined by RFC 2132, using the **option** command in DHCP pool configuration command mode
- Next server in the DHCP server boot process, using **next-server** in the DHCP pool configuration command mode

Configuring Client Class

DHCP client class provides a logical container for a set of client properties, allowing the assignment of a client property set to a DHCP client rather than configuring each client separately. Client-classes are created within a DHCP pool context using the **client-class** command. There are two modes in which a client-class can be assigned, by:

- Directly associating a client-class with a client binding using either the **hardware-address** or **client-identifier** commands
- Receiving a dynamic request with DHCP option 77 (user class) client-class match

DHCP Configuration Example

In the following example client-class **class1** will be configured with a default router 3.3.3.3 and a DNS server 4.4.4.4. When we assign client-class **class1** to client 00:11:22:33:44:55 using the **hardware-address** command within DHCP pool **pool1**, the pool settings for default router (1.1.1.1) will be overwritten by the client-class **class1** settings for this client and any client that should receive a dynamic request with DHCP option 77 specifying client-class **class1**. The DNS server setting will be neither the **pool1** setting nor the **class1** setting. It will be manually set in the host configuration mode for this client to IP address 5.5.5.5. If it were not manually set, it would take the setting specified in **class1**.

pool1 settings also include:

- Domain name of MyCompany.com
- Boot file: dhcpboot
- The assigning of WWW servers 10.70.0.10 10.70.0.11 10.70.0.12 to this pool using option 72 (WWW servers)
- A DHCP boot process next server: 10.70.0.12
- A pool lease of 100 days

These settings will apply to any client configured within pool1 that is not overwritten by either a client class setting or a received option setting.

The example first configures a local pool **pool1** to either automatically or allow the manual setting of IP addresses from the 10.60.0.0 subnet. IP addresses 10.60.0.10 - 30 are excluded from the local **pool1**. These addresses cannot be automatically or manually assigned to clients in this pool. DHCP pool configuration is then entered for **pool1** setting the default router to 1.1.1.1 and the DNS server to 2.2.2.2. When client classes are not applied, these values will be configured along with all the other values listed for this pool.

Client-class class1 is configured as specified above. The client-class class1 is applied to client 00:11:22:33:44:55. Entering host configuration mode for this client, the DNS server is set to IP address 5.5.5.5. This setting will override the class1 DNS server setting for this client. The host IP address for this client is manually set to 10.60.0.1 from the local pool. If the client IP address were not manually set, the client IP address would have been automatically set from the local pool of addresses configured for **pool1**.

```
S Chassis(rw-config)->ip local pool pool1 10.60.1.0 255.255.255.0
S Chassis(rw-config-ip-local-pool)->exclude 10.60.1.10 20
S Chassis(rw-config-ip-local-pool)->exit
S Chassis(rw-config)->ip dhcp pool pool1
S Chassis(rw-config-dhcp-pool)->domain-name MyCompany.com
S Chassis(rw-config-dhcp-pool)->bootfile dhcpboot
S Chassis(rw-config-dhcp-pool)->option 72 ip 10.70.0.10 10.70.0.11 10.70.0.12
S Chassis(rw-config-dhcp-pool)->next-server 10.70.0.12
S Chassis(rw-config-dhcp-pool)->lease 100
S Chassis(rw-config-dhcp-pool)->default-router 1.1.1.1
S Chassis(rw-config-dhcp-pool)->dns-server 2.2.2.2
S Chassis(rw-config-dhcp-pool)->client-class class1
S Chassis(rw-config-dhcp-class)->default-router 3.3.3.3
S Chassis(rw-config-dhcp-class)->dns-server 4.4.4.4
S Chassis(rw-config-dhcp-class)->exit
S Chassis(rw-config-dhcp-pool)->hardware-address 00:11:22:33:44:55 client-class
class1
S Chassis(rw-config-dhcp-host)->dns-server 5.5.5.5
S Chassis(rw-config-dhcp-host)->host 10.60.0.1
S Chassis(rw-config-dhcp-host)->exit
S Chassis(rw-config-dhcp-pool)->exit
S Chassis(rw-config)->
```

DHCPv6 Overview

The IPv6 Dynamic Host Configuration Protocol (DHCPv6) provides services for delivering DHCPv6 server options to requesting clients. DHCPv6 options are contained in a DHCPv6 pool. The pool is assigned to the DHCPv6 server to which the client requests server options information.

Use the **ipv6 dhcp pool** command in global router configuration mode to create the DHCPv6 server pool and enter pool configuration mode.

Use the **ipv6 dhcp server** command in interface configuration mode to assign the pool to the DHCPv6 server. The assigned pool must already be created before you assign the pool to a DHCPv6 server.

Use this **domain-name** command to return one or more domain names when responding to a DHCPv6 client request.

Use the **dns-server** command to assign one or more DNS servers to DHCPv6 clients.

Use the **nis-domain-name** command to return one or more Network Information Services (NIS) domain names when responding to a DHCPv6 client request.

Use the **nis-server** command to assign one or more Network Information Services (NIS) servers to DHCPv6 clients.

Use the **nisp-domain-name** command to return one or more Network Information Services (NIS) version 2 domain names when responding to a DHCPv6 client request.

Use the **nisp-server** command to assign one or more Network Information Services (NIS) version 2 servers to DHCPv6 clients.

Use the **sip-domain-name** command to return one or more Session Initiation Protocol (SIP) domain names when responding to a DHCPv6 client request.

Use the **sip-server** command to assign one or more Session Initiation Protocol (SIP) servers to DHCPv6 clients.

Use the **sntp-server** command to assign a Simple Network Time Protocol (SNTP) server to DHCPv6 clients.

Use the **unicast-server** command to assign a unicast server to DHCPv6 clients.

Use the **information-refresh** command to configure the amount of time a client should wait before refreshing information from the DHCPv6 server.

Table 9-10 DHCPv6 Server Supported Options

DHCPv6 Option	Option Code	RFC
Unicast Server	**12**	**RFC 3315**
SIP Domain Name	**21**	**RFC 3319**
SIP Server	**22**	**RFC 3319**
DNS Server	**23**	**RFC 3646**
Domain Search List	**24**	**RFC 3646**
NIS Server	**27**	**RFC 3898**
NISP server	**28**	**RFC 3898**
NIS Domain Name	**29**	**RFC 3898**
NISP Domain Name	**30**	**RFC 3898**
SNTP Server	**31**	**RFC 4075**

Table 9-10 DHCPv6 Server Supported Options

DHCPv6 Option	Option Code	RFC
Information Refresh Time	**32**	**RFC 4242**

DHCPv6 Server Option Information Configuration Example

This example enables the DHCPv6 server on VLAN 1 with the DHCPv6 option information pool docPool by:

- Creating the docPool DHCPv6 option information pool and entering DHCPv6 option configuration mode and assigning the following options to the pool:
  - The **myEnterprise.com** domain name
  - An IPv6 DNS server at addresses **1111::12**, **1111::13**, and **1111::14**
  - The **myNisEnterprise.com** NIS domain name
  - An IPv6 NIS-DNS server at addresses **1111::12**, **1111::13**, and **1111::14**
  - The **myNispEnterprise.com** NISP domain name
  - An IPv6 NISP-DNS server at addresses **1111::12**, **1111::13**, and **1111::14**
  - The **mySipEnterprise.com** SIP domain name
  - An IPv6 SIP-DNS server at addresses **1111::12**, **1111::13**, and **1111::14**
  - An SNTP server at address **1111::15**
  - A unicast server at address **1111::15**
  - A client information refresh wait of **12** hours
- Configuring the DHCP server on VLAN **44** with a **dhcpPool1** DHCPv6 pool

```
S Chassis(rw)->configure
S Chassis(rw-config)->ipv6 dhcp pool docPool
S Chassis(rw-config-dhcp-v6-pool)->domain-name myEnterprise.com
S Chassis(rw-config-dhcp-v6-pool)->dns-server 1111::12 1111::13 1111::14
S Chassis(rw-config-dhcp-v6-pool)->nis-domain-name myNisEnterprise.com
S Chassis(rw-config-dhcp-v6-pool)->nis-dns-server 1111::12 1111::13 1111::14
S Chassis(rw-config-dhcp-v6-pool)->nisp-domain-name myNispEnterprise.com
S Chassis(rw-config-dhcp-v6-pool)->nis-dns-server 1111::12 1111::13 1111::14
S Chassis(rw-config-dhcp-v6-pool)->sip-domain-name mySipEnterprise.com
S Chassis(rw-config-dhcp-v6-pool)->sip-dns-server 1111::12 1111::13 1111::14
S Chassis(rw-config-dhcp-v6-pool)->sntp-server 1111::15
S Chassis(rw-config-dhcp-v6-pool)->unicast-server 1111::15
S Chassis(rw-config-dhcp-v6-pool)->information-refresh 0 12 0
S Chassis(rw-config-dhcp-v6-pool)->exit
S Chassis(rw-config)->interface vlan 44
S Chassis(rw-config-intf-vlan.0.44)->ipv6 address 9999::1/64
S Chassis(rw-config-intf-vlan.0.44)->ipv6 dhcp server dhcpPool1
S Chassis(rw-config-intf-vlan.0.44)->ipv6 forwarding
S Chassis(rw-config-intf-vlan.0.44)->no shutdown
S Chassis(rw-config-intf-vlan.0.44)->exit
```

```
S Chassis(rw-config)->
```

IPv6 DHCP Relay Source and Destination Interfaces

By default the router interface that receives the DHCP request is the DHCP request source interface for the router. You can configure an always-up global source interface for the device such as a loop-back interface. The configured global source-interface can be overridden at the interface level.

Use the **ipv6 dhcp relay source-interface** command in global configuration mode to configure a global source interface for the router.

Use the **ipv6 dhcp relay source-interface** command in interface configuration mode to have the specified interface override the globally configured source interface for this interface.

The destination server interface must be specified when the DHCPv6 destination server address is either link-local or multicast IPv6. Specifying a destination interface is not required if the DHCPv6 destination server address is a global address.

The DHCP Solicit message is a multicast message to the all DHCP server address (ff02::1:2). The all DHCP server address only crosses network segments when explicitly routed. If your network has multiple segments, you must configure a DHCP relay agent on the router interface for each segment, so that all DHCP solicit messages can be forwarded to your DHCP server.

If a destination interface is not specified, because the DHCPv6 server address is a global address, the interface is determined by a standard routing table lookup.

Use the **ipv6 dhcp relay destination** command in interface configuration mode to configure the IPv6 DHCP relay agent to forward an IPv6 DHCP request from a client or other relay agent to the destination server or next relay agent address.

Configuring DHCP

This section provides details for the configuration of DHCP on the S-Series products.

Table 9-11 lists DHCP parameters and their default values.

Table 9-11 Default DHCP Parameters

Parameter	Description	Default Value
DHCP interface state	Specifies whether DHCP is enabled or disabled on a routing interface.	disabled
number of ping packets	Specifies the number of packets a DHCP server sends to an IP address before assigning the address to a requesting client.	2
ping timeout	Specifies the amount of time the DHCP server will wait for a ping reply from an IP address before timing out.	500 milliseconds
information refresh	Specifies the amount of time an DHCPv6 client will wait before requesting an option information refresh from the DHCPv6 server.	1 day

Procedure 9-10 describes enabling the DHCP feature and client configuration.

Procedure 9-10 Enabling the DHCP Server and Configuring Automatic Address Assignment

Step	Task	Command(s)
1.	Enable DHCP on the routing interface in interface configuration command mode. DHCP is enabled by default.	**ip dhcp server**
2.	Configure the local address pool to be used as a DHCP subnet for automatic IP address assignment.	**ip local pool** *name subnet mask*
3.	Optionally, in local address pool configuration mode, exclude a range of IP addresses from the configured local pool subnet, specifying the beginning IP address and the number of additional addresses to exclude.	**exclude** *ip-address number*
4.	Enter DHCP address pool configuration command mode for the specified pool.	**ip dhcp pool** *name*
5.	Specify, in DHCP pool or client-class mode, the lease duration for an IP address dynamically assigned by a DHCP server to a client.	**lease** {*days* [*hours*] [*minutes*]}
6.	In DHCP pool configuration mode, enable DHCP host configuration mode and optionally associate a client class with a DHCP client.	**client-identifier** *unique-identifier* [*client-class name*]
7.	In DHCP pool configuration mode, specify parameters for a new DHCP client address.	**hardware-address** *hardware-address* [*type*]
8.	Specify, in configuration command mode, the number of packets a DHCP Server sends to a pool address as part of a ping operation.	**ip dhcp ping packets** *number*
9.	Specify, in configuration command mode, the number of milliseconds the DHCP server will wait for a ping reply from an IP address before timing out.	**ip dhcp ping timeout** *milliseconds*
10.	In either configuration command mode or interface configuration mode, specify an always up source interface of IPv6 DHCP relay forwarded messages.	**ipv6 dhcp relay source-interface** *interface*
11.	In interface command mode, configure the IPv6 DHCP relay agent to forward an IPv6 DHCP request from a client or other relay agent to the destination server or relay agent address.	**ipv6 dhcp relay destination** *ipv6-address* [*destination-interface*] [global] [**vrf** *vrf*]

Table 9-12 describes how to configure the router.

Table 9-12 Configuring Static IP Address Assignment

Task	Command(s)
Optionally, configure static IP address assignment in DHCP host configuration command mode by specifying an host IP address and network mask for a static DHCP binding. Use either the **hardware-address** or **client-identifier** command in DHCP pool configuration command mode to enter host configuration command mode.	**host address** [*mask* | *prefix-length*]

Procedure 9-11 describes DHCP client configuration.

Procedure 9-11 DHCP Client Configuration

Step	Task	Command(s)
1.	Optionally, in DHCP host or pool configuration command mode, specify a domain name for the DHCP client.	**domain-name** *name*
2.	Specify, in DHCP host or pool configuration command mode, one or more DNS server IP addresses to the DHCP clients.	**dns-server** *address* [*address2...address8*]
3.	Specify, in DHCP host or pool configuration command mode, one or more NetBIOS WINS servers to the DHCP clients.	**netbios-name-server** *address* [*address2...address8*]
4.	Specify, in DHCP host or pool configuration command mode, one or more node types to the DHCP clients. • **h-node** — hybrid (recommended) • **b-node** — broadcast • **p-node** — peer-to-peer • **m-mode** — mixed	**netbios-node-type** *type*
5.	Optionally, in DHCP host or pool configuration command mode, assign routers to a DHCP client's default router list.	**default-router** *address* [*address2...address8*]
6.	Specify, in DHCP host or pool configuration command mode, the default boot image for the DHCP client.	**bootfile** *filename*
7.	Optionally, in DHCP host or pool configuration command mode, specify the next server in the DHCP server boot process.	**next-server** *ip-address*
8.	Optionally, in DHCP host or pool configuration command mode, configure DHCP options.	**option code** [**instance** *number*] {**ascii** *string* | **hex** *string* | **ip** *address*}
9.	Optionally, in client configuration command mode, assign a name to a DHCP client. Optionally, assign the named client to a client class.	**client-name** *name* [**client-class** *name*]
10.	Optionally, in DHCP host or pool configuration command mode, configure a client class.	**client-class** *name*

Procedure 9-12 describes enabling the DHCP feature and client configuration.

Procedure 9-12 Configuring DHCPv6 Information Option Pools

Step	Task	Command(s)
1.	In global configuration mode, create a DHCPv6 option pool and enter DHCPv6 pool configuration mode.	**ipv6 dhcp pool** *poolname*
2.	In interface configuration mode, assign a DHCPv6 information option pool to a DHCPv6 server.	**ipv6 dhcp server** *poolname*

Procedure 9-13 describes DHCPv6 client configuration. All commands are configured in DHCPv6 pool configuration command mode.

Procedure 9-13 DHCPv6 Client Configuration

Step	Task	Command(s)
1.	Specify a domain name for the DHCPv6 client.	**domain-name** *name*
2.	Specify one or more DNS server IP addresses to the DHCP clients.	**dns-server** *address* [*address2...address8*]
3.	Specify one or more Network Information Services (NIS) domain names to return when responding to a DHCPv6 client request.	**nis-domain-name** *domain* [*domain2 ... domain8*]
4.	Specify one or more Network Information Services (NIS) servers to assign to DHCPv6 clients.	**nis-server** *address* [*address2...address8*]
5.	Specify one or more Network Information Services (NIS) version 2 domain names to return when responding to a DHCPv6 client request.	**nisp-domain-name** *domain* [*domain2 ... domain8*]
6.	Specify one or more Network Information Services (NIS) version 2 servers to assign to DHCPv6 clients.	**nis-server** *address* [*address2...address8*]
7.	Specify one or more Session Initiation Protocol (SIP) domain names to return when responding to a DHCPv6 client request.	**sip-domain-name** *domain* [*domain2 ... domain8*]
8.	Specify one or more Session Initiation Protocol (SIP) servers to assign to DHCPv6 clients.	**sip-server** *address* [*address2...address8*]
9.	Specify a Simple Network Time Protocol (SNTP) server to assign to DHCPv6 clients.	**sntp-server** *address*
10.	Specify a unicast server to assign to DHCPv6 clients.	**unicast-server** *address*
11.	Specify the amount of time a client should wait before refreshing information from the DHCPv6 server.	**information-refresh** {**infinite** | *days* [[*hours*] [*minutes*]}

Table 9-13 describes how to manage and display DHCP.

Table 9-13 Managing and Displaying DHCP

Task	Command(s)
To display IP DHCP bindings, in any command mode enter:	**show ip dhcp binding** [*ip-address*]
To display DHCP server statistics, in any command mode enter:	**show ip dhcp server statistics**
To delete one or all automatic DHCP address bindings, in configuration command mode enter:	**clear ip dhcp binding** {*address* | ***}**
To clear ip dhcp server statistics, in configuration command mode enter:	**clear ip dhcp server statistics**

Node Alias Overview

Node alias provides for the defining of objects which can be used for the discovery of end systems on a per port basis. Because the S-Series firmware sees all packets that transit a port as members of a flow, node alias uses that flow defining capability to map key system objects such as VLAN ID, Source IP address, MAC address, host name, and protocol that define the end-users transiting the node alias enabled port. Enabling all ports for node alias allows for the building of a network wide cross-reference of key user elements providing the network administrator with a powerful troubleshooting tool.

Node alias creates an entry for each unique set of elements discovered when investigating the packets that transit the node alias enabled port. Node alias entries can be configured for all protocols or per protocol.

Configuring Node Alias

This section describes how to configure Node Alias on the S-Series products.

Procedure 9-14 describes how to configure node alias on switch ports.

Procedure 9-14 Configuring Node Alias

Step	Task	Command(s)
1.	Optionally disable node alias on switch ports. All ports and LAGs are enabled by default.	**set nodealias disable** [**protocols** *protocols*] port-string
2.	Optionally change the maximum number of entries allowed for the specified switch port.	**set nodealias maxentries** port-string

Table 9-14 describes how to display and manage the node alias on the S-Series device.

Table 9-14 Managing Node Alias

Task	Command(s)
To display the current port node alias state and maximum entries settings.	**show nodealias config** [*port-string*]
To display node alias entries for all or the specified port(s).	**show nodealias** [*port-string*]
To display node alias entries for the specified MAC address, optionally narrowing the search by protocol and port. The MAC address can be specified as a partial MAC address.	**show nodealias mac** *mac_address* [*protocol*] [*port-string*]

Table 9-14 Managing Node Alias (continued)

Task	Command(s)
To display node alias entries for the specified protocol, optionally narrowing the search by port. In the case of the IP protocol, an IP address in full or partial form can be specified.	**show nodealias protocol** {*protocol*} [**ip_address** *ip-address*] [port-string]
To clear a specified node alias entry or all entries for the specified port(s).	**clear nodealias** {**port** *port-string* | **alias-id** *alias-id* | [**protocols** *protocols*]}
To reset node alias state to enabled and clear the maximum entries value for the specified port(s).	**clear nodealias config** *port-string*

Setting Node Alias State and Max Entries

Node alias state and maximum entries settings are set using the **set nodealias** command in any command mode. Use the **show nodealias config** command to display the current nodealias state and maximum entries setting for this device.

The following example enables node alias on port ge.1.1, sets the maximum entries for ge.1.1 to 100, and displays all entries using the VRRP protocol:

```
S-Series(rw)->set nodealias enable ge.1.1
S-Series(rw)->set nodealias maxentries 100
S-Series(rw)->show nodealias protocol vrrp ge.1.1

Port: ge.1.1   Time:   2009-07-24 16:20:37
-------------------------------------------------------
Alias ID         = 194020          Active           = true
Vlan ID          = 1               MAC Address      = 00-00-5e-00-01-01
Protocol         = vrrp            Rtr ID           = 0x01
Rtr priority     = 0xff
```

The following example displays all entries on port ge.1.1 with a MAC address beginning with 00-90:

```
S-Series(rw)->show nodealias mac 00-90 ge.1.1

Port: ge.1.1   Time:   2009-07-24 16:28:47
-------------------------------------------------------
Alias ID         = 194067          Active           = true
Vlan ID          = 1               MAC Address      = 00-90-27-17-13-e7
Protocol         = ip              Source IP        = 10.21.2.95
```

The following example displays all entries on port ge.1.1 with an IP subnet of 10.21.*.*

```
S-Series(rw)->show nodealias protocol ip ip_address 10.21 ge.1.1

Port: ge.1.1   Time:   2009-07-25 08:12:33
-------------------------------------------------------
Alias ID         = 194426          Active           = true
Vlan ID          = 1               MAC Address      = 00-00-5e-00-01-01
Protocol         = ip              Source IP        = 10.21.64.1
```

```
.
.
.

Port: ge.1.1   Time:   2009-07-25 08:25:15
-------------------------------------------------------
Alias ID         = 194460          Active           = true
Vlan ID          = 1               MAC Address      = 00-01-f4-5b-5f-a7
Protocol         = ip              Source IP        = 10.21.64.1


Port: ge.1.1   Time:   2009-07-25 08:14:45
-------------------------------------------------------
Alias ID         = 194435          Active           = true
Vlan ID          = 1               MAC Address      = 00-e0-63-86-2b-bf
Protocol         = ip              Source IP        = 10.21.64.2
```

MAC Address Settings Overview

MAC address settings configuration provides for the ability to:

- Configure a timeout period for aging learned MAC addresses
- Limit specified layer two multicast addresses to specific ports within a VLAN
- Statically enter unicast MAC addresses into the filtering database (FID). Static MAC addresses can be permanent or ageable
- Enable the ability to treat static unicast MAC addresses as a multicast address

Age Time

Both learned and statically configured MAC addresses can be assigned an age in seconds after which they will be flushed from the FID. The default value is 300 seconds.

Use the **set mac agetime** command in any command mode to configure the MAC age-time for MAC addresses on this device.

The following example sets the age-time for MAC addresses on this device to 600 seconds:

```
S Chassis(rw)->set mac agetime 600
S Chassis(rw)->show mac agetime
Aging time: 600 seconds
S Chassis(rw)->
```

Multicast MAC Address VLAN Port Limit

Specified layer two multicast MAC addresses can be limited to specific ports within a VLAN. You can append or clear ports from the list of ports the multicast MAC address is dynamically learned on or flooded to.

Use the **set mac multicast** command in any command mode to limit the specified multicast MAC address to specific ports within a VLAN. This command creates a static MAC address that forces

frames with a specific multicast destination address to be hardware switched on a the specified VLAN. This command can also be used to flood a unicast MAC address. The command takes a list of ports. The specified port list scopes the flooding to a port set smaller than the egress port set for the VLAN. Multicast frames matching a static MAC address entry are transmitted on each of the specified ports (or all ports if no port is specified) that egresses the VLAN.

Note: A key purpose of the static multicast MAC address feature is to support the multicast version of various NIC-based proprietary load balancing technologies, including Network Load Balancing (NLB). See "Network Load Balanced (NLB) Servers Configured for Multicast" on page 9-41.

The following example specifies that multicast MAC address 00:a4:01:ff:0e:00:01 be limited to port ge.1.1 on VLAN 100:

```
S Chassis(rw)->set mac multicast 00:a4:01:ff:0e:01 100 ge.1.1
Warning: Unicast address converted to multicast 01-A4-01-FF-0E-01
```

Unicast MAC addresses can be statically entered into a FID for a single port. This entry can be configured as either permanent or ageable. If ageable, it will age out the same as a dynamically learned MAC address.

Network Load Balanced (NLB) Servers Configured for Multicast

Network load balancer or similar proprietary load balancing technologies, comprised of multiple physical machines responding to a single "virtual" IP address, expect the switch to flood its traffic to all ports on the destination VLAN. The flooded traffic uses the Extreme Networks device soft forwarding path, subject to it's rate limiters, instead of the device hardware forwarding path. This traffic will also compete for the slow path resources and the first packets from other new flows.

To force the virtual server packets to take a hardware switch path, configure a MAC address static entry in the Filter Database (FDB). If the destination MAC is multicast (the Group bit is set), use the **set mac multicast** command, optionally specifying a port-list that further scopes the flooding, to force the forwarding traffic to use the hardware path (see "Multicast MAC Address VLAN Port Limit" on page 9-40). The set mac multicast command is only supported on frames that ingress and egress on the same VLAN (switched frames).

Static MAC Address Entry

Use the **set mac unicast** command in any command mode to statically enter a unicast MAC address into a FID for a single port.

The following example statically enters unicast MAC address 00:a4:01:ff:0e:01 into FID 1 for port ge.1.1 and sets the MAC address to ageable:

```
S Chassis(rw)->set mac unicast 00:a4:01:ff:0e:01 1 ge.1.1 ageable
S Chassis(rw)->show mac fid 1

MAC Address       FID  Port          Type    Status
----------------- ---- ------------- ------- -------
00-00-5E-00-01-01 1    ge.1.1        learned
00-16-41-A8-8F-D8 1    ge.1.1        learned
00-A0-C9-0A-8F-52 1    ge.1.1        learned
00-A4-01-FF-0E-01 1    ge.1.1        mgmt    ageable
00-B0-D0-B7-D2-C5 1    ge.1.1        learned
S Chassis(rw)->
```

Unicast as Multicast

The unicast as multicast feature causes unicast searches in the filter data base to match on statically configured multicast entries using hardware forwarding. The unicast as multicast feature is used when a data stream originates from or is forwarded to a unicast address that then forwards it to multiple hosts, such as when using Network Load Balancing (NLB). When unicast as multicast is enabled on the device, a lookup is performed to determine if the unicast address has also been configured for multicast on the device. If a multicast address is found, packets are hardware forwarded out the configured VLAN and port(s) as defined in the static multicast configuration by extending the search phase of the Layer 2 lookup to match an unlearned destination MAC address against static multicast MAC entries. The unicast as multicast feature is configured by:

1. Using the **set mac multicast** command, in any command mode, to specify the MAC address to be treated as a multicast address, specifying the VLAN and egress port(s) to use
2. Using the **set mac unicast-as-multicast** command, in any command mode, to enable static unicast MAC addresses to be treated as multicast addresses on this device

The following command enables the unicast as multicast feature on this device:

```
S Chassis(rw)->set mac unicast-as-multicast enable
S Chassis(rw)->show mac unicast-as-multicast
Unicast as multicast: enabled
S Chassis(rw)->
```

New and Moved MAC Address Detection

You can configure this device such that SNMP trap messaging is enabled globally or per port to send notifications, when a new MAC address is first detected, or a preexisting MAC address is moved.

Use the **set newaddrtrap** command in any command mode to enable SNMP trap messaging to report the detection of a new MAC address for the affected ports. Enabling SNMP trap messaging to report the detection of a new MAC address must be enabled globally and enabled on the affected ports as two separate CLI entries. The new MAC address trap feature is disabled by default.

The following example configures SNMP trap messaging to send a notification when a new MAC address is detected on port ge.1.1:

```
S Chassis(rw)->set newaddrtrap enable
S Chassis(rw)->set newaddrtrap ge.1.1 enable
```

Use the **set movedaddrtrap** command in any command mode to enable SNMP trap messaging to report detection of a moved MAC address for the affected ports. Enabling SNMP trap messaging to report detection of a moved MAC address must be enabled globally and on the affected ports as two separate CLI entries. The moved MAC address trap feature is disabled by default.

The following example configures SNMP trap messaging to send a notification when a moved MAC address is detected on port ge.1.1:

```
S Chassis(rw)->set movedaddrtrap enable
S Chassis(rw)->set movedaddrtrap ge.1.1 enable
```

Procedure 9-15 describes how to configure MAC address settings. All commands for this feature can be set in any command mode.

Procedure 9-15 Configuring MAC Address Settings

Step	Task	Command(s)
1.	Optionally, change the age time for MAC addresses FID entries for this device.	**set mac agetime** *time*
2.	Optionally, limit a multicast MAC address to a specific port within a VLAN.	**set mac multicast** *mac-address vlan-id* [*port-string*] {**append** | **clear**}
3.	Optionally, enter a static unicast MAC address into the FID.	**set mac unicast** *mac-address fid receive-port* [**ageable**]
4.	Optionally, enable unicast MAC addresses to be treated as multicast MAC addresses on this device.	**set mac unicast-as-multicast** {**enable** | **disable**}
5.	Optionally, set the maximum number of MAC entries allowed on the device.	**set mac max-entries** {**64K** | **128K**}
6.	Optionally, enable SNMP trap messaging to report the detection of new MAC addresses for the specified port by first globally enabling the feature, followed by enabling the affected ports.	**set newaddrtrap** {**enable** | **disable**} (Global) **set newaddrtrap** *port-string* {**enable** | **disable**} (Port)
7.	Optionally, enable SNMP trap messaging to report the detection of a moved MAC address for the specified port by first globally enabling the feature, followed by enabling the affected ports.	**set movedaddrtrap** {**enable** | **disable**} (Global) **set movedaddrtrap** *port-string* {**enable** | **disable**} (Port)

Terms and Definitions

Table 9-15 lists terms and definitions used in this system configuration discussion.

Table 9-15 System Configuration Terms and Definitions

Term	Definition
age time	The amount of time a non-permanent MAC address will stay in the FIB before becoming marked as invalid.
automatic address assignment	DHCP automatically assigns an IP address from a range of configured addresses to a client for a limited period of time
broadcast listening	An SNTP operational mode for which the SNTP server broadcasts the time adding a configured propagation delay value to compensate for the travel time of the packet from the SNTP server to the SNTP client.
Domain Name Server (DNS) resolver	A session layer protocol that maps network host names to IP addresses and vice versa.
Dynamic Host Configuration Protocol (DHCP)	A network layer protocol that implements automatic or manual assignment of IP addresses and other configuration information to client devices by servers.
entry	A grouping of key packet objects reported by node alias that define a single flow for this port.
FID	The filtering database that contains the MAC addresses for this device.
manual address assignment	The client's IP address is assigned by the network administrator, DHCP is used only to convey the assigned address to the client.

Table 9-15 System Configuration Terms and Definitions (continued)

Term	Definition
node alias	An S-Series feature that analyzes flows transiting a port for key packet objects that can be used as a cross-reference that port's end users.
poll-interval	The time between SNTP update requests by the client to the server in unicast operations mode.
poll-timeout	The time a unicast SNTP client waits before sending another update request to the SNTP server.
precedence	A value used to determine the order in which SNTP servers will be polled in unicast operational mode.
Secure Shell (SSH)	security feature provides a secure encrypted communications method between a client and the switch to the entire session, providing data privacy and integrity that is an alternative to the unsecure Telnet protocol.
Simple Network Time Protocol (SNTP)	A protocol that provides for the synchronizing of system time for managed devices across a network.
unicast as multicast	A feature that treats a unicast MAC address as if it were a multicast MAC address by extending the search phase of layer 2 lookup to match the unlearned destination MAC address against the static Multicast MAC entries on this device.
unicast polling	An SNTP operational mode for which the client directly requests updates from the SNTP server.

10

Security Mode Configuration

This chapter provides information about configuring and monitoring security modes on S-Series devices.

How to Use Security Mode in Your Network

There are three aspects to setting the security mode on your device:

- FIPS security which determines the authentication and encryption algorithms supported on your system
- Security profile which provides for either normal (standard) operation or the setting of the government C2 security rating which, for a subset of security related commands, sets:
  - Non-standard command parameter default values
  - Non-standard command parameter range values
  - Access to the command
- Boot menu access which enables or disables access to the boot menu

Note: Super-user administrative privilege is required to access security mode configuration for FIPS, security profile, and boot menu access.

FIPS Security Mode

When enabled, FIPS security mode puts the switch into Federal Information Processing Standards (FIPS) mode. FIPS security mode is a mode where only FIPS approved authentication and encryption algorithms and methods are used. FIPS security mode defaults to disabled. FIPS security mode must be manually enabled using the **set security fips mode** command. The **show security fips mode** command displays the current FIPS security mode state for the device.

If FIPS security mode is enabled, only the SHA1 authentication algorithm is supported.

Note: Changing the FIPS security mode of the switch requires a system reset.

Security Profile Mode

C2 security mode is a security rating established by the U.S. National Computer Security Center (NCSC) and specifies that a product passes the Department of Defense (DoD) Trusted Computer System Evaluation Criteria (TCSEC) tests. A product that meets the C2 security mode rating provides at least the minimum allowable levels of confidence demanded for government agencies and offices and other organizations that process classified or secure information.

Security profile mode determines access to some commands based upon user mode and the sensitivity of the command to security considerations. Security profile can also affect parameter range and default values for some commands.

Security profile mode is set using the **set security profile** command and can be set to **C2** or **normal**. The **show security profile** command displays the current security profile setting for the device.

Note: Changing the security profile mode of the switch requires a system reset.

Boot Access Security Mode

On the S-Series device, the boot menu is called the System Image Loader menu. This menu provides for such boot related functionality as the ability to:

- Delete an image file
- Set the boot file image
- Start a ZMODEM download
- Display available image files or current boot image file
- Clear the persistent storage

By default you can gain access to this menu as the device is booting by pressing any key once you see the line:

```
Press any key to enter System Image Loader menu
```

Pressing any key places you at the System Image Loader prompt:

```
[System Image Loader]:
```

See "Setting the Boot Firmware Image" on page 3-8 for additional boot menu information.

Access to the boot menu can be enabled or disabled using the **set security boot-access** command. The **show security boot-access** command displays the current boot menu access setting for the device.

Disabling access to the boot menu affects all user privilege modes, including super-user.

Security Profile Mode Default Parameter Setting Changes

Some command parameter default settings change when changing the security profile mode. Table 10-1 details command parameter default setting changes per security profile mode.

Table 10-1 Security Profile Mode Command Parameter Default Setting Changes

Description	Command	Normal Default	C2 Default
Sets the time an idle console, SSH or Telnet CLI session will remain connected before being logged out. See "Using the CLI" on page 2-1 for additional configuration information.	**set logout**	10 minutes	15 minutes
Sets the number of minutes to lockout the default admin super-user account after maximum login attempts. See "User Management Overview" on page 9-7 for additional configuration information.	**set system lockout time**	0 minutes	8 minutes
Sets the minimum interval in minutes between password changes allowed for non-super-users. See "User Management Overview" on page 9-7 for additional configuration information.	**set system password change-frequency**	0 never	1 day
Sets the number of inactive days before a non-super-user account is locked out. See "User Management Overview" on page 9-7 for additional configuration information.	**set system lockout inactive**	0 never	90 days
Sets a grace period in either the number of logins or days before the password is locked out. See "User Management Overview" on page 9-7 for additional configuration information.	**set system password grace-period logins**	0 off	3 logins
Sets the number of days after a password expires before the password is locked out. See "User Management Overview" on page 9-7 for additional configuration information.	**set system password grace-period time**	0 off	30 days
Sets the SNMP user configuration privacy.	**set snmp user encryption**	usmNoPriv Protocol	usmAesCfg 128Protocol
Sets the SNMP user configuration authentication.	**set snmp user authentication**	usmNoAuth Protocol	usmHMACSHA AuthProtocol

Security Profile Mode Parameter Range Changes

The **set system lockout attempts** command parameter range can change when changing the security profile mode. Table 10-2 details the command parameter range change per security profile mode.

Table 10-2 Security Profile Mode Command Parameter Range Changes

Description	Command	Normal Default	C2 Default
Sets the number of failed login attempts before locking out (disabling) a read-write or read-only user account. See "User Management Overview" on page 9-7 for additional configuration information.	**set system lockout attempts**	1 – 15	2 - 5

C2 Security Profile Mode Command Access Changes

Some commands that are accessible in normal security profile mode are not accessible in C2 security profile mode. For some commands this change in access depends upon the user mode. Table 10-3 details security profile mode command access changes.

Table 10-3 Security Profile mode Command Access Changes

Description	Command	Normal Default	C2 Default
Sets the authentication type required for this user as MD5 or SHA. Only MD5 is affected by C2 security profile mode.	**set snmp user authentication md5**	Allowed	MD5 Cloaked
Sets the privacy protocol to Advanced Encryption Standard (AES) or Data Encryption Standard (DES). Only DES is affected by C2 security profile mode.	**set snmp user encryption des**	Allowed	DES Cloaked
Creates a new SNMPv3 user.	**set snmp user**	RW and SU user modes	SU user mode only
Sets the properties for one or more console ports. Only VT100 is affected by C2 security profile. See "Console Port Parameters" on page 6-3 for additional configuration information.	**set console vt100**	RW and SU user modes	SU user mode only

C2 Security Profile Mode Read-Write User Mode Changes

Some Read-Write user mode functionality accessible in normal security profile mode is not accessible when in the C2 security profile mode. Table 10-4 details Read-Write user mode functionality that is not accessible when in C2 security profile mode.

Table 10-4 Read-Write Functionality Not Accessible in C2 Security Profile Mode

Description	Command
Secure directory including secure logs. See "Configuration and Image File Display Commands" on page 3-9 for additional configuration information.	**dir**

Table 10-4 Read-Write Functionality Not Accessible in C2 Security Profile Mode

Description	Command
The display of messages logged on all blades. See "Interpreting Messages" on page 31-6 for additional configuration information.	**show logging buffer**
Script access to secure logs. See "Running a Configuration Script" on page 3-8 for additional configuration information.	**script**
Display technical support-related information output. See Table 9-3 on page 9-5 for additional configuration information.	**show support**
Display of encrypted passwords is cloaked in the **show config** command output. See "Executing a Configuration" on page 3-3 for additional configuration information.	**show config**
Display of encrypted passwords is cloaked in the **show file** command output. See Table 9-3 on page 9-5 for additional configuration information.	**show file**
The non-append version of the configure command is not available. See "Executing a Configuration" on page 3-3 for additional configuration information.	**configure**
Ability to create, modify, or delete snmp users, access views, traps configuration and engine ID is not available. See "Configuring SNMP" on page 20-7	{**set** | **clear**} **snmp** {**user** | **access** | **notify** | **engine-id**}
Ability to create, modify, or delete the authentication login method. See "Setting the Authentication Login Method" on page 9-9 for additional configuration information.	{**set** | **clear**} **authentication login**
Ability to create, modify, or delete system login, lockout, or password. See "User Management Overview" on page 9-7 for additional configuration information.	{**set** | **clear**} **system** {**login** | **lockout** | **password**}
Ability to create, modify, or delete console settings. See "Console Port Parameters" on page 6-3 for additional configuration information.	{**set** | **clear**} **console**
Ability to create, modify, or delete logging local, application, default, server or here. See "Syslog Overview" on page 31-2 for additional configuration information.	{**set** | **clear**} **logging** {**local** | **application** | **default** | **server** | **here**}
Ability to display or set the C2 security profile mode.	{**show** | **set**} **security profile**
Ability to display or set the FIPS security mode.	{**show** | **set**} **security fips mode**
Ability to display or set the security boot access mode.	{**show** | **set**} **security boot-access**
Ability to clear the configuration on all modules.	**clear config all**

C2 Security Profile Mode Read-Only User Mode Changes

Some Read-Only user mode functionality accessible in normal security profile mode is not accessible when in the C2 security profile mode. The following table provides a list of Read-Only user mode functions that are not accessible when in C2 security profile mode.

Table 10-5 Read-Only Functionality Not Accessible in C2 Security Profile Mode

Description	Command
Secure directory including secure logs.	**dir**
The display of messages logged on all blades.	**show logging buffer**
Script access to secure logs.	**script**
Display technical support-related information output.	**show support**
Display of encrypted passwords is cloaked in the **show config** and **show file** command outputs.	**show config** **show file**
The non-append version of the configure command is not available.	**configure**
Ability to display or set the C2 security profile.	**show security profile**
Ability to display the FIPS security mode.	**show security fips mode**
Ability to display the security boot access mode.	**show security boot-access**

Implementing Security Mode

To implement security mode on your network:

- Optionally restrict authentication and encryption algorithm support on the device to FIPS approved algorithms by enabling FIPS mode.
- Optionally set the security profile for the device to the C2 security level by setting the security profile to C2 mode.
- Optionally disable access to the system boot (System Image Loader) menu.

Configuring Security Mode

Table 10-6 describes security mode configuration on the Extreme Networks S-Series devices.

Table 10-6 Configuring Security Mode on the Device

Task	Command(s)
Optionally, enable FIPS security mode restricting authentication and encryption algorithms to FIPS approved algorithms.	**set security fips mode** {**enable** | **disable**}
Optionally, set the security profile to C2 mode, changing a subset of security sensitive command default and range values, as well as command access, to meet the C2 security rating specification.	**set security profile** {**c2** | **normal**}
Optionally, disable access to the boot menu during bootup.	**set security boot-access** {**enable** | **disable**}

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

Security Mode Display Commands

Table 10-7 lists security mode show commands.

Table 10-7 Security Mode Show Commands

Task	Command
To display the current boot access state for this device.	**show security boot-access**
To display the current security FIPS mode state for the device.	**show security fips mode**
To display the current security profile for the device.	**show security profile**

Refer to the *Extreme Networks S-Series CLI Reference* for a description of the output of each command.

Security Mode Configuration Example

This security mode configuration example:

- Enables FIPS mode on the device, restricting the authentication and encryption algorithms to the FIPS approved SHA1 algorithm
- Sets the device security profile to C2:
  - Changing command parameter defaults as specified in Table 10-1 on page 10-3
  - Changing command parameter ranges as specified in Table 10-2 on page 10-4
  - Changing command access as specified in Table 10-3 on page 10-4
  - Denying read-write access to commands specified in Table 10-4 on page 10-4
  - Denying read-only access to commands specified in Table 10-5 on page 10-6
- Disables access to the boot menu during bootup on the device

```
S Chassis(su)->set security fips mode enable
This command will reset the system.  Are you sure you want to continue? (y/n) [n]y
Resetting system ...
S Chassis(su)->set security profile c2
This command will reset the system.  Are you sure you want to continue? (y/n) [n]y
Resetting system ...
S Chassis(su)->set security boot-access disable
```

Terms and Definitions

Table 10-8 lists terms and definitions used in this security mode configuration discussion.

Table 10-8 Security Mode Configuration Terms and Definitions

Term	Definition
C2 security mode	C2 security mode is a security rating established by the U.S. National Computer Security Center (NCSC) and specifies that a product passes the Department of Defense (DoD) Trusted Computer System Evaluation Criteria (TCSEC) tests.

Table 10-8 Security Mode Configuration Terms and Definitions (continued)

Term	Definition
FIPS security	FIPS security mode puts the switch into Federal Information Processing Standards (FIPS) mode, where only FIPS approved authentication and encryption algorithms and methods are used.
security profile	Security profile sets the device security mode to either a normal (standard) level of security or to the C2 security mode.
system boot menu	The System Image Loader menu accessible during boot up of an S-Series device.

11

IPsec Protocol Configuration

This chapter provides information about configuring and monitoring the IPsec protocol on S-Series devices.

How to Use IPsec in Your Network

The Internet Protocol Security Architecture (IPsec), defined in RFC 4301, describes how to provide a set of security services for traffic at the IP layer in both IPv4 and IPv6 environments. As described in the RFC, for this release, security services are provided through use of the Encapsulating Security Payload (ESP) traffic security protocol, and through the use of cryptographic key management procedures and protocols.

The IPsec implementation on the S-Series provides the following functionality:

- IPsec and IKE (Internet Key Exchange protocol) are defined for the RADIUS host application only. This implementation supports configuring the default Security Association (SA) with servers configured for RADIUS, and the RADIUS application helps define the IPsec flow.
- Only the Encapsulating Security Payload (ESP) mode of operation is supported. Authentication Header (AH) mode is not supported.
- IKEv1 is supported.

Note: Although the use of certificates will be supported for IPsec in future releases, in the current release, only use of a shared secret is supported.

- HMAC-SHA1 is the supported IKE integrity mechanism.
- 3DES and the Advanced Encryption Standard (AES) encryption algorithms are supported. AES supports key lengths of 128, 192, and 256 bits.
- IPsec does not prevent the independent simultaneous use of MSCHAP-V2 style encryption of user passwords between the switch and the RADIUS server.
- If FIPS security mode is enabled, using the **set security fips mode** command, only the SHA1 authentication algorithm is supported.

IPsec Implementation Requirements

The following parameter configuration is required to implement IPsec on your network:

- IPsec is disabled by default. You must enable IPsec.
- IPsec is disabled by default for RADIUS transactions. You must configure the RADIUS management MS-CHAPv2 password attribute for IPsec to work. See the **set radius mgmt attribute** command details in the RADIUS commands chapter of the *Extreme Networks S-Series CLI Reference* for information on configuring the RADIUS management MS-CHAPv2 password attribute.

Configure the IPsec default instance by assigning an IKE map to it.

Optional IPsec and IKE configuration includes enabling IPsec traps.

Required Manual Configuration

For this release, a number of IPsec parameters do not support default values and must be manually configured.

IKE Proposal

For this release, the following IKE proposal parameters do not support default values and must be manually configured:

- The IKE Diffie-Hellman group.
- The IKE proposal encryption.
- The IKE proposal SHA1 hash and integrity (authentication).

IKE Policy

The following IKE policy parameters must be manually configured:

- The Authentication Pre-Shared Key (PSK)
- The IKE policy lifetime. This release does not provide a default value.
- The policy SA peer (server).
- The IKE proposal assignment to the IKE policy.
- The IKE version. This release does not provide a default value.

IKE Map

For this release, the following IKE map parameters must be manually configured:

- The UDP protocol. This release does not provide a default value.
- The destination IPv4 or IPv6 address (server).
- The encapsulation mode. This release does not provide a default value.
- The IKE map lifetime and bandwidth. This release does not provide a default value.
- The IKE policy assigned to the IKE map.
- The IKE proposal assigned to the IKE map.
- The source IPv4 or IPv6 address (local device).

Understanding the IPsec Protocol

IPsec is an end-to-end security scheme protocol suite that secures IP communications using authentication and encryption of each communication session IP packet. IPsec can be used to protect data flows on a host-to-host and host-to-network basis. IPsec protects any application traffic across an IP network. Applications do not need to be specifically designed to use IPsec.

The S-Series IPsec implementation uses the ESP and SA protocols from the IPsec protocol suite. ESP provides for packet:

- Authenticity – ensures that the owner of the packet is who he claims to be
- Integrity – ensures that the contents of the packet have not been tampered with
- Confidentiality – ensures that information is accessible only to those authorized to have access

ESP operates directly on top of IP, using IP protocol number 50.

The Security Association (SA) protocol provides a bundle of algorithms and data required for ESP operations that are the basis for IPsec. The algorithms and data configured within an SA are used to encrypt and authenticate a particular flow in one direction. In a standard bi-directional communications session, two SAs are used, one for each direction. Security associations are established using the Internet Security Association and Key Management Protocol which provides for manual configuration of pre-shared secrets (keys) using the Internet Key Exchange (IKE).

IPsec identifies the SA that determines the protection to provide to an outgoing packet based upon a Security Parameter Index (SPI) and the packet header destination address. The SPI is an index to the security association database.

The S-Series IPsec implementation supports the configuration of a default SA. A default SA is configured by entering the IPsec default instance configuration mode and assigning an IKE map to the default SA.

IKE Map

An IKE map groups together all algorithms and parameters that make up the SA. An IKE map contains the following parameters:

- The IKE proposal which groups the IKE map algorithms configured for the SA
- The IKE policy which groups policy related parameters configured for the SA
- The source and destination IP address and port for the SA
- The encapsulation type for the SA
- The map lifetime in time and bandwidth
- The transmission protocol (UDP) used by the SA
- Whether or not encryption is required

Use the **crypto ike-map** command in global VRF router configuration mode to create or modify an IKE map and enter IKE map configuration mode.

IKE Proposal

The IKE proposal groups together the IKE map algorithms configured for the SA.

There are two IKE modes to which proposals are assigned: main mode and quick mode. The same IKE proposal can be assigned to both modes, or each mode can be assigned a unique IKE proposal depending upon your configuration needs.

The main mode or key exchange proposal is assigned to an IKE map in IKE map configuration mode. Main mode is the IKE negotiation that establishes a secure channel, known as the Internet Security Association and Key Management Protocol (ISAKMP) SA, between two devices.

Quick mode (also known as Phase 2) is the IKE negotiation that establishes a secure channel between two computers to protect data. Quick mode negotiates on behalf of the IPsec SAs. During quick mode, keying material is refreshed or, if necessary, new keys are generated. The quick mode proposal is assigned to an IKE policy using the **proposal** command in IKE policy configuration mode.

Use the **crypto ike-proposal** command in global VRF router configuration mode to create or modify an IKE proposal. Specify the name of the IKE proposal when entering the command. Upon entering the command, you are placed in IKE proposal configuration mode for the named proposal.

See Table 11-1 for a description of IKE proposal parameters.

Use the **proposal** command in IKE map configuration mode to assign a main mode (key exchange) proposal to an IKE map.

Table 11-1 IKE Proposal Parameters

Parameter	Description
IKE Diffie-Hellman (DH) group	IKE Diffie-Hellman (DH) group is a key derivation algorithm that generates the IPsec SA key. There are three algorithms supporting key sizes 768, 1024, and 2048 bits. The larger the generated key, the greater the security, but also the greater the system overhead. Use the **dh_group** command in IKE proposal configuration mode to set the IKE DH group algorithm for the proposal.
Encryption	Encryption is the process of transforming information, usually referred to as plaintext, using an algorithm, called a cipher, to make it unreadable to anyone except those possessing the associated key. The IKE proposal supports four encryption types: • **3des** – Triple Data Encryption Standard encryption algorithm • **aes128cbc** – The Advanced Encryption Standard (AES) 128 bit key size Cipher-Block Chaining (CBC) encryption algorithm. • **aes192cbc** – The Advanced Encryption Standard (AES) 192 bit key size Cipher-Block Chaining (CBC) encryption algorithm. • **aes256cbc** – The Advanced Encryption Standard (AES) 1256 bit key size Cipher-Block Chaining (CBC) encryption algorithm. This release does not support a default encryption algorithm. You must manually enter an encryption algorithm. Use the **encryption** command in IKE proposal configuration mode to set the encryption algorithm for the IKE proposal.
Hash	The hash algorithm is used during phase 1 negotiation between the SA authenticating devices. This release supports the Secure Hash Algorithm 1 (SHA1) hash. This release does not support a hash default value. You must manually enter the hash algorithm for one to be configured. Use the **hash** command in IKE proposal configuration mode to configure the hash algorithm for the IKE proposal.

Table 11-1 IKE Proposal Parameters (continued)

Parameter	Description
Integrity	Integrity, also referred to as data authentication, verifies that the data has not been altered as opposed to a user authentication which verifies the identity of the user. This release supports SHA1 integrity. SHA1 produces a 160-bit message digest for which no known attacks or partial attacks have yet been demonstrated. This release does not support a default integrity algorithm. You must manually enter the integrity algorithm for one to be configured. Use the **integrity** command in IKE proposal configuration mode to configure the integrity algorithm for the IKE proposal.

IKE Policy

The IKE policy groups together policy related parameters configured for the SA. Use the **crypto ike-policy** command in global VRF router configuration mode to create or modify an IKE policy. Specify the name of the IKE policy when entering the command. Upon entering the command you are placed in IKE policy configuration mode for the named policy. See Table 11-2 for a description of IKE policy parameters.

Table 11-2 IKE Policy Parameters

Parameter	Description
Authentication Pre-shared Key	The authentication PSK is a pre-shared authentication key that is used to initiate the connection and exchange encryption keys during the session. Use the **authentication psk** command in IKE policy configuration mode to configure the authentication pre-shared key for the SA.
Initial Contact	If the local host has rebooted, peers may have SAs that are no longer valid. If the initial contact feature is enabled, upon reboot an initial contact message is sent to a peer so that it will delete old SAs. Use the **initial-contact** command to enable the initial contact feature for the SA. The initial contact feature is disabled by default.
Lifetime	The IKE policy lifetime specifies the life cycle of an ISAKMP SA and is configured in minutes. The policy lifetime determines when a policy times out. A lifetime renegotiation automatically occurs before the lifetime is to expire. If the renegotiation is unsuccessful, the policy expires. Use the **lifetime time** command in IKE policy configuration mode to configure an IKE policy timeout period.
Passive Mode	Passive mode configures the IKE policy to wait for the peer to initiate the IKE session. By default a device is in active mode and constantly polls to see if the peer is up. Use the passive command in IKE policy configuration mode to configure the IKE policy for passive mode.

Table 11-2 IKE Policy Parameters (continued)

Parameter	Description
Peer	An IPv4 or IPv6 peer is specified for the SA using the **peer** command in IKE policy configuration mode. The IKE policy peer configuration determines whether the associated map is IPv4 or IPv6 during the phase 1 main mode negotiation. The CLI allows you to enter an inconsistent configuration between the local and peer address IP types. Inconsistent local and peer configurations will cause the IKE map to not be programmed.
Proposal (Quick Mode)	The quick mode proposal, used to establish and refresh user-level SAs, is assigned to an IKE policy using the **proposal** command in IKE policy configuration mode.
Version	The S-Series supports IKE version 1 for this release. Use the **version** command in IKE policy configuration mode to specify the IKE version used for the policy. This release does not support a default IKE version. You must manually enter an IKE version.

Source and Destination Address and Port

Source and destinations addresses need to be configured for the IKE map. The source address is the local address. The destination address is the remote address. Both IPv4 and IPv6 addresses are supported. Address ranges are supported using the slash (/) length notation.

Use the **dst** command in IKE map configuration mode to configure a destination address for the IKE map.

Use the **src** command in IKE map configuration mode to configure a source address for the IKE map.

The default SA port for the IKE map is any port. The source and destination ports can be specified for the SA when a specific protocol such as UDP is being authenticated. If you wish to limit the IKE map to a specific port, configure the source and destination port.

Use the **dst-port** command in IKE map configuration mode to configure the IKE map for a specific destination port.

Use the **src-port** command in IKE map configuration mode to configure the IKE map for a specific source port.

Encapsulation

The SA encapsulation is determined by the type of communications required and determines whether the whole packet or only the data portion of the packet is encrypted and authenticated. There are two modes of encapsulation:

- Transport mode is used for host-to-host communications. In transport mode, only the transferred data of the IP packet is encrypted or authenticated. The routing is intact, since the IP header is neither modified nor encrypted; however, when the authentication header is used, the IP addresses cannot be translated, because to do so would invalidate the hash value.
- Tunnel mode is used to create virtual private networks. In tunnel mode, the entire IP packet is encrypted or authenticated. It is then encapsulated into a new IP packet with a new IP header.

This release does not support a default SA encapsulation. You must manually configure IKE map encapsulation.

Use the **encapsulation** command in IKE map configuration mode to specify the encapsulation mode to use for the SA.

SA Lifetime

A lifetime can be set for the SA in both seconds and aggregate bandwidth. When a lifetime expires the SA is renegotiated as a security measure.

This release does not support a default SA lifetime. You must manually configure an SA lifetime for this IKE map.

Use the **lifetime** command in IKE map configuration mode to configure a lifetime value for the SA.

Transmission Protocol

The UDP protocol is supported for SA packet transmission. This release does not support a default SA transmission protocol. You must manually configure an SA transmission protocol for this IKE map.

Use the **protocol udp** command to specify UDP as the SA transmission protocol.

Encryption Request

By default, encryption is required to be used for the SA both locally and by the peer. If the peer does not support encryption, packets are not sent for the SA. If encryption request is enabled and the peer does not support encryption, packets are sent unencrypted. Use the **request** command to set the requirement for encryption to request for the SA. Request is disabled by default.

Configuring IPsec

IKE Proposal Configuration

Procedure 11-1 describes IKE proposal configuration on the Extreme Networks S-Series devices.

Procedure 11-1 Configuring an IKE Proposal

Step	Task	Command(s)
1.	Enter IKE proposal configuration mode, from the global VRF router configuration mode, to create a new or modify an existing IKE proposal.	**crypto ike-proposal** *proposal-identifier*
2.	In IKE proposal configuration mode, configure the IKE Diffie-Hellman (DH) key exchange group for the SA: • DH group 1 (modp768) • DH group 2 (modp1024) • DH group 14 (modp2048)	**dh_group** {**1** | **2** | **14**}
3.	In IKE proposal configuration mode, configure the encryption algorithm for the IKE proposal.	**encryption** {**3des** | **aes128cbc** | **aes192cbc** | **aes256cbc**}

Procedure 11-1 Configuring an IKE Proposal (continued)

Step	Task	Command(s)
4.	In IKE proposal configuration mode, configure the hash algorithm for the IKE proposal.	**hash sha1**
5.	In IKE proposal configuration mode, configure the integrity (data authentication) algorithm for the IKE proposal.	**integrity sha1**

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

IKE Policy Configuration

Procedure 11-2 describes IKE policy configuration on the Extreme Networks S-Series devices.

Procedure 11-2 Configuring an IKE Policy

Step	Task	Command(s)
1.	Enter IKE policy configuration mode, from the global VRF router configuration mode, to create a new or modify an existing IKE policy.	**crypto ike-policy** *policy-identifier*
2.	In IKE policy configuration mode, configure the authentication pre-shared key (PSK) for the IKE policy.	**authentication psk** *pre-shared-key*
3.	In IKE policy configuration mode, optionally enable initial contact feature for the IKE policy.	**initial-contact**
4.	In IKE policy configuration mode, configure the lifetime for the IKE policy.	**lifetime time** *minutes*
5.	In IKE policy configuration mode, optionally enable passive mode for the IKE policy	**passive**
6.	In IKE policy configuration mode, configure the SA peer for the IKE policy.	**peer** *address*
7.	In IKE policy configuration mode, assign an IKE proposal to the IKE policy.	**proposal** *proposal-identifier*
8.	In IKE policy configuration mode, configure the IKE version for the IKE policy.	**version** *version*

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

IKE Map Configuration

Procedure 11-3 describes IKE map configuration on the Extreme Networks S-Series devices.

Procedure 11-3 Configuring an IKE Map

Step	Task	Command(s)
1.	Enter IKE map configuration mode, from the global VRF router configuration mode, to create a new or modify an existing IKE map.	**crypto ike-map** *map-identifier*
2.	In IKE map configuration mode, configure a source address for the IKE map.	**src** *address*

Procedure 11-3 Configuring an IKE Map (continued)

Step	Task	Command(s)
3.	In IKE map configuration mode, optionally configure a source port for the IKE map. The source port defaults to any port.	**src-port** *port*
4.	In IKE map configuration mode, configure a destination (peer) address for the IKE map.	**dst** *address*
5.	In IKE map configuration mode, optionally configure a destination port for the IKE map. The destination port defaults to any port.	**dst-port** *port*
6.	In IKE map configuration mode, configure the encapsulation mode for the IKE map.	**encapsulation** {**tunnel** | **transport**}
7.	In IKE map configuration mode, configure the lifetime in time or bandwidth for the IKE map. Both values can be configured using separate command entries.	**lifetime** {**time** *minutes* | **bandwidth** *kilobytes*}
8.	In IKE map configuration mode, assign the specified IKE policy to the IKE map.	**policy** *policy-identifier*
9.	In IKE map configuration mode, assign the specified main mode key exchange IKE proposal to the IKE map.	**proposal** *proposal-identifier*
10.	In IKE map configuration mode, configure the IKE map with the UDP transmission protocol.	**protocol udp**
11.	In IKE map configuration mode, request that encryption be used by the SA and to not used it if encryption is not supported by the peer. Defaults to encryption required.	**request**

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

IPsec Configuration

Procedure 11-4 describes IPsec configuration on the Extreme Networks S-Series devices.

Procedure 11-4 Configuring IPsec

Step	Task	Command(s)
1.	Enter the IPsec default SA configuration mode, from the global VRF router configuration mode, to configure IPsec on the device.	**crypto ipsec default**
2.	In IPsec default SA configuration mode, assign an IKE map to the IPsec default SA.	**ike map** *ike-map*
3.	In global VRF router configuration mode, optionally enable IPsec traps.	**crypto ipsec trap-enable**
4.	In global VRF router configuration mode, enable IPsec on the router.	**crypto ipsec enable**

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

IPsec Display Commands

Table 11-3 lists IPsec show commands.

Table 11-3 IPsec Show Commands

Task	Command
To display IKE statistics.	**show ike stats**
To display IKE proposal configuration.	**show ike proposal**
To display IKE policy configuration.	**show ike policy**
To display IKE map configuration.	**show ike map**
To display IKE SA information.	**show ike sa**
To display IPsec counters.	**show ipsec counters** [**all** | **ipsec** | **global** | **memory** | **resources** | **task**]
To display IPsec map information.	**show ipsec map**
To display IPsec SA information.	**show ipsec sa** [**spi** *spi*] [**instance_id** *instance_id*] [**index** *index*] [**ipv4** | **ipv6**] [**brief**]
To display IPsec flow information.	**show ipsec flow** [**spi** *spi*] [**instance_id** *instance_id*] [**index** *index*] [**ipv4** | **ipv6**] [**brief**]
To display IPsec instance information.	**show ipsec instance** [**vlan** *vlan-id*] [**instance_id** *instance_id*] [**index** *index*] [**static** | **dynamic**] [**brief**]
To display IPsec interface information.	**show ipsec instance** [**vlan** *vlan-id*] [**instance_id** *instance_id*] [**static** | **dynamic**] [**brief**]

Refer to the *Extreme Networks S-Series CLI Reference* for a description of the output of each command.

IPsec Configuration Example

No specific order is required for configuring the IKE map that is applied to the default SA. Any required parameter or algorithm not properly configured will display as an incomplete configuration, but will not prevent you from configuring other IKE map parameters. A suggested order of IPsec configuration is:

1. Configure the main mode key exchange IKE proposal for the SA. Optionally configure a quick mode data protection IKE proposal if the quick mode proposal is different from the main mode proposal.
2. Configure the IKE policy for the SA.
3. Configure the IKE map for the SA.
4. Enter IPsec default instance configuration mode.
5. Apply the IKE map to the default SA
6. Enable IPsec

The following configuration example will follow the IPsec configuration order suggested above.

IKE Proposal

As indicated in "IKE Proposal" on page 11-3, there are two IKE modes to which an IKE proposal is assigned:

- **Main**, which is assigned to an IKE map
- **Quick**, which is assigned to an IKE policy

Each IKE mode can be assigned a unique IKE proposal or the same proposal may be assigned to both modes, depending upon your configuration requirements. For this example we will configure a single IKE proposal named **winRadiusPro**, to be used in both IKE modes, with the following values:

- IKE Diffie-Hellman key exchange group – **14**
- Encryption – **aes128cbc**
- Hash – **SHA1**
- Integrity – **SHA1**

```
S Chassis(su)->configure
S Chassis(su-config)->crypto ike-proposal winRadiusPro
S Chassis(su-crypto-proposal)->dh_group 14
S Chassis(su-crypto-proposal)->encryption aes128cbc
S Chassis(su-crypto-proposal)->hash sha1
S Chassis(su-crypto-proposal)->integrity sha1
S Chassis(su-crypto-proposal)->exit
S Chassis(su-config)->
```

IKE Policy

The IKE policy for this example is named **winRadiusPol**. The initial contact and passive mode features will not be enabled for this configuration. The **winRadiusPol** IKE policy is configured with the following values:

- Authentication pre-shared key – **testkey**
- Lifetime – **360** minutes
- Peer address – **1.1.191.22**
- IKE quick proposal – **winRadiusPro**
- IKE version – **1**

```
S Chassis(su-config)->crypto ike-policy winRadiusPol
S Chassis(su-crypto-policy)->authentication psk testkey
S Chassis(su-crypto-policy)->lifetime time 360
S Chassis(su-crypto-policy)->peer 1.1.191.22
S Chassis(su-crypto-policy)->proposal winRadiusPro
S Chassis(su-crypto-policy)->version 1
S Chassis(su-crypto-policy)->exit
S Chassis(su-config)->
```

IKE Map

The IKE map for this example is named **winRadius**. IKE map parameters are configured with the following values:

- IKE main proposal – **winRadiusPro**
- IKE policy – **winRadiusPol**
- Source IP address – **192.1.1.0/24**

- Source port – standard RADIUS port **500**
- Destination IP address – **192.2.2.0/24**
- Destination port – standard RADIUS port **500**
- Encapsulation type – **transport**
- Lifetime time – **5** minutes
- Lifetime bandwidth – **100000** kilobytes
- The transmission protocol – **udp**
- Encryption request – **enabled**

```
S Chassis(su-config)->crypto ike-map winRadius
S Chassis(su-crypto-map)->proposal winRadiusPro
S Chassis(su-crypto-map)->policy winRadiusPol
S Chassis(su-crypto-map)->src 192.1.1.0/24
S Chassis(su-crypto-map)->src-port 500
S Chassis(su-crypto-map)->dst 192.2.2.0/24
S Chassis(su-crypto-map)->dst-port 500
S Chassis(su-crypto-map)->encapsulation transport
S Chassis(su-crypto-map)->lifetime time 5
S Chassis(su-crypto-map)->lifetime bandwidth 100000
S Chassis(su-crypto-map)->protocol udp
S Chassis(su-crypto-map)->request
S Chassis(su-crypto-map)->exit
S Chassis(su-config)->
```

IPsec

For this release an IPsec default instance is configurable. You assign the IKE map **winRadius** to the IPsec default instance within IPsec default instance configuration mode. You enable IPsec on the router in global VRF router configuration mode. For this IPsec configuration example we will also enable IPsec traps.

```
S Chassis(su-config)->crypto ipsec default
S Chassis(su-crypto-ipsec-defaul)->ike map winRadius
S Chassis(su-crypto-ipsec-defaul)->exit
S Chassis(su-config)->crypto ipsec trap-enable
S Chassis(su-config)->crypto ipsec enable
S Chassis(su-config)->
```

Terms and Definitions

Table 11-4 lists terms and definitions used in this IPsec configuration discussion.

Table 11-4 IPsec Configuration Terms and Definitions

Term	Definition
Encapsulating Security Payload (ESP)	An IPv4 and IPv6 packet header designed to provide a mix of security services including: confidentiality, data origin authentication, connectionless integrity, depending upon supported and configured SA configuration.

Table 11-4 IPsec Configuration Terms and Definitions (continued)

Term	Definition
Encryption	The process of transforming information, usually referred to as plaintext, using an algorithm, called a cipher, to make it unreadable to anyone except those possessing the associated key.
ESP Authenticity	An ESP feature that ensures that the owner of the packet is who he claims to be.
ESP Confidentiality	An ESP feature that ensures that information is accessible only to those authorized to have access.
ESP Integrity	An ESP feature, also referred to as data authentication, that ensures that the contents of the packet have not been tampered with.
Hash	The Secure Hash Algorithm 1 (SHA1) hash algorithm is used during phase 1 negotiation between the SA authenticating devices.
IKE Diffie-Hellman Group	A method of exchanging keys allowing two parties that have no prior knowledge of each other to jointly establish a shared secret key over an insecure communications channel.
IKE Map	A bundling of all algorithms and parameters that make up the SA.
IKE Policy	A combination of security parameters that exist both locally and on the peer, to be used during the IKE SA negotiation.
IKE Proposal	A set of parameters applied to both Phase I and Phase II IPSec negotiations during which the two peers establish a secure connection by which they then negotiate the Phase 2 parameters.
Initial Contact	A feature that when enabled sends an initial contact message to the peer upon reboot instructing the peer to delete old SAs.
Internet Key Exchange protocol (IKE)	The protocol used to set up a Security Association (SA) in the IPsec protocol suite.
IPsec	The Internet Protocol Security Architecture, defined in RFC 4301, that provides a set of security services for traffic at the IP layer in both IPv4 and IPv6 environments.
Security Association (SA)	The establishment of shared security attributes between two network entities to support secure communication within the IPsec protocol suite.
Security Parameter Index (SPI)	An index to the security association database that helps in differentiating between two traffic streams where different encryption rules and algorithms may be in use.

12

Public-Key Infrastructure (PKI) Configuration

This chapter provides information about configuring and monitoring Public-Key Infrastructure (PKI) in an SSH server context on S-Series devices.

Using Public-Key Infrastructure (PKI) in Your Network

The S-Series PKI implementation supports the secure authentication of an SSH client to an Extreme Networks S-Series device using an X.509 certificate and authorization using RADIUS, TACACS, or local policy.

There are three primary aspects to PKI configuration:

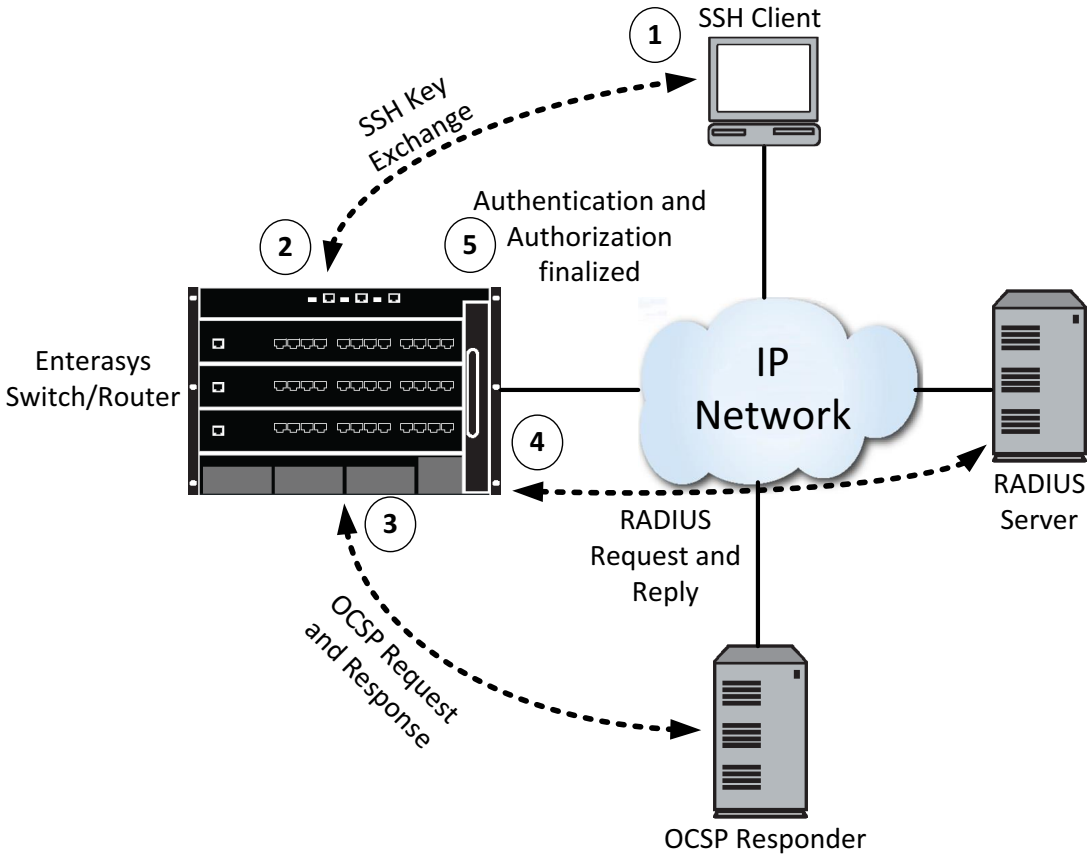
- **X.509 certificate** – The specification of a certificate issued by a Certification Authority (CA) that binds a public key to an organizational or common name or an alternative DNS-entry. The X.509 certificate commands allow users to enter X.509 certificates via the command line and to group these certificates into lists. An SSH server requiring PKI services references these certificate lists.
- **Online Certificate Status Protocol** (**OCSP**) – An Internet protocol, defined in RFC 2560, used for obtaining the revocation status of an X.509 digital certificate. The OCSP commands are used to enable, disable and configure certificate revocation checking.
- **Authentication and Authorization**– The verification of the user certificate's issuance chain back to the certificate authority by the SSH server in order to determine whether the user is who they claim to be followed by a verification of the validity of the public user certificate. The authentication commands define a set of rules used for extracting a user's authentication credentials from the X.509 certificate's subject field. The extracted credential is then presented to a RADIUS, TACSACS+ or local authentication server.

Note: The SSH server must be configured for SSH client authentication using PKI. See "Configuring Secure Shell" on page 9-22 for SSH server authentication configuration details.

There are no PKI MIB objects. PKI is exclusively managed by the CLI. CLI users with admin access (**su**) can set, show and clear all of the PKI configuration objects. Users with read-only (**ro**) or read-write (**rw**) access are restricted to displaying show commands.

Figure 12-1 on page 12-2 presents a PKI login flow overview in a RADIUS server authorization context.

Figure 12-1 Public-Key Infrastructure Login Flow Overview



Callout **1** is the initial series of message exchanges initiated by the SSH client. The S-Series device providing the SSH client with the list of supported authentication methods one of which is public key. The SSH client responds with its public key certificate.

At Callout **2**, the S-Series device checks to make sure the certificate signature from SSH client matches a trusted certificate authority's certificate defined in PKI certificate authority list on the S-Series SSH server.

At Callout **3**, the S-Series device sends an OCSP request that contains the Client's certificate serial number to the OCSP responder to check the validity of the Clients X.509 certificate, and the OCSP Responder uses the serial number to look up the revocation status of the SSH client's certificate. If the OCSP responder determines that the certificate has not been revoked by the certificate authority, the server sends back a GOOD response. The responder certificate is an OCSP signing certificate issued by the CA that issued the certificate that is being validated. Supported certificates are common issuer, Delegated Trust Model (DTM), and Trusted Responder Model (TRM) as defined in FRC 5280. When using TRM, use the **set pki ocsp signature-ca-list** command to specify the trusted list. Lists are created using the **set pki certificate** command.

At Callout **4**, the S-Series device queries the setting of the PKI authorization user name and potentially prompts for the RADIUS password. These values will be used to verify the Authorization of the SSH client's user.

At Callout **5**, Radius Authorization is configured on the S-Series device. The resulting Radius Access Request contains the appropriate username and password. Radius Server sends Access-Accept message and the SSH client is now both authenticated (PKI) and authorized (RADIUS) and SSH negotiates a PTY and a shell to use for the user login session.

Implementing Public-Key Infrastructure

To implement Public-Key Infrastructure:

1. Add one or more PEM formatted CA certificates to a certificate list.
2. Configure OCSP with a list of trusted CA certificates used to verify OCSP response signatures.
3. Optionally, configure an alternate OCSP responder (OCSR) URL for the OCSR used to check revocation status.
4. Perform one, but not both, of the following:
   - Restrict the system to a single specified authorization credential which must be shared by all users.
   - Configure a dynamic extracted username from the X.509 certificate subject field.
5. Configure the SSH server for PKI (see "Configuring Secure Shell" on page 9-22).

Public-Key Infrastructure Configuration Overview

The X.509 Certificate

PKI uses the X.509 certificate to authenticate an SSH client with the S-Series device SSH server. The X.509 certificate is issued by a CA and binds a public key to an organizational name, common name, or DNS-entry. A PKI service is configured with one or more X.509 certificates. X.509 certificates are grouped in certificate lists. When using PKI services, SSH references these certificate lists when authenticating.

The X.509 certificate contains:

- User Information: a subject (username), issuer (the certificate signer), and a validity period made up of a start and stop time
- Public Key
- CA Signature

Use the **set pki certificate** command to configure PKI with an X.509 certificate and group the configured X.509 certificates in a certificate list.

The user entering the command must have admin (**su**) privilege. Users with read-only, read-write, or admin privilege can display PKI settings using the **show pki certificate** command.

Once you enter the command specifying the name of the certificate list to be entered, you are asked to enter the PKI certificate:

```
Enter the PEM encoded certificate-list-name certificate
```

Certificate data must be entered in Privacy Enhanced Mail (PEM) format, complete with the appropriate X.509 header **-----BEGIN CERTIFICATE-----** and footer **-----END CERTIFICATE-----**. Certificate entry is terminated by entering a blank line or the word "quit" on a line by itself.

Certificate information then displays. If you did not specify the **no-confirm** command option, you are asked to confirm the entered certificate.

This example shows how to set the myTrustedOcspSigningCerts PKI certificate, followed by a display of the entered certificate details:

```
S Chassis(su)->set pki certificate myTrustedOcspSigningCerts
Enter the PEM encoded myTrustedOcspSigningCerts certificate
End with a blank line or the word "quit" on a line by itself
-----BEGIN TRUSTED CERTIFICATE-----
MIIELjCCAxagAwIBAgIBBDANBgkqhkiG9w0BAQUFADBbMQswCQYDVQQGEwJVUzES
MBAGA1UEChMJRW50ZXJhc3lzMQwwCgYDVQQLEwNEb0QxDDAKBgNVBAsTA1BLSTEc
MBoGA1UEAxMTRXN5cyBKSVRDIFJvb3QgQ0EgMjAeFw0xMjAyMjExODQ0MTRaFw0y
MjAyMTgxODQ0MTRaMGsxCzAJBgNVBAYTAlVTMRIwEAYDVQQKEwlFbnRlcmFzeXMx
DDAKBgNVBAsTA0RvRDEMMAoGA1UECxMDUEtJMSwwKgYDVQQDEyNFc3lzIEpJVEMg
Um9vdCBDQSAyIE9DU1AgRGVsZWdhdGUgMjCCASIwDQYJKoZIhvcNAQEBBQADggEP
ADCCAQoCggEBAKvefxWIoURH/32iw8mS64MIc0k0+/8zN2lHf/s+T+MbqlmUqriC
Ax2JfCGM1jcpgQB4gdMU0fqMsgb1aQ5Vy3adtAzj7jZ9IS3OmX2O0ZBRi4rXr1dg
NukkfOdSBg68/pzzjdaZEsbeeXNdZnbtlemex+9KvBJ9TLw8pt4ZxQF12AIulRAI
Ov4WVcpnHHQL7WAcEcF56xqcYLkDYKDHhqkwanM8kEnHptWvTVqv9hEr054wu88a
lqzPYLnhNdY8mqsOAFuBM/kJcblSZjb+VI4bfwOAAn/SikbBqn9+9jG4lE1WUPDB
sWIdfZt6p+7tF3kx+ayfx0aYvFGunoi6RrECAwEAAaOB7DCB6TAOBgNVHQ8BAf8E
BAMCAYYwgYMGA1UdIwR8MHqAFFckAV1bJeN4QrJH3z97+YOQyrLgoV+kXTBbMQsw
CQYDVQQGEwJVUzESMBAGA1UEChMJRW50ZXJhc3lzMQwwCgYDVQQLEwNEb0QxDDAK
BgNVBAsTA1BLSTEcMBoGA1UEAxMTRXN5cyBKSVRDIFJvb3QgQ0EgMoIBBTAdBgNV
HQ4EFgQUS9Nou/9KbX2HFzFcsWqJf3HklyIwDAYDVR0TAQH/BAIwADATBgNVHSUE
DDAKBggrBgEFBQcDCTAPBgkrBgEFBQcwAQUEAgUAMA0GCSqGSIb3DQEBBQUAA4IB
AQCXKen2sXv68AaA7JK1uJhVD9xRuWw7O+J3Q8zA4B/BM5vkhiZZMK+Ro70HaQSI
ebAjrXsZ1VUD1pS5nkud2TawYwICyL8jxxbIX9nnIC6esr9shmCaxv/pCXMI5iZr
3zPism/n8OJpk6ZR75F/8Tnt8lUXrSFvJdwxb76nFR6zPStNorSuSgrZaGtmftUj
xZs7/PKXxWoryZmfua6oIg7SACWApBSu6Jhj7lgS6wAvow4K3WCbso+afmnpcNT7
kMkWJO7J4jUaKS/yjn8xkO2HhZZ+g1Lh1lK00i+hOx515aUHj2DpxMNQtiTvNnJr
5LJ+xqz0gfSDJB385ZTM6o4b
-----END TRUSTED CERTIFICATE-----
quit
Entered certificate has the following attributes:
  Fingerprint: a2:33:a9:df:df:8a:fb:9a:d2:f0:5e:c0:c3:8a:8a:4b:ad:0a:6f:1b
        Issuer: C=US, O=Enterasys, OU=DoD, OU=PKI, CN=Esys JITC Root CA 2
        Validity
```

```
            Not Before: Feb 21 18:44:14 2012 GMT
            Not After : Feb 18 18:44:14 2022 GMT
        Subject: C=US, O=Enterasys, OU=DoD, OU=PKI, CN=Esys JITC Root CA 2 OCSP
Delegate 2


Do you accept this certificate (y/n) [n]?y
S Chassis(su)->
```

Enabling Certificate Revocation Checking

Certificate revocation checking uses OCSP to determine whether a certificate, presented by the SSH client to the SSH server and bound to the public key and password, has been revoked by the CA prior to its expiration date.

A Certificate Authority (CA) may need to revoke an issued certificate's authorization prior to the issued certificate's expiration date. Some reasons for revocation include:

- The user was compromised (keyCompromise)
- A CA in the chain was compromised (cACompromise)
- A newer certificate was issued (superseded)

When OCSP is disabled, checking is not performed and the revocation status of all certificates is assumed to be good (not revoked).

When OCSP is enabled, the switch will attempt to obtain revocation status from one of the available OCSP Responders (OCSRs). If an OCSR replies with a revocation status of good, certificate chain verification will resume. If an OCSR replies with a request failure or with a certificate revocation status other than good (revoked or unknown), certificate authentication will fail. Request failures can be:

- Malformed Request
- OCSR Internal Error
- Try Later
- Signature Required
- Unauthorized

If the queried OCSR cannot be reached or does not reply, an alternate OCSR will be queried. If the list of available OCSRs is exhausted, revocation checking as well as certificate verification will be declared a failure.

The certificate used to sign an OCSR response must itself be successfully verified and revocation checked. Any verification failure or revocation of the OCSP signing certificate will be treated as an authentication failure of the user certificate.

At this time, only the leaf certificate in a user certificate chain will have its revocation status checked. It is assumed that if one of intermediate or root CA certificates in the chain has been revoked, then the OCSP Responder will not return "successful" when the user certificate is queried.

Use the **set pki ocsp** command to globally enable or disable OCSP certificate revocation checking.

This example shows how to disable OCSP certificate revocation checking on the device:

```
S Chassis(su)->set pki ocsp disable
S Chassis(rw)->
```

Specifying an OCSP Signature Certificate Authority List

OCSP signing certificate trust is established by matching a signing certificate with a local configuration of the OCSP signing authority in question. This option is specified in Section 4.2.2.2 Authorized Responders of RFC 2560 *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP* as a way of verifying that the entity which issued the OCSP signing certificate is actually authorized to sign a particular certificate's OCSP response message. The OCSP CA list is only required for TRM; it is not used for DTM and common issuer.

The specified PKI certificate list is configured using the **set pki certificate** command. This list must contain the expected OCSP Response signing certificate. Additionally, this certificate must contain a trusted use extension which permits OCSP signing.

A "trusted use extension" can be appended to a certificate using OpenSSL. The following example appends a trusted use extension specifying an original file and the trusted file: **ocsp-sig-ca.pem** is the original certificate file and the output file **trusted-ocsp-sig-ca.pem** is the trusted file:

```
% openssl x509 -in ocsp-sig-ca.pem -addtrust OCSPSigning -out
trusted-ocsp-sig-ca.pem
```

What follows is an example of an original certificate followed by the openssl command output trusted certificate with the modifications to the original certificate bolded:

```
-----BEGIN CERTIFICATE-----
MIICgTCCAeqgAwIBAgIJAMng4JQ0MOeIMA0GCSqGSIb3DQEBBQUAMGAxCzAJBgNV
BAYTAlVTMRIwEAYDVQQKEwlFbnRlcmFzeXMxDDAKBgNVBAsTA0RvRDEMMAoGA1UE
CxMDUEtJMSEwHwYDVQQDExhFc3lzIEpJVEMgT0NTUCBSZXNwb25kZXIwHhcNMTIw
MjE3MTg0MzEwWhcNMjIwMjE0MTg0MzEwWjBgMQswCQYDVQQGEwJVUzESMBAGA1UE
ChMJRW50ZXJhc3lzMQwwCgYDVQQLEwNEb0QxDDAKBgNVBAsTA1BLSTEhMB8GA1UE
AxMYRXN5cyBKSVRDIE9DU1AgUmVzcG9uZGVyMIGfMA0GCSqGSIb3DQEBAQUAA4GN
ADCBiQKBgQCuyC9QHBpP/n6aOS+Cx0mbgsQTS1LAUUCwxjvJdILGVfdjFB8PKG+o
W4jm7FKuRHR7uzBvAFzD9DbVkziHl2yIsy4SeiSBTQpNvHPjvUcec3rTlw7saiTw
B+CTqEm1pxcEdRKTvawK2k1ujHML1MABP2CA3SEptO+Ude4UkXMBywIDAQABo0Mw
QTAdBgNVHQ4EFgQUYFhsLiklZh0riJ1Hg7d4HPcLlBUwCwYDVR0PBAQDAgGGMBMG
A1UdJQQMMAoGCCsGAQUFBwMJMA0GCSqGSIb3DQEBBQUAA4GBADU4aQ6f8pHWLd7z
vZ8pJ8e8UCvKok1LmdXbax5TBonyyLmb7AjLrOWjZ7LKSufJL1KOBsetd5Q49LFK
h70V2fRWpGNQszpAV60WfidkNvQ0koZczEjYRQOCtMDUqxMHxsMv2MLEVE9QuGLt
+NWjeeF03E1DT3C4mnbVsTyWPZij
-----END CERTIFICATE-----

-----BEGIN TRUSTED CERTIFICATE-----
MIICgTCCAeqgAwIBAgIJAMng4JQ0MOeIMA0GCSqGSIb3DQEBBQUAMGAxCzAJBgNV
BAYTAlVTMRIwEAYDVQQKEwlFbnRlcmFzeXMxDDAKBgNVBAsTA0RvRDEMMAoGA1UE
CxMDUEtJMSEwHwYDVQQDExhFc3lzIEpJVEMgT0NTUCBSZXNwb25kZXIwHhcNMTIw
MjE3MTg0MzEwWhcNMjIwMjE0MTg0MzEwWjBgMQswCQYDVQQGEwJVUzESMBAGA1UE
ChMJRW50ZXJhc3lzMQwwCgYDVQQLEwNEb0QxDDAKBgNVBAsTA1BLSTEhMB8GA1UE
AxMYRXN5cyBKSVRDIE9DU1AgUmVzcG9uZGVyMIGfMA0GCSqGSIb3DQEBAQUAA4GN
ADCBiQKBgQCuyC9QHBpP/n6aOS+Cx0mbgsQTS1LAUUCwxjvJdILGVfdjFB8PKG+o
W4jm7FKuRHR7uzBvAFzD9DbVkziHl2yIsy4SeiSBTQpNvHPjvUcec3rTlw7saiTw
B+CTqEm1pxcEdRKTvawK2k1ujHML1MABP2CA3SEptO+Ude4UkXMBywIDAQABo0Mw
QTAdBgNVHQ4EFgQUYFhsLiklZh0riJ1Hg7d4HPcLlBUwCwYDVR0PBAQDAgGGMBMG
```

```
A1UdJQQMMAoGCCsGAQUFBwMJMA0GCSqGSIb3DQEBBQUAA4GBADU4aQ6f8pHWLd7z
vZ8pJ8e8UCvKok1LmdXbax5TBonyyLmb7AjLrOWjZ7LKSufJL1KOBsetd5Q49LFK
h70V2fRWpGNQszpAV60WfidkNvQ0koZczEjYRQOCtMDUqxMHxsMv2MLEVE9QuGLt
+NWjeeF03E1DT3C4mnbVsTyWPZij MAwwCgYIKwYBBQUHAwk=
-----END TRUSTED CERTIFICATE-----
```

Trusted certificates are added to a PKI certificate list the same as any other certificate using the **set pki certificate** command.

Use the **set pki oscp signature-ca-list** command to specify a list of trusted CA certificates used to verify OCSP response signatures.

Enabling the Nonce Extension

OCSP can be vulnerable to replay attacks, where a signed good response is captured by a malicious intermediary and replayed to the client at a later date after the subject certificate may have been revoked. OCSP overcomes this by including a nonce extension in the request that must be included in the corresponding response. If the corresponding OSCP response does not contain a matching nonce, the certificate verification will fail.

When OCSP nonce is enabled, the nonce extension is added to the outgoing OCSP request. If the corresponding OSCP response does not contain a matching nonce, then certificate verification will fail.

Use the **set pki ocsp nonce** command to enable or disable the inclusion of a nonce extension in the outgoing OCSP request that must be included in the corresponding response.

Configuring an Alternative OCSP Responder

X.509 certificates may contain an optional AIA extension which contains one or more addresses of OCSP Responders (OCSRs) to be used to check revocation status. In addition to these certificate OCSRs, one alternate OCSR URL may be configured. If this alternate responder is designated as preferred, then it will be tried before the certificate's AIA responders. If not preferred, then the alternate responder will be tried after the AIA responders.

Use the **set pki ocsp responder** commander to configure an alternate OCSP responder (OCSR) URL for the OCSR used to check revocation status.

Specifying a Single Authorization Username for the System

An X.509 certificate can contain information about the roles or privileges associated with the certificate. In practice an individual's responsibilities may change over time, and it is cumbersome to revoke and re-issue certificates each time this happens. The ability to specify a fixed global authorization username provides for mapping the certificate content to a local system user database or remote authentication protocol such as RADIUS. Once communication is established with the server requiring authentication, the user is interactively prompted for a password. The username and password combination is presented to the authorization server.

Use the **set pki authorization username** command to restrict the system to a single specified authorization credential which must be shared by all users.

The username can also be specified as an attribute that dynamically extracts the username from the subject field of the X.509 certificate. The **set pki authorization username attribute** command is used to specify an attribute based username configuration.

Dynamically Extracting the Username from the X.509 Subject Field

Each user can have its own set of authorization credentials based upon a specified distinguished name attribute extracted from the X.509 certificate subject field. The distinguished name attribute can be specified as a long name, short name, or an OID. Table 12-1 lists a few examples of the supported distinguished name attributes.

Table 12-1 X.509 Subject Field Distinguished Name Attributes

Attribute	Long Name	Short Name	OID
Country Name	countryName	C	2.5.4.6
Organization Name	organizationName	O	2.5.4.10
Organizational Unit Name	organizationalUnitName	OU	2.5.4.11
Common Name	commonName	CN	2.5.4.3

The username can be prefixed with a fixed string. For example, if the distinguished name attribute is **Extremenetworks** and the specified prefix is **foo**, the extracted username will be **fooExtremenetworks**.

In some instances it may be desirable to use only a subset of the extracted attribute, rather than the entire attribute verbatim. The match option allows for the dynamic application of a regular expression to the extracted attribute. The matching character output is used as the username. The S-Series supports the Extended Regular Expression (ERE) regular expression format.

The username can be suffixed with a fixed string. For example, if the distinguished name attribute is **US**, and the specified suffix is **bar**, the extracted username will be **USbar**.

Use the **set pki authorization username attribute** command to configure a dynamic extracted username from the X.509 certificate subject field.

In the following example, the final ten digits of the CN portion of the certificate subject field along with the @army.mil portion of the RADIUS account user name will be used create a new RADIUS account user name.

The X.509 certificate subject field contains:

```
Subject: C=US, O=U.S. Government, OU=DoD, OU=PKI, OU=DISA,
CN=doe.jane.d.3100020770
```

The resulting RADIUS account user name:

```
3100020770@army.mil
```

To form the RADIUS account user name using the X.509 certificate CN portion of the subject field, enter the following command where:

- commonName = The user name attribute will be based upon the commonName portion of the subject field
- **[0-9]** = Match digits 0 thru 9
- **{10}** = Match ten of those digits
- **$** = Those digits must be at the end of the input
- **@army.mil** = Append @army.mil to the 10 digits

```
S Chassis(su)->set pki authorization username attribute commonName match
[0-9]{10}$ suffix @army.mil
```

Configuring Public-Key Infrastructure

This section provides a table of Public-Key Infrastructure default values and a procedure for configuring a Public-Key Infrastructure system.

Table 12-2 lists Public-Key Infrastructure default values.

Table 12-2 Default Public-Key Infrastructure Parameters

Parameter	Description	Default Value
OCSP certificate revocation checking	A function that determines whether the Certificate Authority (CA) revocation checking is enabled or disabled.	enabled
outgoing OCSP request nonce extension inclusion	Specifies whether the nonce extension is included in the outgoing OCSP request to guard against replay attacks.	enabled

Table 12-3 describes Public-Key Infrastructure configuration on the Extreme Networks S-Series devices. All set commands used to configure Public-Key Infrastructure can be entered in any command mode with admin privilege.

Table 12-3 Configuring PKI

Task	Command(s)
To add a PEM formatted certificate to a certificate list.	**set pki certificate** *pki-cert-list* [**no-confirm**]
To globally enable or disable OCSP certificate revocation checking.	**set pki ocsp** {**enable** | **disable**}
To specify a list of trusted CA certificates used to verify OCSP response signatures.	**set pki ocsp signature-ca-list** *pki-cert-list*
To enable or disable the inclusion of a nonce extension in the outgoing OCSP request that must be included in the corresponding response.	**set pki ocsp nonce** {**enable** | **disable**}
To configure an alternate OCSP responder (OCSR) URL for the OCSR used to check revocation status.	**set pki ocsp responder** *url* [**preferred**]
To restrict the system to a single specified authorization credential which must be shared by all users.	**set pki authorization username** *username*
To configure a dynamic extracted username from the X.509 certificate subject field.	**set pki authorization username attribute** *attribute* [**prefix** *prefix*] [**match** *expression*] [**suffix** *suffix*]

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

Terms and Definitions

Table 12-4 lists terms and definitions used in this PKI configuration discussion.

Table 12-4 PKI Configuration Terms and Definitions

Term	Definition
PKI certificate list	One or more X.509 certificates grouped together in a list.
X.509 certificate	A certificate issued by a certification authority that binds a public key to an organizational or common name or an alternative DNS-entry.

Table 12-4 PKI Configuration Terms and Definitions (continued)

Term	Definition
Online Certificate Status Protocol (OCSP)	An Internet protocol, defined in RFC 2560, used for obtaining the revocation status of an X.509 digital certificate.
OCSP certificate revocation	The ability of a Certificate Authority (CA) to revoke an issued certificate's authorization prior to the issued certificate's expiration date in such cases as a compromised user or CA or the issuing of a newer certificate.
Certificate Authority (CA)	The digital signing and publishing of a public key bound to a given user based upon X.509 certificate private key that provides trust to the user key.
OCSP Responder (OCSR)	An online entity that returns a signed response signifying that the specified certificate in the OCSP request is good, revoked, or unknown.
certificate authorization	The step in the login procedure after authentication that determines what the certificate owner is allowed to do.
certificate authentication	The verification of the user certificate's issuance chain back to the CA by the SSH server in order to determine whether the user is who they claim to be.

13

Tracked Object Manager Configuration

This document provides the following information about configuring the Tracked Object Manager on the Extreme Networks S-Series platform.

Using Tracked Object Manager in Your Network

The Tracked Object Manager provides the ability to track local and remote objects by means of tracked objects and probes. Tracked objects monitor the state of local entities, such as interfaces. Probes monitor the state of remote entries, such as host servers. Each tracked object or probe is a container governed by a set of rules and configurable attributes used to determine the state of a collection of one or more of their respective entities. The run-time states of the tracked objects and probe are either up or down. Other states display a configuration or initiation state.

The Tracked Object Manager provides its services to client applications. An application must register with the Tracked Object Manager in order to use its monitoring services. When a client application wants to follow the state of a local service, the application registers with a tracked object. If the client application wishes to determine the state of a remote server, it creates a probe session. The Tracked Object Manager is responsible for monitoring the state of the configured objects and reports back to the registered client applications when the state of these objects change.

Tracked Objects

Tracked objects monitor the state of different types of local entities. Currently, the **port-group** tracked object type is supported. The tracked object type dictates the rules the tracked objects obey in determining their own state. The tracked object contains the configuration of the local entities and the attributes that allow for rules which affect the state of the tracked object. Client applications register for tracked object state events and perform actions upon receiving state events for that tracked object from the Tracked Object Manager.

The port-group type of tracked object allows users to form port-groups. For example, the port-group tracked object is used by the Link-State application to monitor the state of an upstream port-group and execute an action depending on the state events received from the Tracked Object Manager for that tracked object.

Probes

State probes track the availability of a remote service by actively transmitting network packets to a specified remote host. Tracked Object Manager supports three probe protocol types:

- An ICMP probe that monitors a device, by sending an ICMP ping to the IP address the probe is assigned to.
- A UDP probe that is capable of port service verification, by sending the port a UDP packet and waiting for an ICMP "Port Unreachable" response if the port is down. A UDP probe can also be configured for Application Content Verification (ACV) if the remote server supports a protocol that responds to a UDP packet, such as the UDP Echo protocol.
- A TCP probe that is capable of port service verification, by monitoring the appropriate port for services such as HTTP, Telnet, SMTP, and FTP. A TCP probe can also be configured for ACV for the verification of a layer 7 (OSI model) application running on the server.

The rules and attributes defined by the probe dictate how and when to transmit a packet to a remote host or peer. Unlike tracked objects, a probe does not contain the configuration of the remote entity. Client applications do not register for probes — they create probe sessions. The application provides the IP address and port tuple when creating the session. A probe session, like a tracked object, informs client applications of state events. It is the responsibility of the application to take an action based on the event.

Timing probes gather packet timing measurement for protocol packets. This is a more specialized type of probe, and it does not provide state events. Instead, each request for a probe session provides the client application with the packet transmit and receive times. Currently, ICMP and UDP timing packets are supported. The ICMP timing probes utilize the ICMP echo/reply paradigm. UDP timing probes may be configured to use the UDP-echo or DNS protocol.

The ICMP and UDP probes have the ability to gather packet timing measurements instead of monitoring the state of remote entity. Thus, each request for a probe session provides the client application with the packet transmit and receive times.

Examples of client applications that use probes include:

- Policy Based Routing (PBR), which uses an ICMP probe to monitor a next hop IP address.
- Server Load Balancing (SLB), which monitors an LSNAT real server IP address using an ICMP ping, or a port using TCP or UDP port verification. SLB also can verify an application running on the real server by configuring a TCP or UDP probe for ACV.
- Transparent Web Cache Balancing (TWCB), which uses an ICMP probe to monitor a cache IP address, or TCP or UDP probes to perform port verification on the cache server.
- The Virtual Router Redundancy Protocol (VRRP), which uses an ICMP probe to monitor a critical IP interface.
- The IP service level agreement (SLA) application, which uses an ICMP or UDP timing probe session to collect packet timing information. This type of data collection provides the IP SLA application with the statistics it needs to perform its calculations.

Note: Prior to the S-Series Firmware Release 7.21, the tracked objects functionality was performed in policy based routing by the route map pinger feature, and the probe functionality was performed in SLB and TWCB by fail detection. Both route map pinger and the previous application based fail detection have been removed from the S-Series firmware and have been replaced by the Tracked Object Manager feature.

Scheduling

Probe Session Scheduling

Probe sessions are scheduled by the Tracked Object Manager's scheduler. The scheduler rate limits the number of sessions that run every second using the leaky bucket paradigm. The scheduler's current rate limit is 200 sessions per second.

The probe attributes control the operational mechanics of the scheduler. Each probe session takes several actions during the lifetime of a particular transaction. A probe session retrieving or releasing a system resource may cause a variation in its scheduling and it may also reduce the scheduler’s rate limit for that second interval.

In addition, the scheduler's rate limiter dampens the Tracked Object Manager's CPU utilization during periods of heavy load, with the possible effect of delaying any number of the session's transactions.

Collecting packet timing information requires a constant transmit rate for the ICMP echo requests. The Tracked Object Manager supports this, but enforces a restriction on the receive wait value for ICMP timing probes. The receive wait value CANNOT be larger than the transmit interval. You configure the transmit interval and receive wait values in milliseconds for the timing probe, but the regular ICMP, UDP, and TCP probe attributes are configured in seconds.

Tracked Object Scheduling

Tracked objects utilize the Tracked Object Manager's scheduler when the delay up or down attribute is non-zero. The scheduler acts as a countdown timer for informing the client applications of a tracked object's state change. When a tracked object's state changes, the Tracked Object Manager puts the entry on the scheduling queue for the time period dictated by the delay attribute (either up or down). If the tracked object's state reverts back to its previous state prior to the countdown timer expiring, the Tracked Object Manager does not inform the client applications of the intermittent state change.

State Probe Configuration

State probes monitor the state of a remote service by actively transmitting network packets to a specified remote host. To configure a state probe:

- Create the probe by specifying a probe name and type
- Optionally configure a description to be associated with this probe
- Optionally configure a domain, host or IPv6 address DNS query type to send with this state probe
- Optionally configure an IP address or domain name to verify the DNS query response
- Optionally modify the number of consecutive failed faildetect probes that will determine when the service is declared down
- Optionally modify the interval between faildetect probes
- Optionally modify the number of successful pass detection probes that will determine when a service marked as down will be declared up
- Optionally modify the interval between pass detection probes
- Optionally specify the ACV or DNS Layer 5 protocol to use with this state probe
- Optionally modify the length of time the Tracked Object Manager will wait for a response from the monitored service before declaring that a probe request failed

- For a TCP probe, optionally modify the open interval that sets how long the Tracked Object Manager should wait for the completion of the TCP 3-way handshake
- When configuring ACV on a TCP or UDP probe:
  - Set the request string that will initiate the ACV session on the server
  - Set the reply string that will validate the server response to the request string
  - If required by the protocol being monitored, configure a close string to close the session
- Enable the probe by placing it inservice

The three state probe protocols supported by the Tracked Object Manager are ICMP, UDP, and TCP. Probe parameters are configured in probe configuration mode. You enter probe configuration mode by creating the probe in global configuration mode, specifying the name of the probe and the probe protocol. If the specified probe already exists, Tracked Object Manager enters configuration command mode for the named probe.

The state probe protocol used determines the fail detection method(s) that are available for monitoring the remote service. The fail detection methods supported for monitoring a remote service are:

- Ping
- Port Service Verification
- Application Content Verification (ACV)

Probes that do not yet exist can be assigned to monitor a service, but fail detection will not occur until the probe is created.

Probe Parameters

Probe parameters are configurable by entering probe configuration mode from the global configuration mode.

Description

A probe description of up to 127 printable characters can be configured. If a space character is entered, the description must be enclosed by double quotes (“”). Probe descriptions display in the detailed version of the **show probe** command output.

Application Content Verification Parameters

The following content verification parameters can be set:

- **Request String** – A string used by ACV that the Tracked Object Manager sends to the remote server to initiate verification of an application.
- **Reply String** – A string used by the Tracked Object Manager to validate the server response to the ACV request string.
- **Close String** – A string used by ACV to close a session when required by the protocol.
- **Search-Depth** – The number of characters into the server response to search for the ACV reply string. The reply string must match entirely within the search-depth.

Fail Detection Parameters

The Tracked Object Manager uses fail detection to determine when a service that is currently declared up should be declared down. Fail detection parameters set:

- The number of consecutive failed probe attempts before Tracked Object Manager declares a remote service down
- The delay, in seconds, between probes to a remote service that is currently declared up

Pass Detection Parameters

The Tracked Object Manager uses pass detection to determine when a service that is down should be declared up. Pass detection parameters set:

- The number of consecutive successful probes to a service currently declared down before the Tracked Object Manager declares the service up
- The delay, in seconds, between probes to a service that Tracked Object Manager currently declares down

Common Pass/Fail Parameters

- The time, in seconds, the Tracked Object Manager waits for a response from the monitored service. If a response is received within that time, the attempt passed. If a response is not received within that time, the attempt failed.
- The time, in seconds, the Tracked Object Manager waits for the TCP 3-way handshake to complete. If the handshake completes within that time, the attempt passed. If the handshake does not complete within that time, the attempt failed.

Fail Detection Methods

The fail detection method used determines whether the probe verifies a service, port, or application. The local application determines which fail detection methods are supported.

Ping

A remote service can be configured for the ping failure detection method by setting the probe protocol to ICMP. The ping failure detection method can be used by all S-Series applications supported by the Tracked Object Manager.

Server Port Service Verification

Port service verification is used by LSNAT server load balancing and TWCB to assure that the remote server is up. LSNAT and TWCB configurations support the TCP and UDP probe protocols for port service verification.

TCP port service verification can be enabled on one or more real servers, in a server load balancing configuration, or cache servers, in a TWCB configuration. A connect request is sent out to the server port. If the connect request succeeds then the local application knows the remote server is up.

UDP port service verification can be enabled on one or more real servers, in a server load balancing configuration. LSNAT accomplishes this by sending a UDP packet with "\r\n" (Carriage Return / Line Feed) as data to the UDP port. If the server responds with an ICMP "Port Unreachable" message, it is concluded that the port is not active and the real server is reported as "DOWN". Otherwise, if the LSNAT local application does not get any response at all, it is assumed that the port is active and the server is reported as "UP". The lack of a response could also be the result of the server itself not being available and could produce an erroneous indication of the server being "UP". To avoid this when the probe protocol is UDP, an ICMP ping is used in combination with UDP to ensure that the real server is available. By default LSNAT sets up a faildetect ICMP probe 1 ($slb_default). Set up a faildetect UPD probe 2 to use in conjunction with the ICMP probe.

Application Content Verification

Application Content Verification (ACV) can be enabled on a port to verify the content of an application on one or more servers. ACV is a method of ensuring that the server is responding with the appropriate response given some known good request. By its nature, ACV is protocol-independent and is designed to work with any type of server that communicates via formatted ASCII text messages, including HTTP, FTP, and SMTP.

ACV can be configured on both TCP and UDP probes.

ACV Configured On a UDP Probe

UDP is a connectionless protocol. The UDP server must have a protocol capable of responding to a UDP ACV probe request, such as the UDP Echo protocol. In the case of the UDP Echo protocol, the response is an echo of the probe request sent to the server. In this case, the configured string of the expected reply from the server is the same as the configured request string.

ACV Configured On a TCP Probe

ACV works by sending a request to your application server and searching the response for a certain string. If it finds the string, the server is marked as Up. If the string is not found, the server is marked as Down.

For ACV verification of a TCP server application, you specify the following:

- A string that the router sends to the server. The string can be a simple HTTP command to get a specific HTML page, or it can be a command to execute a user-defined CGI script that tests the operation of the application.
- The reply that the application on each server sends back is used by the router to validate the content. In the case where a specific HTML page is retrieved, the reply can be a string that appears on the page, such as “OK”. If a CGI script is executed on the server, it should return a specific response (for example, “OK”) that the router can verify.

For example, if you sent the following string to your HTTP server, “HEAD / HTTP/1.1\\r\\nHost: www.extremenetworks.com\\r\\n\\r\\n”, you could expect to get a response of a string returned similar to the following:

```
HTTP/1.1 200 OK
Date: Tue, 9 Feb 2010 20:03:40 GMT
Server: Apache/2.0.40 (Red Hat Linux)
Last-Modified: Wed, 6 Jan 2010 13:56:03 GMT
ETag: “297bc-b52-65f942c0”
Accept-Ranges: bytes
Content-Length: 2898
```

You can search for a reply string of “200 OK”. This would result in a successful verification of the service.

Because ACV can search for a string in only the first 255 bytes of the response, in most HTTP cases the response will have to be in the packet's HTTP header (that is, you will not be able to search for a string contained in the web page itself).

Some protocols such as FTP or SMTP require users to issue a command to close the session after making the request. An ACV close string can be configured and sent by the Tracked Object Manager to the server to close the session.

Preset Default ICMP Probes

Tracked Object Manager allocates a probe entry for each client application. A preset default ICMP probe for each supported application exists when you boot your system, although not all client applications may use their default probe. You cannot modify or delete default ICMP probes.

How a default ICMP probe is handled depends upon the application the default probe is associated with. Default ICMP probes associated with non-server-based applications such as policy based routing and VRRP are manually applied. Default ICMP probes associated with server-based applications such as server load balancing and TWCB are auto-applied.

Use the **show probe default** command to display a list of the current default probes. Use the **show probe** *probe-name* **detail** command to display the description and attributes of the default probes.

Manually Applied Default ICMP Probes

Manually applied default ICMP probes are treated the same as an administratively created ICMP probe and are provided for your convenience, should the preset parameter values meet your needs.

The Policy Based Routing (PBR) default ICMP probe must be manually applied. Use the **route-map probe** command in global configuration mode to apply the PBR default ICMP probe (**$pbr_default**) to monitor the specified next hop IP address. When configuring a default ICMP probe, the probe cannot be specified by name. Use the **default** keyword when configuring the default route-map probe.

The following example configures the default **$pbr_default** ICMP probe to monitor IP address **125.50.25.1**:

```
S Chassis(su-config)->route-map probe 125.50.25.1 probe-name default
```

The VRRP default ICMP probe is used to monitor remote critical IP addresses. When configuring a default ICMP probe, the probe cannot be specified by name. The VRRP default probe is configured when the **remote** keyword is specified. Use the **vrrp critical-ip** command in interface configuration mode, specifying the **remote** keyword, to apply the VRRP default probe to a critical IP interface.

This example sets the internet facing IP address **20.20.20.2** on VLAN **20** as the critical-IP address for VRRP instance **1**, sets the decrement operational priority to **100** should the interface go down, and assigns the VRRP default probe **$vrrp_default** to monitor the interface:

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 20
S Chassis(rw-config-intf-vlan.0.20)->vrrp critical-ip 1 20.20.20.2 100 remote
probe-name $vrrp_default
S Chassis(rw-config-intf-vlan.0.20)->no shutdown
S Chassis(rw-config-intf-vlan.0.20)->
```

Auto-Applied Default ICMP Probes

Server load balancing and TWCB support the configuration of any combination of up to two ICMP ping, TCP, or UDP probes. When configuring multiple probes on a server-based application, the probe is configured as probe **one** or probe **two**. Whenever probe **one** is not administratively configured, probe **one** is auto-configured to the default ICMP probe for that server context. The **$slb_default** probe is auto-configured for probe **one** in a real server context. The **$twcb_default** probe is auto-configured for probe **one** in a cache server context.

The probe type setting allows you to set whether configured probes are active or inactive for a server context. The probe type setting does not change the probe configuration. When probe type is set to **probe**, the probe configuration for the server context is active; probes are sent to the server

in accordance with the configured settings. When probe type is set to **none**, the probe configuration is inactive; no probes are sent for the server context. The default probe type is **probe**.

Auto applied probes can be overwritten when configuring an administratively created probe, by specifying probe **one** in the appropriate server context.

In a server configuration context, probe configuration can be reset to factory default values by resetting fail detection for that server context. Resetting fail detection in a server configuration context:

- Sets the probe type to the default value of **probe**
- Sets the probe for probe **one** to the default probe for the server context
- Removes any configured probe configuration for probe **two**

Configuring a Probe for Policy Based Routing

The route-map manager supports the assigning of an ICMP probe to monitor a next hop IP address. The route-map facility uses the Tracked Object Manager to monitor the IP address, but the ICMP probe is not assigned to a specific route-map. If a next hop IP address is declared down, it is removed from the next hop selection process for all route-maps specifying this address as a next hop, until it is declared up again. The assigned ICMP probe will ping port 0 of the specified IPv4 or IPv6 address.

Use the **route-map probe** command in router configuration mode to assign an ICMP probe to monitor the specified next hop IP address. Create a probe, using the **probe** command. A default ICMP probe can not be specified by name. Use the **default** keyword to assign the default policy based routing ICMP probe.

This example shows how to create the ICMP probe **ICMP-PBR** and assign it to a route-map probe to monitor next hop IP addresses **101.10.1.252** and **2000::1301:0:21f:45ff:fe4d:8722**. The fail detection count is set to **5** attempts, and the fail detection interval is set to **5** seconds. The two assigned sessions are displayed:

```
S Chassis(su-config)->probe ICMP-PBR icmp
S Chassis(su-config-probe)->faildetect count 5 interval 5
S Chassis(su-config-probe)->inservice
S Chassis(su-config-probe)->exit
S Chassis(su-config)->route-map probe 101.10.1.252 probe-name ICMP-PBR
S Chassis(su-config)->route-map probe 2000::1301:0:21f:45ff:fe4d:8722 probe-name
ICMP-PBR
S Chassis(su-config)->show probe sessions

Client Codes: P-policy based routing, S-SLB, V-VRRP, W-TWCB
              T-tracked object probe
...
Probe: ICMP-PBR, icmp
IP Address                        Port  Status    StChngs Last Change   Clients
--------------------------------- ----- --------- ------- ------------- -------
101.10.1.252                      0     Up        1            0h0m30s  P
2000::1301:0:21f:45ff:fe4d:8722   0     Up        1            0h0m40s  P
Displayed 2 sessions
...
S Chassis(su-config)->
```

Configuring a Probe for Server Load Balancing

Server load balancing provides the ability to assign two probes to monitor a real server. ICMP probe monitoring of a real server occurs by default, using the predefined ICMP probe **$slb_default**, assigned to probe **one**. See "Preset Default ICMP Probes" on page 13-7 for preset default ICMP probe details.

Probes are assigned to a real server configuration using the **faildetect probe** command in real server configuration mode. When assigning a probe to a real server, specify probe **one** or **two**, and the name of the probe. Any preexisting probe is overwritten when assigning a probe.

Default ICMP probes can not be assigned by specifying the name of the probe. When probe **one** has not been administratively configured, the default ICMP probe for that server context is auto-configured for probe **one**.

Layer 7 real server applications can be verified by configuring a TCP or UDP probe with ACV.

This example shows how to:

- Create a TCP probe named **TCP-HTTP**
- Set the fail detection interval to **5** seconds
- Set the pass detection interval to **5** seconds
- Configure the ACV request and reply strings
- Place the probe inservice
- Display a detailed level of configuration information for the probe
- Assign the probe to probe **one** of the **10.1.2.3** port **80** real server in the server farm **myproductHTTP**:
- Enable the real server configuration

```
S Chassis(su)->configure
S Chassis(su-config)->probe TCP-HTTP tcp
S Chassis(su-config-probe)->faildetect interval 5
S Chassis(su-config-probe)->passdetect interval 5
S Chassis(su-config-probe)->acv request "GET / HTTP/1.1\\r\\nHost:
2.0.0.5\\r\\n\\r\\n"
S Chassis(su-config-probe)->acv reply "HTTP/1.1 200 OK\\r\\n"
S Chassis(su-config-probe)->inservice
S Chassis(su-config-probe)->show probe TCP-HTTP detail
Probe:                        TCP-HTTP  Type:                           tcp-acv
Administrative state:        inservice  Session count:                        1
Fail-detect count:                   3  Pass-detect count:                   3
Fail-detect interval:                5  Pass-detect interval:                5
3-way TCP handshake wait time:       5  Server response wait time:           10
Application Content Verification:
 Request-string: GET / HTTP/1.1\\r\\nHost: 2.0.0.5\\r\\n\\r\\n
 Reply-string:   HTTP/1.1 200 OK\\r\\n
 Close-string:
 Search-Depth:   255
```

```
S Chassis(su-config-probe)->exit
S Chassis(su-config)->ip slb serverfarm myproductHTTP
S Chassis(su-config-slb-sfarm)->real 10.1.2.3 port 80
S Chassis(su-config-slb-real)->faildetect probe one TCP-HTTP
S Chassis(su-config-slb-real)->inservice
S Chassis(su-config-slb-real)->
```

Configuring a Probe for TWCB

TWCB provides the ability to assign two probes to monitor a cache server. ICMP probe monitoring of a cache server occurs by default, using the predefined ICMP probe **$twcb_default**, assigned to probe **one**. See "Preset Default ICMP Probes" on page 13-7 for preset default ICMP probe details.

Probes are assigned to a cache server configuration using the **faildetect probe** command in cache server configuration mode. When assigning a probe to a cache server, specify probe **one** or **two**, and the name of the probe. Any preexisting probe is overwritten when assigning a probe.

Default ICMP probes can not be assigned by specifying the name of the probe. When probe **one** has not been administratively configured, the default ICMP probe for that server context is auto-configured for probe **one**.

Layer 7 real server applications can be verified by configuring a TCP probe for application content verification.

This example shows how to:

- Create a TCP probe named **TCP-HTTP**
- Configure the ACV request and reply strings
- Place the probe inservice
- Display a detailed level of configuration information for the probe
- Assign the probe to probe **one** of the **186.89.10.51** cache server on the TWCB server farm **s1Server**:
- Assign port **8080** as the TCP port to be monitored.
- Enable the real server configuration

```
S Chassis(su)->configure
S Chassis(su-config)->probe TCP-HTTP tcp
S Chassis(su-config-probe)->inservice
S Chassis(su-config-probe)->acv request "GET / HTTP/1.1\\r\\nHost:
2.0.0.5\\r\\n\\r\\n"
S Chassis(su-config-probe)->acv reply "HTTP/1.1 200 OK\\r\\n"
S Chassis(su-config-probe)->show probe TCP-HTTP detail
Probe:                          TCP-HTTP  Type:                              tcp-acv
Administrative state:         inservice   Session count:                         1
Fail-detect count:                    3   Pass-detect count:                     3
Fail-detect interval:                 5   Pass-detect interval:                  5
3-way TCP handshake wait time:        5   Server response wait time:            10
Application Content Verification:
 Request-string: GET / HTTP/1.1\\r\\nHost: 2.0.0.5\\r\\n\\r\\n
 Reply-string:   HTTP/1.1 200 OK\\r\\n
```

```
 Close-string:
 Search-Depth:   255
S Chassis(su-config-probe)->exit
S Chassis(su-config)->ip twcb wcserverfarm s1Server
S Chassis(config-twcb-wcsfarm)->cache 186.89.10.51
S Chassis(config-twcb-cache)->faildetect probe one TCP-HTTP
S Chassis(config-twcb-cache)->faildetect app-port 8080
S Chassis(config-twcb-cache)->inservice
S Chassis(config-twcb-cache)->
```

Configuring a Probe for VRRP

VRRP supports the assigning of an ICMP probe to monitor a remote VRRP critical IP address. If an administratively configured probe name is not specified when configuring a remote critical IP address, the default VRRP ICMP probe, **$vrrp_default** is auto-configured to monitor the remote critical IP address. See "Preset Default ICMP Probes" on page 13-7 for default ICMP probe details.

This example:

- Creates the **ICMP-VRRP** ICMP probe
- Sets the fail detection and pass detection intervals to **5** seconds
- Sets the internet facing IP address **20.20.20.2** on VLAN **20** as the critical-IP address for VRRP instance **1**
- Sets the decrement operational priority to **10** should the interface go down
- Assigns ICMP probe **ICMP-VRRP** to monitor the interface
- Enables the interface

```
S Chassis(su-config)->probe ICMP-VRRP icmp
S Chassis(su-config-probe)->faildetect interval 5
S Chassis(su-config-probe)->passdetect interval 5
S Chassis(su-config-probe)->inservice
S Chassis(su-config-probe)->exit
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 20
S Chassis(rw-config-intf-vlan.0.20)->vrrp critical-ip 1 20.20.20.2 10 remote
probe-name ICMP-VRRP
S Chassis(rw-config-intf-vlan.0.20)->no shutdown
S Chassis(rw-config-intf-vlan.0.20)->
```

Configuring State Probes

This section provides details for state probe configuration on S-Series products.

Table 13-1 lists state probe default values.

Table 13-1 Default Tracked Object Manager Parameters

Parameter	Description	Default Value
probe faildetect count	The consecutive number of failed attempts before the service is declared down.	3 probes
probe faildetect interval	The delay in seconds between probes to a service that is up.	10 seconds
probe passdetect count	The consecutive number of successful probes to a service marked as down before the service is declared up.	3 probes
probe passdetect interval	The delay between probes to a service marked as down.	300 seconds
probe state	The service state of a configured probe.	not-in-service
receive interval	The time, in seconds, the Tracked Object Manager waits for a response from the monitored service before declaring a failed probe.	10 seconds
search depth	The number of characters into the server response to search for the ACV reply string.	255 characters
SLB faildetect probe one and two	Default probe for server load balancing faildetect probe one and two.	probe one: $slb_default probe two: empty
SLB faildetect type	The default probe behavior for this real server configuration.	probe; fail detection is active
TCP 3-way handshake interval	The interval, in seconds, the track object manager waits for the 3-way handshake to complete.	5 seconds
TWCB faildetect application port	The default TWCB faildetect application port	80
TWCB faildetect probe one and two	Default probe for TWCB faildetect probe one and two.	probe one: $twcb_default probe two: empty
TWCB faildetect type	The default probe behavior for this TWCB cache server.	probe; fail detection is active

Procedure 13-1 describes how to configure state probes. Refer to the "Tracked Object Manager Commands" chapter in the S-Series *CLI Reference* for more information about these commands.

Procedure 13-1 State Probe Configuration

Step	Task	Command(s)
1.	Create a probe, specifying the probe name and protocol type.	**probe** *probe-name* {**icmp** | **tcp** | **udp**}

Procedure 13-1 State Probe Configuration (continued)

Step	Task	Command(s)
2.	Optionally, configure a description to be associated with the probe.	**description** *description-text*
3.	Optionally, modify the number of consecutive failed faildetect probes that determine when the service is declared down.	**faildetect count** *count*
4.	Optionally, modify the interval between fail detection probes.	**faildetect interval** *seconds*
5.	Optionally, modify the number of successful pass detection probes that determine when a service marked as down will be declared up.	**passdetect count** *count*
6.	Optionally, modify the interval between pass detection probes.	**passdetect interval** *seconds*
7.	Optionally, specify the length of time the Tracked Object Manager waits for a response from the monitored service before declaring that a probe has failed.	**receive** *wait-interval*
8.	For a TCP probe, optionally modify the interval that sets how long the Tracked Object Manager waits for the completion of the TCP 3-way handshake.	**open** *wait-interval*
9.	When configuring ACV on a TCP or UDP probe, set the request string that will initiate the ACV session on the server.	**acv request** *request-string*
10.	When configuring ACV on a TCP of UDP probe, set the ACV validation reply string the server responds to the request string with.	**acv reply** *reply-string*
11.	When configuring ACV on a TCP probe, if required by the monitored protocol, configure a close string to close the session.	**acv close** *close-string*
12.	In probe configuration mode, optionally specify a DNS query type	**dns-query type** {[**domain** [**ip** *ip-address* | *name*] | **host** *name* | **ipv6** *ipv6-address*]}
13.	In probe configuration mode, optionally specify a domand name or IP address that will be used to verify the DNS query response.	**dns-verify match** {**address** *ip-address* | **domain** *name*}
14.	In probe configuration mode, optionally specify a Layer 5 protocol to use with this probe.	**l5-type** {**acv** | **dns**}
15.	Enable the probe by placing it inservice.	**inservice**

Timing Probe Configuration

Timing probes gather packet timing measurements using protocol packets. The timing probe protocols currently supported by the Tracked Object Manager are ICMP and UDP.

Probe parameters are configured in probe configuration mode. You enter probe configuration mode by creating the probe in global configuration mode, specifying the name of the probe and

the probe protocol and type. If the specified probe already exists, Tracked Object Manager enters configuration command mode for the named probe.

Timing Probe Parameters

Timing probe parameters are configured in probe configuration mode using the following commands:

Table 13-2 Configuring Timing Probe Parameters

Command	Parameter Description	Default Value
description	A probe description of up to 127 printable characters can be configured. If a space character is entered, the description must be enclosed by double quotes (“”). Probe descriptions are displayed by the **show probe** *probe-name* **detail** command.	None
dns-query type	This command allows you to specify a domain name, host name, or IPv6 address as the DNS query type to be sent with the probe.	None
l5-type	This command allows you to specify that the ACV or DNS Layer 5 protocol will be used with the probe.	None
interval	This command sets the transmit rate of ICMP echo requests. The transmit rate interval must be larger than the length of time specified to wait for an ICMP echo reply (the receive wait time). The system will not allow you to set a transmit interval less than or equal to the receive wait time, so if necessary, change the receive parameter first, before changing the interval value.	2000 milliseconds
packet-options	This command allows you to set the IP type of service or VLAN priority code point value to be included in the ICMP echo requests sent.	Both ToS and PCP are set to 0
receive	This command sets the length of time to wait for an ICMP echo reply. This receive wait time must be smaller than the transmit rate interval. The system will not allow you to set a receive wait time that is greater than or equal to the transmit rate interval, so if necessary, change the interval parameter before changing the receive wait time.	1000 milliseconds
inservice	This command places the probe in service. A scheduled IP SLA entry requires the assigned timing probe to be in service; otherwise, the IP SLA tests do not return data.	out of service

Configuring a Timing Probe for IP SLA

IP SLA requires an ICMP timing probe to test the connection to a destination IP address. You can assign the default ICMP timing probe, **$ipsla_default**, which uses the default ICMP timing probe settings, or you can create and assign a custom ICMP timing probe.

This example:

- Creates the **ICMP-IP-SLA** ICMP timing probe
- Sets the receive wait time to **500** milliseconds

- Sets the transmit interval to **1000** milliseconds
- Puts the ICMP timing probe into service
- Sets **ICMP-IP-SLA** as the ICMP timing probe for IP SLA tests on the destination IP address **125.50.25.1**

```
S Chassis(su-config)->probe ICMP-IP-SLA icmp timing
S Chassis(su-config-probe)->receive 500
S Chassis(su-config-probe)->interval 1000
S Chassis(su-config-probe)->inservice
S Chassis(su-config-probe)->exit
S Chassis(su-config)->sla entry 1 echo
S Chassis(su-config-sla)->destination 125.50.25.1 probe ICMP-IP-SLA
```

To run the IP SLA entry, you must schedule it. For information about scheduling the IP SLA entry, see Chapter 16, **IP SLA Configuration**.

Procedure

Procedure 13-2 lists the steps to configure an ICMP or UDP timing probe. Refer to the "Tracked Object Manager Commands" chapter in the S-Series CLI Reference for details about using these commands.

Procedure 13-2 Timing Probe Configuration

Step	Task	Command(s)
1.	In global configuration mode, create the timing probe and enter probe timing configuration mode.	**probe** *probe-name* {**icmp** | **udp**} **timing**
2.	In probe timing configuration mode, optionally configure a description of the probe.	**description** *string*
3.	In probe timing configuration mode, optionally change the transmit interval from the default of 2000 milliseconds.	**interval** *millisecs*
4.	In probe timing configuration mode, optionally change the receive wait time from the default of 1000 milliseconds.	**receive** *millisecs*
5.	In probe timing configuration mode, optionally set the IP type of service or VLAN priority code point value to be included in the ICMP echo requests sent.	**packet-options** {**ip-tos** *tos* | **vlan-pcp** *pcp*}
6.	In probe timing configuration mode, optionally specify a DNS query type	**dns-query type** {[**domain** [**ip** *ip-address* | *name*] | **host** *name* | **ipv6** *ipv6-address*]}
7.	In probe configuration mode, optionally specify a Layer 5 protocol to use with this probe.	**l5-type** {**acv** | **dns**}

Procedure 13-2 Timing Probe Configuration (continued)

Step	Task	Command(s)
8.	In probe timing configuration mode, put the probe inservice.	**inservice**
9.	Display information about the probe in any command mode.	**show probe** [*probe-name* [**detail** | **session**]] **show probe sessions** **show probe default**

Tracked Object Configuration

Tracked objects monitor the state of different types of local entities. Currently, the **port-group** tracked object type is supported. The port-group tracked object allows you to monitor the line protocol status of a group of ports.

Tracked object parameters are configured in tracked object configuration mode. You enter tracked object configuration mode by creating the tracked object in global configuration mode, specifying the name of the tracked object and the tracked object type. If the specified tracked object already exists, Tracked Object Manager enters configuration command mode for the named object.

Tracked Object Parameters

Tracked object parameters are configured in tracked object configuration mode using the following commands:

Table 13-3 Configuring Tracked Object Parameters

Command	Parameter Description	Default Value
port	Valid for port-group tracked objects only. Specifies ports to be added to the port group for tracking.	None
threshold count	Valid for port-group tracked objects only. This command sets the port group threshold counts which control the up and down state of the port group tracked object.	
up	The tracked object changes to the “up” state if the number of “up” ports is greater than or equal to the **up** count value. Up count value can range from 1 to 255.	Up: 1
down	The tracked object changes to the “down” state when the number of “up” ports is less than or equal to the **down** count value. Down count value can range from 0 to 254. The down count must be smaller than the up count. The up count must be greater than the down count.	Down: 0
delay up | down	This command configures the amount of time for the Tracked Object Manager to wait prior to informing client applications of a state change, either “up” or “down”. Value can range from 1 to 180 seconds.	Up: 3 seconds Down: 3 seconds
description	A description of up to 127 printable characters can be configured. If a space character is entered, the description must be enclosed by double quotes (“”). Tracked object descriptions are displayed by the **show track** *track-name* **detail** command.	None

Table 13-3 Configuring Tracked Object Parameters (continued)

Command	Parameter Description	Default Value
inservice	This command enables the tracked object .	Disabled

Procedure

Procedure 13-3 on page 13-17 lists the steps to configure a tracked object. Refer to the "Tracked Object Manager Commands" chapter in the S-Series CLI Reference for details about using these commands.

Procedure 13-3 Port Group Tracked Object Configuration

Step	Task	Command(s)
1.	In global configuration mode, create a tracked object, specifying the type, and enter tracked object configuration mode.	**track** *track-name* **port-group**
2.	In port-group tracked object configuration mode, configure the ports to be included in the port-group object.	**port** *port-string*
3.	Optionally, in tracked object configuration mode, change the default delay values.	**delay** {[**up** *secs*]|[**down** *secs*]}
4.	Optionally, in port-group tracked object configuration mode, change the default threshold values.	**threshold count** {[**up** *count*]|[**down** *count*]}
5.	Optionally, in tracked object configuration mode, specify a description for the tracked object.	**description** *string*
6.	In tracked object configuration mode, enable the tracked object.	**inservice**
7.	Display information about the tracked object in any command mode.	**show track** [*track-name* [**detail**]]

Example

This example creates a port-group tracked object named ls_group, changes the defaults for up/down status message delays for the object, gives it a description, puts it in service, then configures the Link-State application to use that tracked object and associate it with downstream ports.

```
S Chassis(su)->configure
S Chassis(su-config)->track ls_group port-group
S Chassis(su-config-track-obj)->delay up 5 down 5
S Chassis(su-config-track-obj)->description "link-state group1"
S Chassis(su-config-track-obj)->port tg.2.1-4
S Chassis(su-config-track-obj)->inservice
S Chassis(su-config-track-obj)->exit
S Chassis(su-config)->exit
S Chassis(su)->show track ls_group detail

Description: link-state group1
Track ls_group
```

```
 Port-Group tg.2.1-4
  4 ports used [0 up, 4 down]
  threshold up 1, down 0
  speed 0 [aggregate]
 Status is Down
  2 changes, last change   2d18h44m01s
 Delay up 5 seconds, down 5 seconds
 Registrants:
  Link-state

Displayed 1 tracked objects
S Chassis(su)->set link-state track ls_group downstream ge.1.1-5
S Chassis(su)>show link-state ls_group detail

Link-state ls_group
 Ports
  Uplinks: tg.2.1-4
  Downlinks: ge.1.1-5
 State is Down, Last action shutdown downlinks
  3 state changes, last change   2d18h47m21s ago

Displayed 1 link-state entries
```

Terms and Definitions

Table 13-4 lists terms and definitions used in this Tracked Object Manager configuration discussion.

Table 13-4 Tracked Object Manager Terms and Definitions

Term	Definition
Tracked Object Manager	The Tracked Object Manager provides the ability to track local and remote objects by means of tracked objects and probes.
state probe	A probe of protocol type ICMP, UDP, or TCP that tracks the availability of a remote service, by actively transmitting network packets to a specified remote host.
timing probe	A probe of protocol type ICMP that gathers packet timing measurements for protocol packets. Timing probes do not provide state events. Instead, each request for a probe session provides the client application with the packet transmit and receive times.
tracked object	Tracked objects monitor the state of different types of local entities.
client application	An application that uses the objects provided by Tracked Object Manager
probe session	An entity consisting of an IP/port tuple created by a client application using a particular probe.
State Probe Terms	
server port verification	A state probe fail detection method used by server load balancing and TWCB to assure that the remote server is up.
application content verification (ACV)	A state probe fail detection method for the verification of application content on a server.
ICMP ping	A fail detection method that sends a ping packet to the IP address of the remote service.
default ICMP probe	A preset probe configured for each of the supported local applications.

Table 13-4 Tracked Object Manager Terms and Definitions (continued)

Term	Definition
DNS query type	A domain or host name or IPv6 address that is sent with the state probe.
DNS verify type	A domain name or IP address that is used to verify the DNS query response.
close string	A string used by ACV to close a session.
reply string	A string used by the Tracked Object Manager to validate the server response to the ACV request string.
request string	A string used by ACV that the local application sends to the remote server to initiate verification of an application.
search depth	A numeric value that specifies the number of characters to search within an ACV response for the ACV reply string.
faildetect count	The number of consecutive failed probe attempts before Tracked Object Manager declares a remote service down.
faildetect interval	The delay, in seconds, between probes to a remote service that is currently declared up.
probe one and two	Up to two probes, that can be a default probe or administratively created probe, labelled **one** and **two**, applied to a server context.
fail detection type	Specifies whether or not fail detection is active in the current server context.
L5-type	Specifies a ACV or DNS Layer 5 protocol to use with this state probe.
open interval	The time, in seconds, the Tracked Object Manager waits for the TCP 3-way handshake to complete.
passdetect count	The number of consecutive successful probe attempts to a service currently declared down before the Tracked Object Manager declares the service up.
passdetect interval	The delay, in seconds, between probes to a remote service that is currently declared down.
receive interval	The time, in seconds, the Tracked Object Manager waits for a response from the remote service before declaring a failed probe.
Timing Probe Terms	
DNS query type	A domain or host name or IPv6 address that is sent with the timing probe.
L5-type	Specifies a ACV or DNS Layer 5 protocol to use with this timing probe.
transmit interval	The transmit rate of ICMP echo requests. The transmit rate interval must be larger than the length of time specified to wait for an ICMP echo reply (the receive wait time).
receive wait time	The length of time to wait for an ICMP echo reply. The receive wait time must be smaller than the transmit rate interval.
Tracked Object Terms	
threshold counts	The port-group threshold counts which control the “up” and “down” state of the port group tracked object. Specifies the number of ports that must be in an up state for the port-group object to be considered up, and the number of ports that must be in a down state for the port-group object to be considered down.
delay	The amount of time for the Tracked Object Manager to wait prior to informing client applications of a state change, either “up” or “down”.

14

Bidirectional Forwarding Detection (BFD)Configuration

This chapter provides information about configuring and monitoring Bidirectional Forwarding Detection (BFD) on S-Series devices.

Using Bidirectional Forwarding Detection (BFD) in Your Network

Bidirectional Forwarding Detection (BFD) provides a mechanism for detecting a communications failure with a forwarding plane next hop in less than one second, independent of media and protocol. With high speed data rates, a failure requiring several seconds to detect results in the loss of a large amount of data. BFD augments the Hello mechanism of various routing protocols that have failure detection times greater than a second. Because routing protocol Hello mechanisms do not tend have the same timing mechanics, BFD also provides a network administrator with a consistent means of reacting to next hop status changes regardless of the routing protocol. BFD shares its primary goal of providing the up or down status of an adjacent system with the Tracked Object Manager. The Extreme Networks BFD solution integrates the BFD application into the Tracked Object Manager through the BFD probe type. This chapter refers to the BFD probe and its configuration. See Chapter 13, **Tracked Object Manager Configuration** for a complete discussion of tracked object probes.

RFC 5880 defines the BFD protocol. RFC 5882 defines BFD interaction with generic applications.

BFD operates in one of two operational modes:

- Asynchronous
- Demand

BFD defaults to the Asynchronous operational mode. When using Asynchronous mode, both peers send periodic Control packets to one another with an application added jitter that over time creates an Asynchronous relationship between the sending and receiving of Control packets.

When using Demand mode, the BFD session has another mechanism to determine if the neighbor is alive, and after an initial interval during which BFD functions in Asynchronous mode, instructs the neighbor to stop sending Control packets.

Both the Asynchronous and Demand modes can use the BFD Echo function. The BFD Echo function tests the forwarding plane by transmitting BFD Echo packets to the neighbor, with the neighbor routing the packet back to the sender via the interface in which the packet was received. The BFD session on the neighbor does not interact with the Echo packets. The Echo Function runs by default and is used in conjunction with a slow-timer, which reduces the frequency of transmitted Control packets from the neighbor to the BFD session. By default, the Echo function operates in conjunction with BFD Asynchronous mode, but can be used as an alternative to the Asynchronous operational mode by turning on Demand mode. The Echo function must be enabled in-order-to turn on Demand mode for the session.

There are minimum transmit and receive Control and Echo packet timing considerations when both Asynchronous mode and the Echo function are used in conjunction with each other. See RFC 5880 for a discussion of these considerations.

Note: Depending upon network configuration, performance numbers will vary. Extreme Networks recommends that you lab test BFD before deploying in a live environment.

Implementing BFD

To implement BFD:

1. Create a named BFD probe in global configuration mode.
2. Optionally modify the BFD probe Control packet parameters: minimum transmit interval, minimum receive interval, and Control packet multiplier.
3. Optionally disable the Echo function, if you do not want to transmit Echo packets.
4. Optionally modify the Echo packet parameters, minimum transmit interval, minimum receive interval, and Echo packet miss count.
5. Optionally enable Demand mode to disable Asynchronous operations and depend upon the Echo function for the detection of a communications failure on the neighbor.
6. Optionally provide a string of up to 127 characters to describe the BFD probe.
7. Optionally, modify the BFD slow timer feature to override the Control min-rx value when Echo mode is active.
8. Place the BFD probe in service.
9. Enable the default or named BFD probe for the interface configuration mode context in which it will operate.
10. In OSPF router configuration mode, direct OSPF to create BFD sessions for its neighbors for the specified interface or all interfaces.

BFD Configuration Overview

For information about...	Refer to page...
BFD in an OSPF Context	14-5

BFD Probe

The BFD feature uses a Tracked Object Manager BFD probe session to detect communications failures with the neighbor. BFD probe sessions are able to detect communications failures with a neighbor in less than a second and are not protocol dependent. The probe configured on the interface can be the default routing protocol probe ($rte_default) or a named BFD probe. The default routing protocol probe already exists and does not have to be created. See Chapter 13, **Tracked Object Manager Configuration** for a complete discussion of tracked object probes.

Use the **probe bfd** command, in global configuration mode, to create a named BFD probe and enter BFD probe configuration mode.

Once you have made any optional modification to the BFD session, place the probe inservice using the **inservice** command in BFD probe configuration mode.

This example enters configuration mode for a BFD probe named **bfdProbe1**:

```
S Chassis(su)->configure
S Chassis(su-config)->probe bfdProbe1 bfd
S Chassis(su-config-probe-bfd)->inservice
S Chassis(su-config-probe-bfd)->
```

BFD Operational Modes

BFD Asynchronous mode is not administratively enabled or disabled. BFD operates in Asynchronous mode any time Demand mode is disabled. When Demand mode is configured, BFD remains in Asynchronous mode for a specified length of time prior to enabling Demand mode on the BFD probe session. Once Demand mode is enabled on the BFD session probe, Control packets are no longer sent by the neighbor, A BFD probe session operating in Demand mode is dependent upon Echo packets for verification of the liveliness of a neighbor. Demand mode is disabled by default.

Use the **demand-mode** command, in BFD probe configuration command mode, to enable Demand mode for this BFD probe session after the specified interval in seconds.

This example shows how to enable Demand mode for BFD session **bfdProbe1** after the sessions has been up for **30** seconds:

```
S Chassis(rw)->configure
S Chassis(rw-config)->probe bfdProbe1 bfd
S Chassis(su-config-probe-bfd)->demand-mode 30
S Chassis(su-config-probe-bfd)->
```

Control Packet

When BFD is operating in Asynchronous mode, Control packets are exchanged between the BFD probe session and neighbor to verify communications on the link. Transmit and receive BFD session Control packet intervals can be set to different lengths, and Control packets use the multiplier.

The three configurable Control packet parameters are:

- Minimum transmit interval – Specifies the minimum interval in 50ms increments between the transmission of BFD Control packets.
- Minimum receive interval – Specifies the minimum interval in 50ms increments between received Control packets the BFD Control sessions can support.
- Detection Multiplier – Specifies the value multiplied by the negotiated transmit rate that produces the detection time. The peer will transition the BFD session to the down state if a control packet is not received within the detection time interval.

Use the **Control** command, in BFD probe configuration mode, to modify Control packet parameters.

This example shows how to set the minimum Control packet transmit and receive intervals to 350ms and the detection time multiplier to 5 for BFD probe bfdProbe1:

```
S Chassis(rw)->configure
S Chassis(rw-config)->probe bfdProbe1 bfd
S Chassis(su-config-probe-bfd)->control min-tx 350 min-rx 350 multiplier 5
S Chassis(su-config-probe-bfd)->
```

Echo Function

The Echo function is used to test the forwarding plane of the neighbor. Echo packets are formatted such that the neighbor's forwarding plane redirects the packet back to the sender. The Echo function is enabled by default, and may be used in both operating modes; asynchronous and demand. While in asynchronous mode, the BFD probe session overrides the minimum receive interval, using the slow-timer's configured value, to inform the neighbor to reduce the rate of control packet transmission. If the BFD probe session is in demand mode, the Echo function is the only means of determining the liveliness of the neighbor, as the neighbor does not transmit periodic control packets.

The Echo function provides three configurable attributes the user may modify in the BFD probe configuration mode:

- The minimum transmit interval, configurable in 50ms increments, defines how quickly the echo packets are transmitted
- The minimum receive interval, configurable in 50ms increments, informs the neighbor how quickly it may transmit echo packets
- The missed packet count determines how many transmitted echo packets may be lost before the BFD probe session transitions to the down state

Use the **no echo-mode** command, in BFD probe configuration mode, to disable the Echo function for BFD sessions.

This example shows how to disable the Echo function for BFD session **bfdProbe1**:

```
S Chassis(rw)->configure
S Chassis(rw-config)->probe bfdProbe1 bfd
S Chassis(su-config-probe-bfd)->no echo-mode
S Chassis(su-config-probe-bfd)->
```

Use the **echo** command, in BFD probe configuration mode, to modify Echo packet parameters for BFD sessions.

This example shows how to set the minimum transmit and receive intervals to 350ms and the minimum number of missed consecutive Echo packets to 5 for BFD probe bfdProbe1:

```
S Chassis(rw)->configure
```

```
S Chassis(rw-config)->probe bfdProbe1 bfd
S Chassis(su-config-probe-bfd)->echo min-tx 350 min-rx 350 miss-count 5
S Chassis(su-config-probe-bfd)->
```

Slow Timer

The BFD probe slow-timer attribute acts as an override for the control packet's minimum receive interval attribute. The slow-timer overrides the minimum receive interval when the Echo function is in use, as described in RFC 5880, section 6.8.3. The slow-timer has a range of 1000ms – 30000ms, with a default value of 2000ms, and must be configured in increments of 50ms.

Use the **slow-timer** command, in BFD probe configuration mode, to modify the minimum receive interval override interval when the Echo function is in use.

This example shows how to set the slow timer interval to **2500**ms for the **bfdProbe1** BFD probe:

```
S Chassis(rw)->configure
S Chassis(rw-config)->probe bfdProbe1 bfd
S Chassis(su-config-probe-bfd)->slow-timer 2500
S Chassis(su-config-probe-bfd)->
```

BFD in an OSPF Context

There are two aspects to configuring BFD in an OSPF context:

1. Enable the default or named BFD probe for the OSPF interface configuration mode context in which it will operate.
2. Direct OSPF to create BFD sessions for its neighbors for the specified interface or for all interfaces.

Use the **bfd probe** command, in interface configuration mode, to enable the BFD probe for that interface.

This example shows how to use the BFD probe bfdProbe1 on the VLAN 1 interface:

```
S Chassis(rw)->
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 1
S Chassis(rw-config-intf-vlan.0.1)->bfd probe bfdProbe1
```

Use the **bfd** command in OSPF router configuration mode to specify the interface or all interfaces on which BFD sessions will be created for its neighbors.

This example enters OSPF router configuration **1** and configures OSPF to create BFD sessions for its neighbors on interface VLAN **1000**:

```
S Chassis(su)->configure
S Chassis(su-config)->router ospf 1
S Chassis(su-config-ospf-1)->bfd interface vlan.0.1000
S Chassis(su-config-ospf-1)->
```

BFD with Graceful Restart

If multiple fabric cards are operational in a multi-slot system, it is possible to keep the BFD session up during a failover. In order to accomplish this, Echo function needs to be in use and the BFD Detection Time must be greater than the time it takes to elect a new master after failover.

For BFD to support graceful restart:

- Demand mode must be enabled on the BFD session neighbor to prevent the restarting local system from sending BFD control packets
- The BFD session neighbor must be using the Echo function.

Configuring BFD

This section provides a table of BFD default values and a procedure for configuring a BFD on your system.

Table 14-1 lists BFD default values.

Table 14-1 Default BFD Parameters

Parameter	Description	Default Value
Minimum Transmit Interval	The minimum time period in milli-seconds between the transmission of BFD Control or Echo packets	250ms
Minimum Receive Interval	The minimum time period in milli-seconds in which a local BFD session expects to receive a BFD Control or Echo packet.	250ms
Detection time Multiplier	Specifies the value multiplied by the negotiated transmit rate that produces the detection time, causing the peer to transition the BFD session to the down state if a control packet is not received within the detection time interval.	4 packets
Echo Packet Miss Count	The minimum number of consecutive Echo packets that can be missed before the BFD session transitions to down.	3
Demand mode	A BFD mode of operation which assumes that the Echo function rather than Asynchronous operations will verify liveliness on the interface.	Disabled
Echo mode	A BFD feature that tests the forwarding plane by transmitting BFD Echo packets to the remote peer, with the remote peer routing the packet back to the sender via the interface in which the packet was received.	Enabled
Slow Timer	A timer parameter that overrides the Control min-rx value when the Echo feature is active.	2000ms

Procedure 14-1 describes BFD configuration on the Extreme Networks S-Series devices.

Procedure 14-1 Configuring FEATURE

Step	Task	Command(s)
1.	Create a named BFD probe in global configuration mode.	**probe** *probe-name* **bfd**
2.	Optionally, in BFD probe configuration mode, modify the BFD probe Control packet parameters: minimum transmit interval, minimum receive interval, and Control packet multiplier.	**control** {**min-tx** *interval* | **min-rx** *interval* | **multiplier** *number*}

Procedure 14-1 Configuring FEATURE (continued)

Step	Task	Command(s)
3.	Optionally, in BFD probe configuration mode, disable the Echo function, if you are operating in Asynchronous mode.	**no echo-mode**
4.	Optionally, in BFD probe configuration mode, modify the Echo packet parameters, minimum transmit interval, minimum receive interval, and Echo packet miss count.	**echo** {**min-tx** *interval* | **min-rx** *interval* | **miss-count** *number*}
5.	Optionally, in BFD probe configuration mode, enable Demand mode to disable Asynchronous operations and depend upon the Echo function for the detection of a communications failure on the remote interface.	**demand-mode** *up-time*
6.	Optionally, in BFD probe configuration mode, provide a string of up to 127 characters to describe the BFD probe.	**description "***string***"**
7.	Optionally, in BFD probe configuration mode, modify the BFD slow timer feature to override the Control min-rx value when Echo mode is active.	**slow-timer** *interval*
8.	In BFD probe configuration mode, place the BFD probe in service.	**inservice**
9.	In interface configuration mode, enable the default or named BFD probe for the OSPF interface configuration mode context in which it will operate.	**bfd probe** {**default** | *probe-name*}
10.	In router configuration mode, direct OSPF to create BFD sessions for its neighbors for the specified interface or all interfaces.	**bfd** {**all-interfaces** | **interface** *interface-name*}

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

Terms and Definitions

Table 14-2 lists terms and definitions used in this security mode configuration discussion.

Table 14-2 FEATURE Configuration Terms and Definitions

Term	Definition
Bidirectional Forwarding Detection (BFD)	A mechanism that provides for detecting a communications failure with a forwarding plane next hop in less than one second, independent of media and protocol.
Asynchronous Mode	The default BFD operational mode for which both peers send periodic Control packets to one another with an application added jitter that over time creates an Asynchronous relationship between the sending and receiving of Control packets.
Demand Mode	A BFD operational mode for which the sending of Control packets is disabled because the local peer has another mechanism to determine if the remote peer is alive

Table 14-2 FEATURE Configuration Terms and Definitions (continued)

Term	Definition
Echo Mode	A BFD mechanism used by both the Asynchronous and Demand modes that tests the forwarding plane by transmitting BFD Echo packets to the remote peer, with the remote peer routing the packet back to the sender via the interface in which the packet was received.
Minimum Transmit Interval	The minimum interval in 50ms increments between the transmission of BFD Control or Echo packets.
Minimum Receive Interval	The minimum interval in 50ms increments between received Control or Echo packets the BFD session can support.
Multiplier	A numeric value multiplied by the negotiated transmit rate that produces the detection time.
Miss Count	The minimum number of consecutive Echo packets that can be missed before the BFD session transitions to down.

15

Link-State Configuration

This chapter describes how to configure a Link-State entry.

Using the Link-State Application in Your Network

You may have devices in your network that have failover capabilities that enhance network redundancy, but they require an action by the switch to which they are connected to trigger that functionality. The Link-State application provides a facility that triggers link loss on downstream links if the associated upstream links go down.

For example, the Link-State application facilitates use of the NIC adapter teaming capability of network servers. NIC adapter teaming constructs primary and secondary relationships with directly connected switches. The switches provide upstream links that connect to other devices that provide the required network access needed by the servers. If the upstream links for the primary relationship lose connectivity, the Link-State feature forces the shutdown of the downstream stream links, triggering the NIC's teaming functionality. The NIC use its secondary relationship to avoid losing data.

The Link-State application uses port-group tracked objects to monitor the state of the upstream links. The Link-State application associates with downstream links, while the upstream links are associated with tracked objects (part of the Tracked Object Manager functionality). If the Tracked Object Manager detects a state change with the upstream links, the Link-State application is informed. If the upstream links are down, the Link-State application brings down the link to the downstream ports, causing link loss. The downstream device reacts to this and initiates its failover capability. Similarly, if the upstream links are up, the Link-State application attempts to bring up the downstream links. There may be other protocols or applications in the system that prevent the link from coming up.

When the Link-State application needs to influence the operational state of the downstream ports, it sets their operational status to down. In order for this feature to function, you must enable the force link down feature with the **set forcelinkdown enable** command. You can display the cause for port operation status down with the **show port operstatuscause** command.

Configuring Link-State

Procedure 15-1 lists the steps to configure a Link-State entry. Refer to the S-Series *CLI Reference* for details about using the commands listed.

Procedure 15-1 Configuring Link-State Entries

Step	Task	Command(s)
1.	In configuration mode, create the tracked object for the desired uplink ports.	Refer to the “Tracked Object Manager Configuration” chapter.
2.	Create the Link-State entry that associates the tracked object with the desired downstream ports.	**set link-state track** *object-name* **downstream** *port-string*
3.	Optionally, display information about the Link-State entry.	**show link-state** [*object-name* [**detail**]]
4.	Enable force link down on the switch.	**set forcelinkdown enable**

This example creates a port-group tracked object named ls_group, changes the defaults for up/down status message delays for the object, gives the tracked object a description and puts it in service, then configures the Link-State application to use that tracked object and associate it with downstream ports. Force link down is then enabled on the switch.

```
S Chassis(su)->configure
S Chassis(su-config)->track ls_group port-group
S Chassis(su-config-track-obj)->delay up 5 down 5
S Chassis(su-config-track-obj)->description “link-state group1”
S Chassis(su-config-track-obj)->port tg.2.1-4
S Chassis(su-config-track-obj)->inservice
S Chassis(su-config-track-obj)->exit
S Chassis(su-config)->exit
S Chassis(su)->show track ls_group detail

Description: link-state group1
Track ls_group
 Port-Group tg.2.1-4
  4 ports used [0 up, 4 down]
  threshold up 1, down 0
  speed 0 [aggregate]
 Status is Down
  2 changes, last change   2d18h44m01s
 Delay up 5 seconds, down 5 seconds
 Registrants:
  Link-state

Displayed 1 tracked objects
S Chassis(su)->set link-state track ls_group downstream ge.1.1-5
S Chassis(su)>show link-state ls_group detail

Link-state ls_group
 Ports
  Uplinks: tg.2.1-4
  Downlinks: ge.1.1-5
 State is Down, Last action shutdown downlinks
  3 state changes, last change   2d18h47m21s ago

Displayed 1 link-state entries
S Chassis(su)>set forcelinkdown enable
```

16

IP SLA Configuration

This document provides the following information about configuring IP SLA entries on the Extreme Networks S-Series platform.

Using IP SLA in Your Network

Service level agreements (SLAs) between Enterprise IT departments and end-users provide service guarantees for business critical applications. These agreements require performance monitoring of the network on a continual basis. The IP SLA feature allows you to configure, schedule, and monitor end-to-end packet timing measurements.

Use these timing measurements to understand how each service is performing on you network and, if necessary, deploy additional network applications more effectively or troubleshoot existing applications.

IP SLA collects, aggregates, and provides the ability to store the statistics gathered from the timing measurements for each session request. IP SLA performs the session requests through the Tracked Object Manager's ICMP timing probe feature. The Tracked Object Manager transmits ICMP echo packets to the destination provided by IP SLA and reports the timing information back to IP SLA.

IP SLA uses the timing information to calculate round-trip delay. Additional information provided by the Tracked Object Manager indicates if a packet was lost or is out of order. IP SLA provides a small storage area to keep the timing information for statistical modeling of the network.

Constraints and Limitations

IP SLA uses the Tracked Object Manager's ICMP timing probe functionality to perform ICMP echo requests to capture timing information. The Tracked Object Manager throttles the amount of traffic it generates (128 ICMP echo requests per second, 64 UDP or TCP requests per second) to limit the amount of CPU used. Because the Tracked Object Manager is a shared resource, the throttle applies to all applications that use the Tracked Object Manager. The Tracked Object Manager further restricts statistic gathering probes to one request every tenth of a second (maximum of ten requests per seconds).

IP SLA allocates memory to support eight entries. The default functionality of the entry consumes one session from the Tracked Object Manager's resource pool during a scheduled test. The Tracked Object Manager has a limit of 2000 sessions. If you configure an IP SLA entry to monitor

paths, the number of sessions required for the entry increases by the product of the number of paths multiplied by the number of hops.

IP SLA also allocates 30,000 statistical entries, with each entry using approximately 240 bytes, for the distribution and history mechanisms. The statistical entry contains the data for the round-trip-time metric. The number of statistical entry resources required to start the test is dependent on the distribution count, number of paths and hops, and the number of history buckets. The calculation is as follows:

(distribution count + history buckets) * (paths * hops + 1)

For more information about the memory used by IP SLA entries and statistical entries, see "System Resources Affected by IP SLA" on page 16-3.

Monitoring Paths

The IP SLA **monitor** command allows you to probe up to eight hops along four different equal-cost paths to the destination host. Path monitoring is restricted to the abilities of traceroute, which uses UDP packets. IP SLA uses the traceroute program to determine the hops along the different equal-cost paths. The traceroute program runs after IP SLA creates the probe session for the destination host. When there is more than one equal-cost path to the destination, the traceroute program runs for each path sequentially. After the traceroute completes for each path, IP SLA creates probe sessions for up to eight hops along that path. Each of these probe sessions stores the round-trip-time data.

Scheduling Tests

Use the IP SLA schedule mode to start and stop tests. The schedule mode includes options to set the following:

- The start time of the test cycle
- The number of tests in the test cycle
- The duration of each test in the test cycle
- The interval between tests in the test cycle
- Whether the test cycle is repeated
- You can schedule the test to repeat and the delay between each test.

When the test starts, it starts a new statistical collection and, if configured, a history collection. In both cases, the IP SLA scheduler clears the old collection of all of its data. The scheduler then reserves all of the statistic entries it needs to perform the test. After the resources are reserved, the application creates the probe session with the Tracked Object Manager.

Reported Statistics

IP SLA calculates the round-trip time from the timing information provided in the packet. If the packet is late or out-of-order, IP SLA updates the corresponding counter for the entry and disregards the rest of the data.

If IP SLA needs to correlate this data into a distribution or history collection, the event is stored accordingly.

If an ICMP timing probe assigned to an IP SLA entry is placed out of service while the IP SLA entry is running, the IP SLA entry will not report statistics until the ICMP timing probe is placed back in service.

The statistics for each test is stored in an IP SLA entry collection, which you can view with the **show sla** commands. For more information, see "IP SLA Display Commands" on page 16-7.

The statistics are stored locally on each blade. If a failover occurs on the blade, the IP SLA data is lost.

Measurements

IP SLA uses the timestamps in the ICMP timing packet provided by the Tracked Object Manager to measure the round-trip delay.

- **EchoTxTime**—The timestamp added to the ICMP echo request just prior to calculating the IP checksum and transmitting the packet to its destination.
- **EchoRxTime**—The timestamp of the arrival time of the ICMP echo request at the destination. The Tracked Object Manager adds this timestamp to the ICMP echo reply prior to the ReplyTxTime timestamp.
- **ReplyTxTime**—The timestamp added to the ICMP echo reply just prior to calculating the IP checksum and transmitting the packet back to the source of the ICMP echo request.
- **ReplyRxTime**—The timestamp of the arrival time of the ICMP echo reply.

With the timing information, you can create a statistical model the network. For all measurements, the minimum, average, and maximum values are included in the output of any collection. Also included is the sum of the values for each category and the sum of squares. The sums of the squares partition the variance into manageable portions.

To calculate the standard deviation with the data provided, use the computing formula for standard deviation.

Standard deviation = $\sqrt{(sumsquared \div samples) - ((sum \div samples)^2)}$

The following are the round-trip delay calculations.

- Round-trip delay (RTD): **ReplyRxTime – EchoTxTime**
- Wire-time RTD (WRTD): ((**ReplyRxTime** – **EchoTxTime**) – (**ReplyTxTime** – **EchoRxTime**))
- Host-time RTD (HRTD): RTD – WRTD

The calculation of the round-trip delay is always available. If the **EchoRxTime** and **ReplyTxTime** are available, the wire-time and host-time round-trip delay are calculated.

System Resources Affected by IP SLA

IP SLA affects the following system resources:

- Memory—The memory consumed by IP SLA is approximately 8 Megabytes. The 30,000 statistical data entries consume a little over 7 Megabytes while the eight IP SLA entries consume less than 1 Megabyte.
- CPU—Through its use of the Tracked Object Manager, IP SLA may have a significant effect on CPU utilization. The number of probe sessions and frequency of ICMP echo requests sent on behalf of these sessions affect the CPU. The functioning of IP SLA and its scheduler does not have a large effect on the CPU, with the exception of displaying a large amount of data collected for an IP SLA entry.
- Storage and persistence—The information pertaining to the configuration is persistent and it is stored in nonvolatile storage. The timing statistics are stored in RAM and IP SLA DOES NOT distribute this data to the other line cards.

- Network bandwidth—If you configure path monitoring, IP SLA uses the traceroute program to determine the path to the destination host. Indirectly, IP SLA creates probe sessions with the Tracked Object Manager, which uses the network bandwidth to perform its tasks.

IP SLA Syslog Messages

IP SLA generates the following Syslog messages:

- **Queue Overflow**—Occurs when, due to heavy usage, IP SLA cannot queue session statistics data from the Tracked Object Manager.

  Logging Level 5 (warning): Queue overflow

  <timestamp> <prefix> [vrf.process] <action> IP SLA application: Timing statistics queue overflow. Data dropped %d.

  Example:

  ```
  <166>Jan 13 16:36:20 10.21.130.55 Ipsla [1.tTrckStats] IP SLA application:
  Timing statistics queue overflow.  Data dropped 5.
  ```

- **Scheduler Started**—Occurs after the boot process and indicates IP SLA will start scheduling tests. The IP SLA scheduler does not start until the Tracked Object Manager starts.

  Logging Level 7 (information): Scheduler started

  <timestamp> <prefix> [vrf.process] INFO: Scheduler Started

  Example:

  ```
  FRI NOV 16 10:13:12 2012 IPsla[1.tIpSlaSchd] INFO: Scheduler Started
  ```

- **Destination Not Configured**—Occurs when the IP SLA entry attempts to start a test, but the destination IP address was not configured.

  Logging Level 6 (notice): Destination not configured

  <timestamp> <prefix> [vrf.process] NOTICE: Scheduled entry <n> could not run. Check destination IP address.

  Example:

  ```
  FRI NOV 16 10:17:46 2012 IPsla[1.tIpSlaSchd] NOTICE: Scheduled entry 4 could
  not run.  Check destination IP address.
  ```

- **Probe Deletion**—Occurs if you delete an ICMP timing probe while IP SLA entries using that probe are running test cycles.

  Logging Level 6 (notice): Probe deletion

  <timestamp> <prefix> [vrf.process] NOTICE: Scheduler [entry <n>]: Probe <name> deleted, ending test cycle

  Example:

  ```
  FRI NOV 16 10:25:09 2012 IPsla[1.tIpSlaEtsc] NOTICE: Scheduler [entry 4]: Probe
  ICMP-TIMING deleted, ending test cycle
  ```

Configuring IP SLA

Once you have determined how to implement IP SLA on your S-Series device, the following sections will help you configure IP SLA.

Default Settings

This section provides details for IP SLA configuration on S-Series devices.

Table 16-1 lists IP SLA default values.

Table 16-1 Default IP SLA Values

Parameter	Description	Default Value
collections	The number of tests for which statistical data is kept. The default value clears the data prior to each test execution.	1 collection
distribution count	The number of distributions of measured statistics.	0 distributions
distribution interval	The length of a distribution.	25 milliseconds
history ageout	The amount of time before the application frees the resources accumulated for a history collection. A value of zero indicates the application does not free any of the resources.	0 minutes
history bucket	The number of timing information storage units in a history collection.	15 buckets
history samples	The number of samples to collect in a static-depth bucket.	16 samples
history interval	The length of a timed-depth bucket.	30 seconds
history collections	The number of history collections to maintain. The default value indicates that no storage information is kept.	0 collections
history collections wrap	Indicates whether the storage history wraps when the number of collections exceeds the value specified by **history collections**.	no
monitor path-count	The number of paths to keep track of.	0 paths
monitor hop-count	The number of hops to keep track of.	1 hop
duration	The length of a test.	30 seconds
repetitions	The number of times a test is executed.	1
recurrence	The time between test cycles.	0 seconds
frequency	The time between each test in the test cycle.	30 seconds

IP SLA Configuration Procedure

Procedure 16-1 describes how to configure an IP SLA entry. Refer to the "IP SLA" chapter in the S-Series *CLI Reference* for more information about these commands.

Procedure 16-1 IP SLA Configuration

Step	Task	Command(s)
1.	In configuration mode, create an IP SLA entry, specifying the entry number and the entry type. At this time, echo is the only supported entry type.	**config** **sla entry** *<1-8>* **echo**
2.	Set the destination and the ICMP timing probe to be used by the IP SLA entry. You can use the default ICMP timing probe or you can create an ICMP timing probe using the Tracked Object Manager. For more information about creating ICMP timing probes, see "Timing Probe Configuration" on page 13-13.	**destination** *IP-address* **probe** {**default** | *probe-name*} [**port** *port*]
3.	Set the number of statistical collections for the IP SLA entry.	**collections** *<1-10>*
4.	Set the distribution count and interval for the IP SLA entry.	**distribution** {[**count** *count*] [**interval** *milliseconds*]}
5.	Set the history collection information for the IP SLA entry.	**history ageout** *minutes* **history buckets** *buckets* { [**samples** *samples*] |[**interval** *seconds*] } **history collections** *collections* [**wrap**]
6.	Set the number of hops and paths to be monitored by the IP SLA entry.	**monitor** [**hop-count** *count*] [**path-count** *count*]
7.	Exit the IP SLA entry configuration mode	exit
8.	Enter the IP SLA schedule mode.	**sla schedule**
9.	Schedule the IP SLA entry.	**entry** *ip-sla-entry* { [**start** { { [**time** *<yyyy-mm-dd:hh.mm.ss>*] | [**now**] | [**after** *<5-300>*]}][**duration** *<30-3600>*] [**frequency** *<30-3600>*] [**recurrence** *<120-7776000>*] [**repetitions** *<1-10>*] [**reset**] | [**stop**] }

Example IP SLA Configuration

IP SLA requires an ICMP timing probe to test the connection to a destination IP address. You can assign the default ICMP timing probe, **$ipsla_default**, which uses the default ICMP timing probe settings, or you can create and assign a custom ICMP timing probe.

This example:

- Creates IP SLA entry 2
- Sets the destination IP address for the ICMP requests
- Sets the ICMP timing probe to the default ICMP timing probe, $ipsla_default
- Sets the number of collections to 2

- Sets the distribution count to 4 and the interval to 20 milliseconds
- Sets the history ageout time to 18 minutes
- Sets the history buckets to 10 buckets with 22 samples in each bucket
- Sets the history collections to 2 and enables collection wrapping
- Sets the hops and paths to monitor (8 hops, 2 paths)
- Exits the IP SLA configuration mode
- Enters the IP SLA schedule mode
- Schedules IP SLA entry 2 to start test cycles 60 seconds after the command is entered. Each test cycle includes five tests that are each 60 seconds long. The interval between the tests in the test cycle is set to 60 seconds. The next test cycle will begin one second after the current test cycle ends.

```
S Chassis(su-config)->sla entry 2 echo
S Chassis(su-config-sla)->destination 1.1.1.1 probe default
S Chassis(su-config-sla)->collections 2
S Chassis(su-config-sla)->distribution count 4 interval 20
S Chassis(su-config-sla)->history ageout 18
S Chassis(su-config-sla)->history buckets 10 samples 22
S Chassis(su-config-sla)->history collections 2 wrap
S Chassis(su-config-sla)->monitor hop-count 8 path-count 2
S Chassis(su-config-sla)->exit
S Chassis(su-config)->sla schedule
S Chassis(su-config-sla-sched)->entry 2 start after 60 recurrence 601 duration 60
repetitions 5 frequency 60
```

IP SLA Display Commands

Table 16-2 lists the IP SLA show commands.

Table 16-2 IP SLA Show Commands

Task	Command
To display configuration and schedule information for all IP SLA entries.	**show sla**
To display the schedule information for all IP SLA entries.	**show sla scheduler**
To display configuration and schedule information for a specific IP SLA entry.	**show sla entry** *entry number*
To display detailed configuration and schedule information for a specific IP SLA entry.	**show sla entry** *entry number* **detail**
To display statistical distribution data for a specific IP SLA entry. Use the **collection** option to filter the output.	**show sla entry** *entry number* **distribution** [**collection** *collection number* [**destination** | [**path** *path number* [**hop** *hop number*]]]]
To display statistical history data for a specific IP SLA entry. Use the **collection** option to filter the output.	**show sla entry** *entry number* **history** [**collection** *collection number* [**destination**] | [**path** *path number* [**hop** *hop number* [**bucket** *bucket number*]]]]]

Table 16-2 IP SLA Show Commands

Task	Command
To display statistical summary data for a specific IP SLA entry. Use the **collection** option to filter the output.	**show sla entry** *entry number* **summary** [**collection** *collection number* [**destination** | [**path** *path number* [**hop** *hop number*]]]]
To display the number of system resources left.	**show limits application sla-entry-data**

Refer to the *Extreme Networks S-Series CLI Reference* for a description of the output of each command.

17

Power over Ethernet Configuration

This chapter provides information about configuring and monitoring Power over Ethernet (PoE) on the S-Series devices.

Important Notice

This section applies only to PoE-equipped S-Series devices. Consult the *Hardware Installation Guide* shipped with your product to determine if it is PoE-equipped.

How to Use PoE in Your Network

PoE, defined in IEEE standards 802.3af and 802.3at, refers to the ability to provide 54 Vdc (for 802.3at) or 48 Vdc (for 802.3af) operational power through an Ethernet cable from a switch or other device that can provide a PoE-compliant port connection to a powered device (PD). Examples of PDs include:

- Voice over IP devices such as PoE-compliant digital telephones
- Devices that support Wireless Application Protocol (WAP) such as wireless access points and security cameras

Ethernet implementations employ differential signals over twisted pair cables. This requires a minimum of two twisted pairs for a single physical link. Both ends of the cable are isolated with transformers blocking any DC or common mode voltage on the signal pair. PoE exploits this fact by using two twisted pairs as the two conductors to supply a direct current to a PD. One pair carries the power supply current and the other pair provides a path for the return current.

Using PoE allows you to operate PDs in locations without local power (that is, without AC outlets). Having such a network setup can reduce the costs associated with installing electrical wiring and AC outlets to power the various devices.

Implementing PoE

You can configure PoE on your PoE-compliant Extreme Networks device through the CLI-based procedures presented in the section "Configuring PoE" on page 17-3. As part of your plan to implement PoE in your network, you should ensure the following:

- The power requirements of your PDs are within the limits of the PoE standards.

- Your PoE-compliant Extreme Networks device can supply enough power to run your PDs. See Table 17-1 for power ranges based on each device class.

Table 17-1 PoE Powered Device Classes

Class	Power Output at Port	Power Range Used by Device
0	15.4 watts	0.44 to 12.95 watts
1	4.0 watts	0.44 to 3.84 watts
2	7.0 watts	3.84 to 6.49 watts
3	15.4 watts	6.49 to 12.95 watts
4	Reserved (802.3af)	Treat as class 0 (802.3af)

If SNMP traps are enabled, the Extreme Networks device generates a trap to notify the network administrator if a power state occurs on a PD (for example, when a PD is powered up or unplugged)

If insufficient power is available for an attached PD, the corresponding port LED on the Extreme Networks device turns amber, when the port is in PoE mode. The LED also turns amber if a PoE fault occurs (for example, a short in the Ethernet cable).

Allocation of PoE Power to Modules

The switch firmware determines the power available for PoE based on hardware configuration, power supply status, and power supply redundancy mode. The system calculates and reserves the correct amount of power required by the installed hardware components and then makes the balance of power available for PoE. When any change is made to the hardware configuration, power supply status, or redundancy mode, the firmware recalculates the power available for PoE.

On the S-Series switch, you can manually configure the maximum percentage of PoE power available to the chassis as a percentage of the total installed PoE power with the **set inlinepower available** command. If the power needed or requested exceeds the power available, the system will generate a trap to notify the system manager, if traps are enabled.

The power available for PoE is distributed based on the configured allocation mode, set with the **set inlinepower mode** command:

- **Automatic** mode, in which available power is distributed evenly to PoE-capable modules based on PoE port count. (This is the default mode.) Any change in available power, due to a change in power supply status or redundancy mode or to the addition or removal of modules, will trigger an automatic redistribution of power.
- **Manual** mode, in which the power budget for each PoE-capable module is manually configured, using either CLI commands or the MIBs. The sum of the wattage configured for each module cannot exceed the total power available on the switch for PoE.

  The power budget for each PoE-capable module can be configured manually on the S-Series switch with the command **set inlinepower assigned**.

  The configured wattage assignments are used to calculate each slot's percentage of total available power. If the total available PoE power is reduced, a redistribution of available power will occur, applying the calculated percentages.

When Manual Mode is Configured

When manual distribution mode is configured, if a PoE module is added to the switch, the PoE power budget for existing modules will **not** be recalculated. The new module will have a power budget of zero until it is manually provisioned. Since the sum of the manually provisioned

wattages cannot exceed the total system power available, it may be necessary to adjust existing budgets to free up power for the new module.

When a PoE module is removed from a switch configured with manual power distribution mode, the PoE budget for each module will **not** be recalculated, based on the assumption that the module removed will be replaced with a new module that should receive the same amount of PoE power.

As noted above, if the total available PoE power is reduced, the power will automatically be redistributed based on applying the calculated percentages. If an additional PoE supply is installed, there is no impact on the assigned PoE since specific wattages have been assigned to each module. Only the "Total Power Detected" value will change. The extra PoE power, however, is available for further redistribution manually.

Management of PoE Power to PDs

For each PoE-capable module or switch, you can configure how its PoE controller makes power available to attached powered devices (PDs). On a per module basis, you can configure:

- **Real-time** mode, in which the PoE controller calculates the power needed by a PD based on the actual power consumption of the attached devices.
- **Class** mode, in which the PoE controller manages power based on the IEEE 802.3af/.3at definition of the class limits advertised by the attached devices. In this mode, the maximum amount of power required by a device in the advertised class is reserved for the port, regardless of the actual amount of power being used by the device.

Power management to PDs is configured with the command **set inlinepower management**. PoE classes are defined in Table 17-1 on page 17-2.

Configuring PoE

Once you have determined how to implement PoE on your S-Series device, the following sections will help you configure PoE.

Default Settings

Table 17-2 lists PoE parameters and their default values.

Table 17-2 Default PoE Parameter Values

Parameter	Description	Default Value
Total Power Available	The percentage of total power available that a chassis can withdraw from the total power detected.	100
Power Allocation Mode	The allocation mode for system power available for PoE.	auto

Table 17-2 Default PoE Parameter Values (continued)

Parameter	Description	Default Value
Power Trap Status	Whether an SNMP trap message is sent when the status of the chassis PoE power supplies or the PoE system redundancy changes.	disable
Usage Trhld	The PoE usage threshold on a module or a Standalone.	75%
PSE Trap Status	Whether an SNMP trap message is sent whenever the status of a module's ports changes, or whenever the module's PoE usage threshold is crossed.	disable
Mgmt Mode	The PoE management mode.	realtime
Admin Status	Whether PoE is enabled on the port.	auto
Priority	Which ports continue to receive power in a low power situation.	low
Power Limit	The maximum power, in milliwatts, allowed on a port.	15400 mW
Power Capability-Selection	The PoE mode selected for the port.	8023af

PoE Configuration Procedure

Procedure 17-1 describes how to configure PoE. Unspecified parameters use their default values.

Procedure 17-1 PoE Configuration

Step	Task	Command(s)
1.	Configure PoE parameters on ports to which PDs are attached. • **admin** — Enables (**auto**) or disables (**off**) PoE on a port. The default setting is **auto**. • **priority** — Sets which ports continue to receive power in a low power situation. If all ports have the same priority and the system has to cut power to the PDs, the PDs attached to the lowest numbered ports have the highest priority for receiving power. The default setting is **low**. • **type** — Associates an alias with a PD, such as "siemens phone." • **powerlimit** — Sets the maximum power, in milliwatts, allowed on a port. Valid values are 0–15400 for 802.3af and 0–34000 for 802.3at. How this parameter is set can affect the class of PD that can be attached to the port. • **capability** — Sets the PoE mode for the port to 8023af (15.4W maximum power) or 8023at (34.0W maximum power).	**set port inlinepower** *port-string* {[**admin** {**off** | **auto**}] [**priority** {**critical** | **high** | **low**}] [**type** *type*] [**powerlimit** *powerlimit*] [**capability** *capability*]}

Procedure 17-1 PoE Configuration (continued)

Step	Task	Command(s)
	Use the **clear** command to set the port's PoE parameters back to the default settings. • **admin** — auto • **priority** — low • **type** — null • **powerlimit** — 15400 • **capability** — 8023af	**clear port inlinepower** *port-string* {[**admin**] [**priority**] [**type**] [**powerlimit**] [**capability**]}
2.	(Optional) Enable an SNMP trap message to be sent when the status of the chassis PoE power supplies or the PoE system redundancy changes.	**set inlinepower powertrap** {**disable** | **enable**}
	Use the **clear** command to reset chassis power trap messaging back to the default state of disabled.	**clear inlinepower powertrap**
3.	(Optional) Enable an SNMP trap message to be sent whenever the status of a module's ports changes, or whenever the module's PoE usage threshold is crossed.	**set inlinepower psetrap** {**disable** | **enable**} *module-number*
	Use the **clear** command to reset PoE trap messaging for a module back to default state of disabled.	**clear inlinepower psetrap** *module-number*
4.	(Optional) Set the PoE usage threshold on a module. Valid values are 1–99 percent. If your S-Series device is a Standalone, specify 1 as the *module-number*.	**set inlinepower threshold** *usage-threshold module-number*
	Use the **clear** command to reset the PoE usage threshold on a specified module to the default value of 75 percent.	**clear inlinepower threshold** *module-number*
5.	(Optional) Set the percentage of total power available that a chassis can withdraw from the total power detected.	**set inlinepower available** *max-percentage*
	Use the **clear** command to reset the percentage of the total power available to a chassis to the default value of 100.	**clear inlinepower available**
6.	(Optional) Set the PoE management mode on a specified module. • **realtime** — Manages power based on the actual power consumption of the ports. • **class** — Manages power based on the IEEE 802.3af definition of the class upper limit for each attached PD. In this mode, the maximum amount of power required by a PD in the advertised class is reserved for the port, regardless of the actual amount of power being used by the device. If your S-Series device is an Standalone, specify 1 as the *module-number*.	**set inlinepower management** {**realtime** | **class**} *module-number*

Procedure 17-1 PoE Configuration (continued)

Step	Task	Command(s)
	Use the **clear** command to reset the PoE management mode on a specified module back to the default setting of **realtime**.	**clear inlinepower management** *module-number*
7.	(Optional) Configure the allocation mode for system power available for PoE. • **auto** — Available power is distributed evenly to PoE modules based on PoE port count. Any change in available power, due to a change in power supply status or redundancy mode or to the addition or removal of modules, triggers an automatic redistribution of power to the PoE controller on each PoE module. • **manual** —The power budget for each PoE module is configured manually, using the **set inlinepower assigned** command. The configured wattage assignments are used to calculate each module's percentage of total available power. If the total available PoE power changes, a redistribution of available power occurs, applying the calculated percentages.	**set inlinepower mode** {**auto** | **manual**}
	Use this command to reset chassis power allocation to the default mode of **auto**.	**clear inlinepower mode**
8.	(Only if the **set inlinepower mode** command is set to **manual**) Assign specific wattage to a PoE module. If the **set inlinepower mode** command is set to **manual**, you must assign power to each PoE module; otherwise, the module ports will not receive power. If the value set with this command is greater than the maximum power percentage specified with the **set inlinepower available** command, a warning will display in the **show inlinepower** output. If you execute these parameters, a ratio of assigned power is applied to each module. If your S-Series device is a Standalone, specify 1 as the *slot-number*.	**set inlinepower assigned** *power-value slot-number*
	Use the **clear** command to clear the power value manually assigned to one or more modules.	**clear inlinepower assigned** [*slot-number*]

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

Example PoE Configuration

An S-Series S3 chassis is configured as follows:

- Two 1200W PoE power supplies are installed in the PoE subsystem. Because one of the power supplies is redundant, the power available for PoE is 1200W.
- Two modules that support PoE are installed in the chassis in slot 1 and slot 2. PDs are connected to all 72 ports on the module in slot 1. No PDs are connected to the module in slot 2.

- System power available for PoE is allocated evenly—600W to each PoE module. With this configuration, there is not enough power available for all of the PDs connected to the module in slot 1.

To make power available for all the PDs connected to the module in slot 1, change the system's power allocation mode:

```
S3(su)->set inlinepower mode manual
```

Now, none of the 1200W available for PoE is assigned to the PoE modules. Assign the 1200W, or some portion of the 1200W to the PoE modules to power the attached PDs.

```
S3(su)->set inlinepower assign 1200 1
```

PoE Display Commands

Table 17-3 lists PoE show commands for S-Series devices.

Table 17-3 PoE Show Commands

Task	Command
Use this command to display PoE properties for a device.	**show inlinepower**
Use this command to display information about the ports that support PoE: • Type of PD attached (if specified) • Administrative and operational status • Priority • Class of PD attached • Power used by the PD	**show port inlinepower** [*port-string*]

Refer to the *Extreme Networks S-Series CLI Reference* for a description of the output of each command.

18

Discovery Protocol Configuration

This chapter provides information about configuring and monitoring discovery protocols on the S-Series devices.

How to Use Neighbor Discovery in Your Network

Neighbor discovery is the Layer 2 process in which a device identifies and advertises itself to its directly connected neighbors. Extreme Networks devices support the following neighbor discovery protocols:

- Link Layer Discovery Protocol (LLDP) and its extension, LLDP-MED, which is the IEEE 802.1AB standard for neighbor discovery
- Enterasys Discovery Protocol, for discovering Extreme Networks devices
- Cisco Discovery Protocol, for discovering Cisco devices

Neighbor discovery is useful for

- Determining an accurate physical network topology
- Creating an inventory of network devices
- Troubleshooting the network

LLDP, Enterasys Discovery Protocol, and Cisco Discovery Protocol are enabled on Extreme Networks devices by default. Though all three discovery protocols can run simultaneously, LLDP is the preferred protocol.

If a device, attached to a port that has been enabled for neighbor discovery, does not support LLDP but supports Enterasys Discovery Protocol or Cisco Discovery Protocol, then one of those protocols is used instead.

Understanding Neighbor Discovery

The neighbor discovery protocols support the Layer 2 process of network devices advertising their identities and capabilities on a LAN and discovering that information about their directly connected neighbors. While Enterasys Discovery Protocol and Cisco Discovery Protocol are vendor-specific protocols, LLDP is an industry standard (IEEE 802.1AB), vendor-neutral protocol.

The LLDP-enabled device periodically advertises information about itself (such as management address, capabilities, media-specific configuration information) in an LLDPDU (Link Layer Discovery Protocol Data Unit), which is sent in a single 802.3 Ethernet frame (see Figure 18-3 on page 18-6). An LLDPDU consists of a set of TLV (type, length, and value) attributes. The information, which is extracted and tabulated by an LLDP-enabled device's peers, is recorded in IEEE-defined management information base (MIB) modules, making it possible for the information to be accessed by a network management system using a management protocol such as SNMP. The information is aged to ensure that it is kept up to date. Ports can be configured to send this information, receive this information, or both.

The LLDP agent operates only in an advertising mode, and hence does not support any means for soliciting information or keeping state between two LLDP entities.

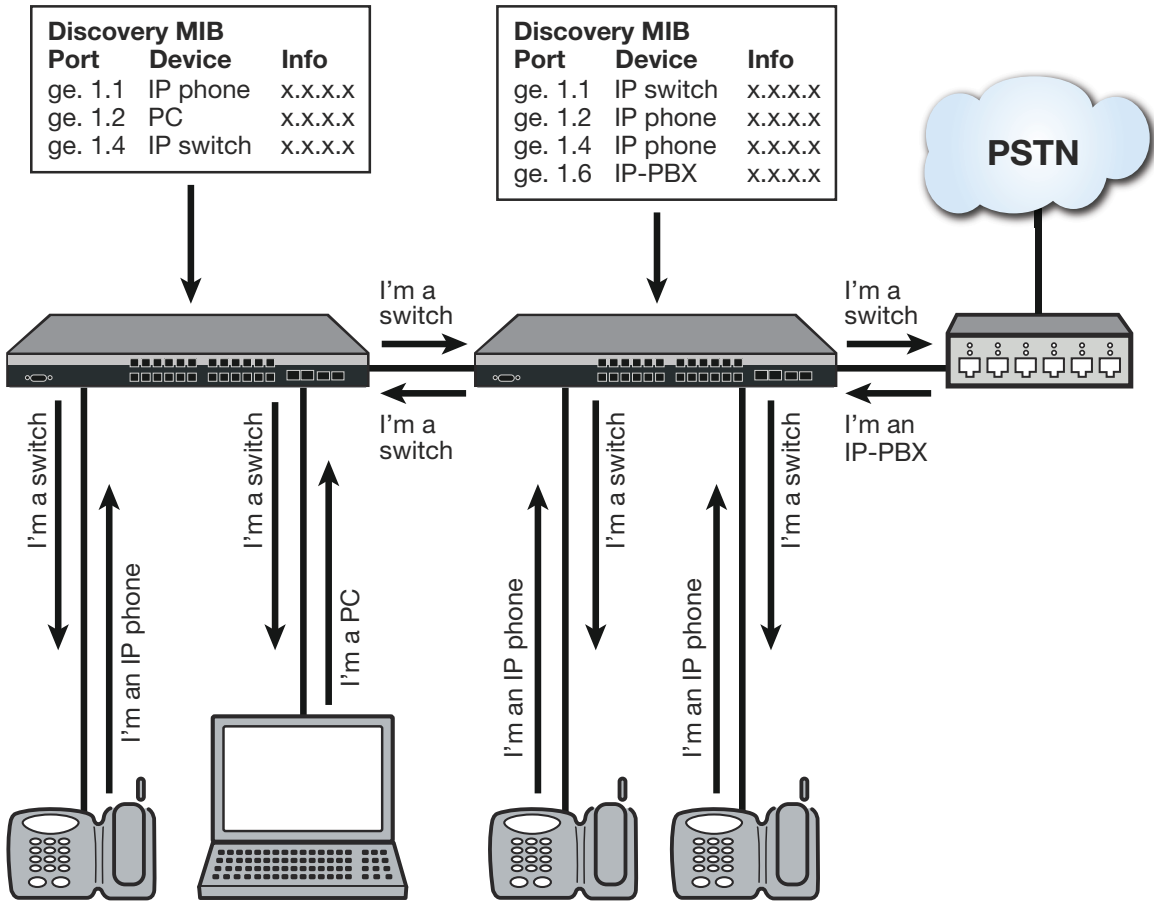
LLDP can be used for many advanced features in a VoIP network environment. These features include basic configuration, network policy configuration, location identification (including for Emergency Call Service/E911), Power over Ethernet management, and inventory management.

To fulfill these needs, the standard provides extensions to IEEE 802.1AB that are specific to the requirements of media endpoint devices in an IEEE 802 LAN. Interaction behavior between the media endpoint devices and the LAN infrastructure elements are also described where they are relevant to correct operation or multi-vendor interoperability. Media endpoint devices addressed include, but are not limited to, IP phones, IP voice/media gateways, IP media servers, and IP communication controllers.

The S-Series device supports a neighbor warning detection feature which enables protocol checking for a set of potential misconfigurations between this device and the neighbor port.

Figure 18-1 on page 18-3 shows an example of LLDP communication between devices, done via Layer 2 with LLDPDU packets. The communication is only between LLDP-enabled devices — the information is not forwarded to other devices.

Figure 18-1 Communication between LLDP-enabled Devices



LLDP-MED

The LLDP-Media Endpoint Discovery (LLDP-MED) extension of LLDP is defined to share information between media endpoint devices such as IP telephones, media gateways, media servers, and network connectivity devices.

Either LLDP or LLDP-MED, but not both, can be used on an interface between two devices. A switch port uses LLDP-MED when it detects that an LLDP-MED device is connected to it.

LLDP-MED provides the following benefits:

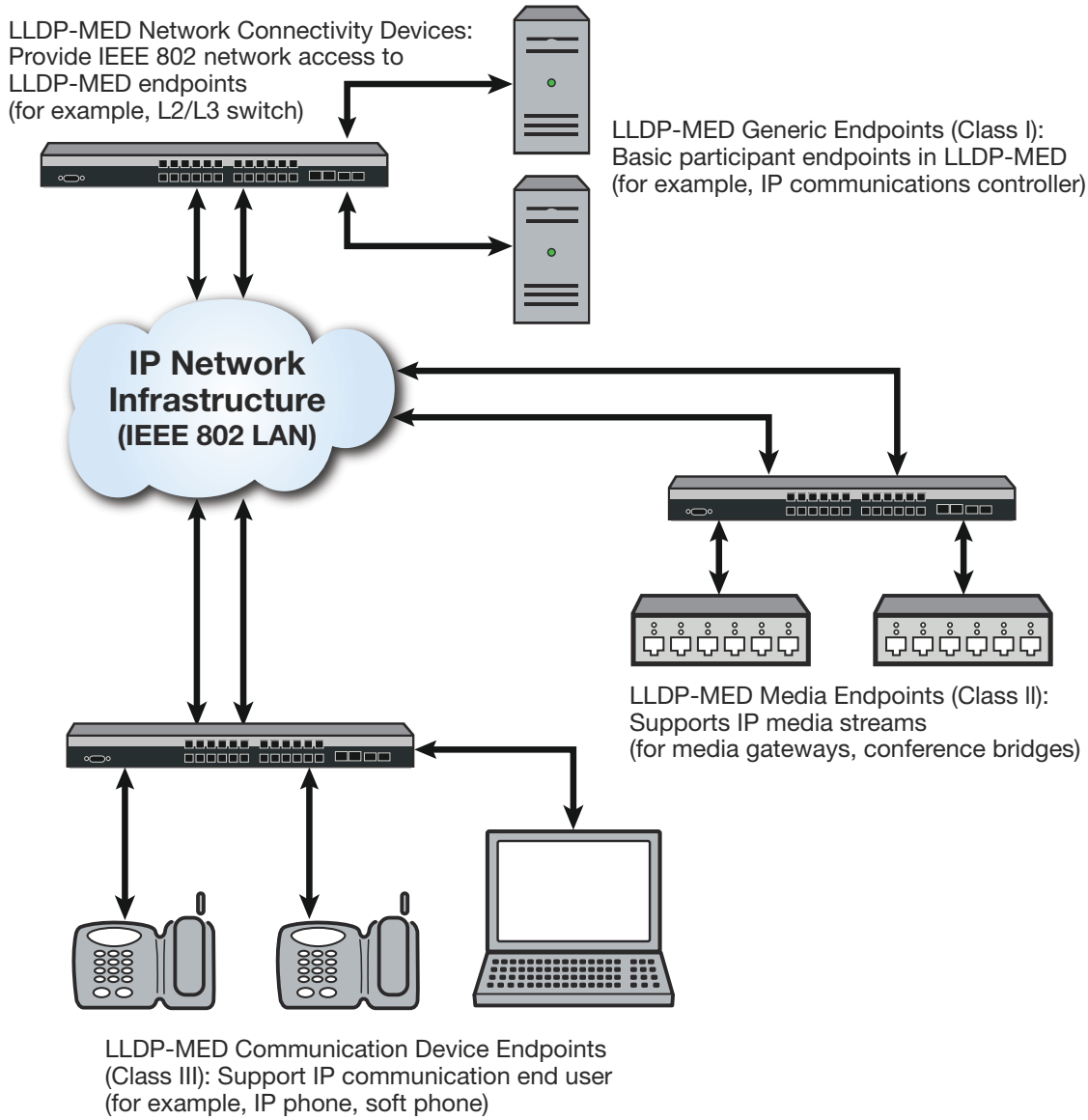
- Auto discovery of LAN policies, such as VLAN ID, 802.1p priority, and DiffServ codepoint settings, leading to plug-and-play networking.
- Device location and topology discovery, allowing creation of location databases and, in the case of VoIP, provision of E911 services.
- Extended and automated power management of Power over Ethernet endpoints
- Inventory management, allowing network administrators to track their network devices and to determine their characteristics, such as manufacturer, software and hardware versions, and serial or asset numbers.

There are two primary LLDP-MED device types (as shown in Figure 18-2 on page 18-5):

- Network connectivity devices, which are LAN access devices such as LAN switch/router, bridge, repeater, wireless access point, or any device that supports the IEEE 802.1AB and MED extensions defined by the standard and can relay IEEE 802 frames via any method.

- Endpoint devices, which have three defined sub-types or classes:
  - LLDP-MED Generic Endpoint (Class I) — All endpoint products that, while requiring the base LLDP discovery services defined in the standard, do not support IP media or act as an end-user communication device, such as IP communications controllers, other communication-related servers, or any device requiring basic services. Discovery services defined in this class include LAN configuration, device location, network policy, power management, and inventory management.
  - LLDP-MED Media Endpoint (Class II) — All endpoint products that have IP media capabilities but that may not be associated with a particular end user, such as voice/media gateways, conference bridges, and media servers. Capabilities include all of the capabilities defined for Generic Endpoint (Class I) and are extended to include aspects related to media streaming. Discovery services defined in this class include media type specific network layer policy discovery.
  - LLDP-MED Communication Endpoint (Class III) — All endpoint products that act as an endpoint user communication device supporting IP media. Capabilities include all of the capabilities defined for the Generic Endpoint (Class I) and Media Endpoint (Class II) devices and are extended to include aspects related to end user devices, such as IP phones, PC-based soft phones, and other communication devices that directly support the end user.

Figure 18-2 LLDP-MED

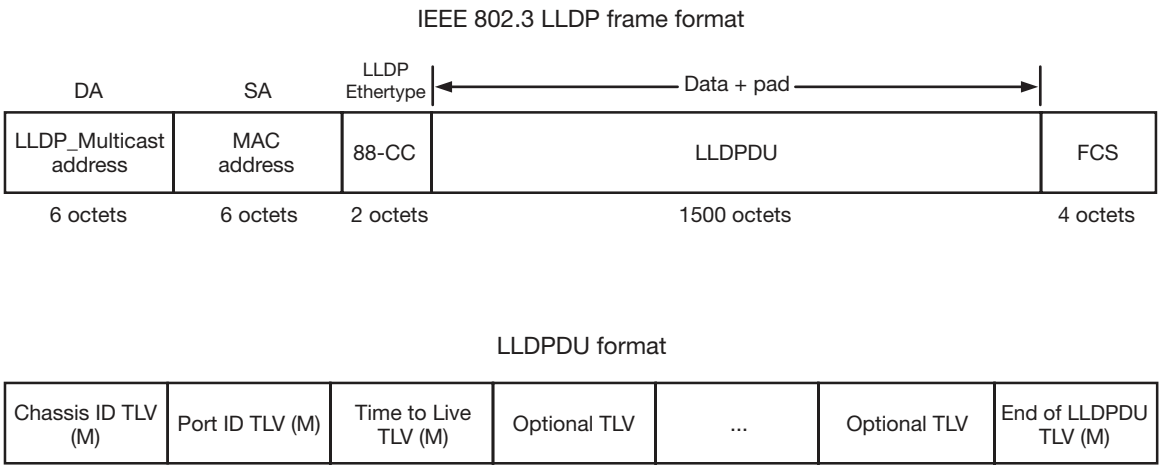


LLDPDU Frames

As shown in Figure 18-3, each LLDPDU frame contains the following mandatory TLVs:

- Chassis ID — The chassis identification for the device that transmitted the LLDP packet.
- Port ID — The identification of the specific port that transmitted the LLDP packet. The receiving LLDP agent joins the chassis ID and the port ID to correspond to the entity connected to the port where the packet was received.
- Time to Live — The length of time that information contained in the receive LLDP packet will be valid.
- End of LLDPDU — Indicates the final TLV of the LLDPDU frame.

Figure 18-3 Frame Format



M = Mandatory TLV (required for all LLDPDUs)

Each LLDPDU frame can also contain the following optional TLVs:

- Port Description — The port from which the LLDP agent transmitted the frame.
- System Name — The system's administratively assigned name.
- System Description — Includes the system's name, hardware version, OS level, and networking software version.
- System Capabilities — A bitmap that defines the primary functions of the system. The currently defined capabilities include, among other things, WLAN access point, router, and telephone.
- Management Address — The IP or MAC address associated with the local LLDP agent that may be used to reach higher layer entities.

An LLDPDU frame can also contain the following extension TLVs:

- 802.1 VLAN extension TLVs describe attributes associated with VLANs:
  - Port VLAN ID — Allows a bridge port to advertise the port's VLAN identifier (PVID) that will be associated with untagged or priority tagged frames it receives.
  - Port & Protocol VLAN ID — Allows a bridge to advertise whether it supports protocol VLANs and, if so, what VLAN IDs these protocols will be associated with.
  - VLAN Name — Allows a bridge to advertise the textual name of any VLAN with which it is configured.
  - Protocol Identity — Allows a bridge to advertise the particular protocols that are accessible through its port.
- 802.3 LAN interface extensions TLVs describe attributes associated with the operation of an 802.3 LAN interface:
  - MAC/PHY Configuration/Status — Advertises the bit-rate and duplex capability of the sending 802.3 node, the current duplex and bit-rating of the sending 802.3 node, and whether these settings were the result of auto-negotiation during link initiation or manual override.
  - Power-Via-MDI — Advertises the power-via-MDI capabilities of the sending 802.3 node.
  - Link-Aggregation — Advertises whether the link is capable of being aggregated, whether it is currently in an aggregation, and, if it is in an aggregation, the port of the aggregation.

  - Maximum Frame Size — Advertises the maximum supported 802.3 frame size of the sending station.

- LLDP-MED extension TLVs:
  - Capabilities — Indicates the network connectivity device's capabilities.
  - Network Policy — Used to configure tagged/untagged VLAN ID/L2 priority/DSCP on LLDP-MED endpoints (for example, IP phones).
  - Location Identification — Provides the location identifier information to communication endpoint devices, based on the configuration of the network connectivity device it is connected to.
  - Extended Power via MDI — Enables advanced power management between LLDP-MED endpoints and network connectivity devices.
  - Inventory Management — Includes hardware revision, firmware revision, software revision, serial number, manufacturer name, model name, and asset ID.

Some TLVs support multiple subtypes. For example, Port ID is sent as an ifName (e.g., ge.1.1) between Extreme Networks devices, but when an LLDP-MED endpoint is detected on a port, that TLV subtype changes to a network address (MAC address), and other MED TLVs are sent, as defined by the MED spec.

Neighbor Warning Detection

The S-Series device supports a neighbor warning detection feature which enables checking for a set of potential misconfigurations between this device and the neighbor port. Warning types are configurable on a per port basis. Generated warnings can be displayed using a warnings show command. Warning types support the detection of neighbor differences in:

- Speed and duplex
- Power class
- MTU
- LACP status
- PFC status

Configuring LLDP

LLDP Configuration Commands

Table 18-1 lists LLDP configuration commands. The table indicates which commands are device specific.

Table 18-1 LLDP Configuration Commands

Task	Command
Set the time, in seconds, between successive LLDP frame transmissions initiated by changes in the LLDP local system information. Default value is 30 seconds.	**set lldp tx-interval** *frequency*
Set the number of LLDP PDU packets sent when entering fast transmission state.	**set lldp tx-fast-count** *count*

Table 18-1 LLDP Configuration Commands (continued)

Task	Command
Set the frequency of LLDP PDU transmissions while in fast transmission state.	**set lldp tx-fast-interval** *frequency*
Set the time-to-live value used in LLDP frames sent by this device. The time-to-live for LLDPDU data is calculated by multiplying the transmit interval by the hold multiplier. The default value is 4.	**set lldp hold-multiplier** *multiplier-val*
Set the minimum interval between LLDP notifications sent by this device. LLDP notifications are sent when a remote system change has been detected. The default value is 5 seconds.	**set lldp trap-interval** *frequency*
Set the number of fast start LLDPDUs to be sent when an LLDP-MED endpoint device, such as a phone, is detected. Network connectivity devices transmit only LLDP TLVs in LLDPDUs until they detect that an LLDP-MED endpoint device has connected to a port. At that point, the network connectivity device starts sending LLDP-MED TLVs at a fast start rate on that port. The default value is 3.	**set lldp med-fast-repeat** *count*
Enable or disable transmitting and processing received LLDPDUs on a port or range of ports.	**set lldp port status** {**tx-enable** | **rx-enable** | **both** | **disable**} *port-string*
Enable or disable sending LLDP traps when a remote system change is detected.	**set lldp port trap** {**enable** | **disable**} *port-string*
Enable or disable sending an LLDP-MED trap when a change in the topology has been sensed on the port (that is, a remote endpoint device has been attached or removed from the port).	**set lldp port med-trap** {**enable** | **disable**} *port-string*
Configure LLDP-MED location information on a port or range of ports. Currently, only Emergency Call Services (ECS) Emergency Location Identification Number (ELIN) is supported. ELIN is a special phone number used to indicate location, and is assigned and associated with small geographies in the organization.It is one of the forms of identification that the location identification TLV provides.	**set lldp port location-info elin** *elin-string port-string*
Select the optional LLDP and LLDP-MED TLVs to be transmitted in LLDPDUs by the specified port or ports.	**set lldp port tx-tlv** {[**all**] | [**port-desc**] [**sys-name**] [**sys-desc**] [**sys-cap**] [**mgmt-addr**] [**vlan-id**] [**stp**] [**lacp**] [**gvrp**] [**mac-phy**] [**poe**] [**link-aggr**] [**max-frame**] [**med-cap**] [**med-pol**] [**med-loc**] [**med-poe**] [**enhanced-trans-config**] [**enhanced-trans-rec**] [**priority-flowctrl**]} *port-string*
Configure network policy for a set of applications on a port or range of ports. The policies configured with this command are sent in LLDPDUs as LLDP-MED Network Policy TLVs. Multiple Network Policy TLVs can be sent in a single LLDPDU.	**set lldp port network-policy** {**all** | **voice** | **voice-signaling** | **guest-voice** | **guest-voice-signaling** | **softphone-voice** | **video-conferencing** | **streaming-video** | **video-signaling**} [**state** {**enable** | **disable**}] [**tag** {**tagged** | **untagged**}] [**vid** {*vlan-id* | **dot1p**}] [**cos** *cos-value*] [**dscp** *dscp-value*] *port-string*
Return LLDP parameters to their default values.	**clear lldp** {**all** | **tx-interval** | **hold-multipler** | **trap-interval** | **med-fast-repeat**}

Table 18-1 LLDP Configuration Commands (continued)

Task	Command
Return the port status to the default value of both (both transmitting and processing received LLDPDUs are enabled).	**clear lldp port status** *port-string*
Return the port LLDP trap setting to the default value of disabled.	**clear lldp port trap** *port-string*
Return the port LLDP-MED trap setting to the default value of disabled.	**clear lldp port med-trap** *port-string*
Return the port ECS ELIN location setting to the default value of null.	**clear lldp port location-info elin** *port-string*
Return network policy for a set of applications on a port or range of ports to default values.	**clear lldp port network-policy** {**all** | **voice** | **voice-signaling** | **guest-voice** | **guest-voice-signaling** | **softphone-voice** | **video-conferencing** | **streaming-video** | **video-signaling**} {[**state**] [**tag**] [**vid**] [**cos**] [**dscp**] } *port-string*
Clear the optional LLDP and LLDP-MED TLVs to be transmitted in LLDPDUs by the specified port or ports to the default value of disabled.	**clear lldp port tx-tlv** {[**all**] | [**port-desc**] [**sys-name**] [**sys-desc**] [**sys-cap**] [**mgmtaddr**] [**vlan-id**] [**stp**] [**lacp**] [**gvrp**] [**mac-phy**] [**poe**] [**link-aggr**] [**max-frame**] [**medcap**] [**med-pol**] [**med-loc**] [**med-poe**]} *port-string*

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

Basic LLDP Configuration

Procedure 18-1 describes the basic steps to configure LLDP on Extreme Networks S-Series devices.

Procedure 18-1 Configuring LLDP (Extreme Networks S-Series)

Step	Task	Command(s)
1.	Configure global system LLDP parameters.	**set lldp tx-interval** **set lldp hold-multiplier** **set lldp trap-interval** **set lldp med-fast-repeat** **clear lldp**
2.	Enable/disable specific ports to: • Transmit and process received LLDPDUs • Send LLDP traps • Send LLDP-MED traps	 • **set/clear lldp port status** • **set/clear lldp port trap** • **set/clear lldp port med-trap**
3.	Configure an ECS ELIN value for specific ports.	**set/clear lldp port location-info**
4.	Configure Network Policy TLVs for specific ports.	**set/clear lldp port network-policy**
5.	Configure which optional TLVs should be sent by specific ports. For example, if you configured an ECS ELIN and/or Network Policy TLVs, you must enable those optional TLVs to be transmitted on the specific ports.	**set/clear lldp tx-tlv**

Example LLDP Configuration: Time to Live

This example sets the transmit interval to 20 seconds and the hold multiplier to 5, which will configure a time-to-live of 100 to be used in the TTL field in the LLDPDU header.

```
S Chassis(rw)->set lldp tx-interval 20
S Chassis(rw)->set lldp hold-multiplier 5
```

Example LLDP Configuration: Location Information

On an S-Series device, after you configure a location information value, you must also configure the port to send the Location Information TLV with the **set lldp port tx-tlv** command. This example configures the ELIN identifier 5551234567 on ports ge.1.1 through ge.1.6 and then configures the ports to send the Location Information TLV.

```
S Chassis(rw)->set lldp port location-info 5551234567 ge.1.1-6
S Chassis(rw)->set lldp port tx-tlv med-loc ge.1.1-6
```

LLDP Display Commands

Table 18-2 lists LLDP show commands. The table indicates which commands are device specific.

Table 18-2 LLDP Show Commands

Task	Command
Display LLDP configuration information.	**show lldp**
Display the LLDP status of one or more ports.	**show lldp port status** [*port-string*]
Display the ports that are enabled to send an LLDP notification when a remote system change has been detected or an LLDP-MED notification when a change in the topology has been sensed.	**show lldp port trap** [*port-string*]
Display information about which optional TLVs have been configured to be transmitted on ports.	**show lldp port tx-tlv** [**data-center-bridging**] [*port-string*]
Display configured location information for one or more ports.	**show lldp port location-info** [*port-string*]
Display the local system information stored for one or more ports.	**show lldp port local-info** [*port-string*]
Display the remote system information stored for a remote device connected to a local port.	**show lldp port remote-info** [*port-string*]
Display LLDP port network policy configuration information.	**show lldp port network policy** {**all** | **voice** | **voice-signaling** | **guest-voice** | **guestvoice-signaling** | **software-voice** | **video-conferencing** | **streaming-video** | **videosignaling**} [*port-string*]

Refer to the *Extreme Networks S-Series CLI Reference* for a description of the output of each command.

Configuring Neighbor Warning Detection

Table 18-3 lists neighbor warning detection configuration and display commands.

Table 18-3 Neighbor Warning Detection Commands

Task	Command
Enable protocol checking for port mis-configuration (warning detection) with its neighbor	**set neighbors warning-detection** *warning-type* [*port-string*] {**enable** | **disable**}
Display neighbors with warnings due to enabled warning detection hits on the port.	**show neighbors warnings** [*warning-type*] [*port-string*]
Display the status of each neighbors warning type for all or specified ports.	**show neighbors warning-detection** [*port-string*]

Configuring Enterasys Discovery Protocol

Enterasys Discovery Protocol Configuration Commands

Table 18-4 lists Enterasys Discovery Protocol configuration commands.

Table 18-4 Enterasys Discovery Protocol Configuration Commands

Task	Command
Enable or disable the Enterasys Discovery Protocol on one or more ports.	**set cdp state** {**auto** | **disable** | **enable**} [*port-string*]
Set a global Enterasys Discovery Protocol authentication code.	**set cdp auth** *auth-code*
Set the message interval frequency (in seconds) of the Enterasys Discovery Protocol.	**set cdp interval** *frequency*
Set the hold time value for Enterasys Discovery Protocol configuration messages.	**set cdp hold-time** *hold-time*
Reset Enterasys Discovery Protocol settings to defaults.	**clear cdp** {[**state**] [**port-state** *port-string*] [**interval**] [**hold-time**] [**auth-code**]}

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

Example Enterasys Discovery Protocol Configuration

This example shows how to globally enable CDP:

```
S Chassis(rw)->set cdp state enable
```

This example shows how to enable the CDP for port ge.1.2:

```
S Chassis(rw)->set cdp state enable ge.1.2
```

This example shows how to disable the CDP for port ge.1.2:

```
S Chassis(rw)->set cdp state disable ge.1.2
```

Enterasys Discovery Protocol Show Commands

Table 18-5 lists Enterasys Discovery Protocol show commands.

Table 18-5 Enterasys Discovery Protocol Show Commands

Task	Command
Display the status of the CDP discovery protocol and message interval on one or more ports.	**show cdp** [*port-string*]
Display Network Neighbor Discovery information from all supported discovery protocols.	**show neighbors** [*port-string*]

Refer to the *Extreme Networks S-Series CLI Reference* for a description of the output of each command.

Configuring Cisco Discovery Protocol

Cisco Discovery Protocol Configuration Commands

Table 18-6 lists Cisco Discovery Protocol configuration commands.

Table 18-6 Cisco Discovery Protocol Configuration Commands

Task	Command
Enable or disable Cisco Discovery Protocol globally on the device.	**set ciscodp status** {**auto** | **enable** | **disable**}
Set the number of seconds between Cisco Discovery Protocol PDU transmissions.	**set ciscodp timer** *time*
Set the time to live (TTL) for Cisco Discovery Protocol PDUs. This is the amount of time (in seconds) neighboring devices will hold PDU transmissions from the sending device.	**set ciscodp holdtime** *time*
Set the status, voice VLAN, extended trust mode, and CoS priority for untrusted traffic for the Cisco Discovery Protocol on one or more ports.	**set ciscodp port** { [**status** {**disable** | **enable**}] [**vvid** {<*vlan-id*> | **none** | **dot1p** | **untagged**}] [**trust-ext** {**trusted** | **untrusted**}] [**cos-ext** *value*] } <*port-string*>
Clear the Cisco Discovery Protocol back to the default values.	**clear ciscodp** { [**status** | **timer** | **holdtime** | **port** {**status** | **vvid** | **trust-ext** | **cos-ext**}] } <*port-string*>

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

Example Cisco Discovery Protocol Configuration

This example shows how to enable Cisco Discovery Protocol on the device:

```
S Chassis(rw)->set ciscodp status enable
```

Cisco Discovery Protocol Show Commands

Table 18-7 lists Cisco Discovery Protocol show commands.

Table 18-7 Cisco Discovery Protocol Show Commands

Task	Command
Display global Cisco Discovery Protocol information.	**show ciscodp**
Display summary information about the Cisco Discovery Protocol on one or more ports.	**show ciscodp port info** [*port-string*]
Display Network Neighbor Discovery information from all supported discovery protocols.	**show neighbors** [*port-string*]

Refer to the *Extreme Networks S-Series CLI Reference* for a description of the output of each command.

19

Data Center Bridging Configuration

This chapter provides information about configuring and monitoring Data Center Bridging (DCB) protocols on S-Series devices.

How to Use Data Center Bridging in Your Network

Data Center Bridging (DCB) enhances Ethernet technology by enabling the convergence of various applications in data centers (such as Local Area Networks (LAN), Storage Area Networks (SAN), and advanced application High Performance Computing (HPC)) onto a single interconnect technology, by providing enhancements to existing 802.1 bridge specifications. Existing high-performance data centers typically comprise multiple application-specific networks that run on different link layer technologies, such as Fibre Channel for storage, InfiniBand for high-performance computing, and Ethernet for network management and LAN connectivity. Data Center Bridging enables 802.1 bridges to be used for the deployment of a converged network where all applications can be run over a single physical infrastructure.

The current Data Center Bridging implementation consists of:

- **Enhanced Transmission Selection** (ETS): Provides a common management framework for assignment of bandwidth to 802.1p CoS-based traffic classes (IEEE 802.1Qaz).
- **Congestion Notification** (**CN**): Allows a device to detect congestion on an egress transmit queue and send a message back to the source to back off the traffic rate to alleviate the congestion (IEEE 802.1Q-2011).
- **Application Priority:** Provides for the advertisement to the link peer of a preferred priority to be applied to frames carrying application-specific traffic.

The base control protocol utilized in Data Center Bridging is the Data Center Bridging Exchange (DCBX) protocol. DCBX can be used by a device: to detect peer device capabilities, to detect mis-configuration of a feature between the peers on a link, to perform configuration of DCB features on the link peer.

Enhanced Transmission Selection, congestion notification, and Application Priority protocols utilize DCBX. DCBX uses LLDP to exchange attributes between two linked peers. LLDP is unidirectional and advertises connectivity and management information about the local station to adjacent stations on the same IEEE 802 LAN. DCBX state machines are invoked when the remote MIB changes and a DCBX TLV is present.

Implementing Data Center Bridging

This chapter describes how to configure DCB features on Extreme Networks platforms.

- Configure ports for Enhanced Transmission Selection specifying the ETS groups and the bandwidth allocation assigned to each ETS group and optionally advertising the ETS configuration using the LLDP-DCB Enhanced Transmission Configuration and Enhanced Transmission Recommendation TLVs.
- Configure ports to advertise an Application Priority setting to the LLDP link peer for the specified application and advertising the Application Priority configuration using the LLDP Application Priority TLV.
- Configure switch ports for congestion notification.

You configure DCB features separately and independently. Together the three features implement DCB.

Enhanced Transmission Selection Configuration

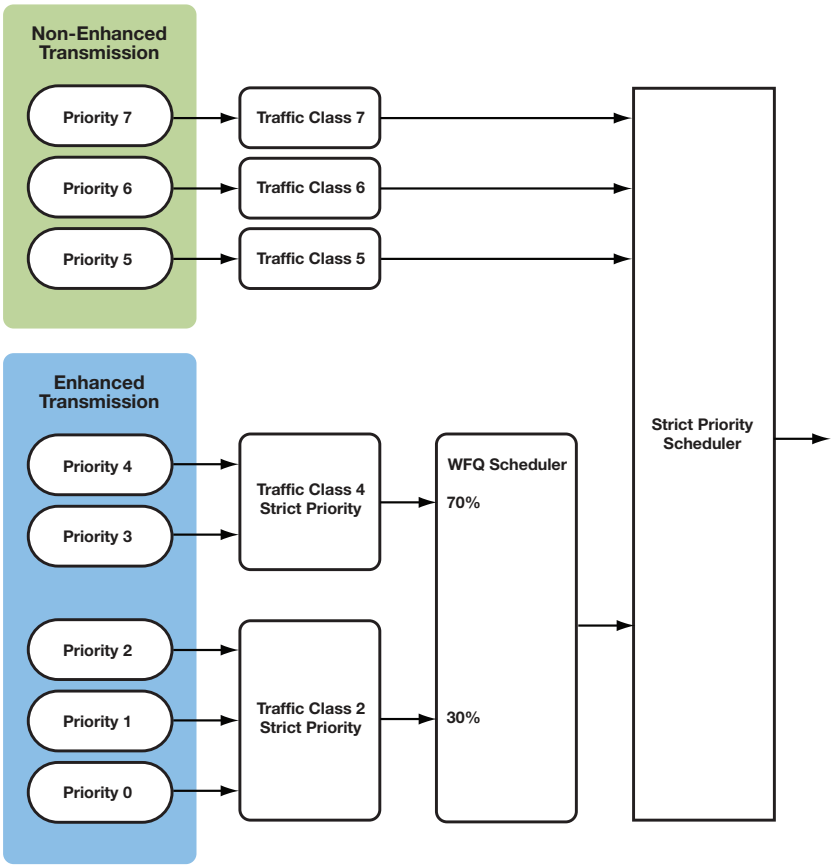
Enhanced Transmission Selection (ETS) queuing provides for configuring two or more traffic class queues (transmit queue (TxQ)) to be allocated for bandwidth that will not be serviced until all non-ETS queues are empty. The firmware services non-ETS selection queues first using strict priority, based upon the priority assigned to the queue. Enhanced Transmission Selection queue contents are forwarded to a fair queue scheduler on a strict priority basis. The fair queue scheduler distributes the remaining bandwidth, after all non-Enhanced Transmission Selection queues are empty, based upon the bandwidth allocation configured for the Enhanced Transmission Selection queues.

Note: Enhanced Transmission Selection queuing is restricted to configurable queues. S-Series modules support both configurable and non-configurable queues. Non-configurable queues are Low Latency Queues (LLQ). LLQs are labeled LLQ in the **show cos port-config** command display. See “Low Latency Queuing” on page 53-6 for LLQ details.

Enhanced Transmission Selection traffic classes (TxQs) and bandwidth allocation are configured using the **set cos port-config txq** command.

Figure 19-1 presents an Enhanced Transmission Selection queuing example. Priority queues 7, 6, and 5 are assigned traffic classes 7, 6, and 5, respectively. These non-ETS queues are serviced first by the strict priority scheduler based upon priority.

Figure 19-1 Enhanced Transmission Selection (ETS) Queuing



802.1 priorities 0 - 4 are configured for Enhanced Transmission Selection queuing. Priorities 4 and 3 are assigned to traffic class 4. Priorities 0, 1, and 2 are assigned to traffic class 2. If all non-ETS queues are empty and there is remaining bandwidth, traffic classes 4 and 2 will be serviced using weighted fair queue scheduling. Based upon Enhanced Transmission Selection bandwidth allocation, the weighted fair queue scheduler will service traffic class 4 at 70 percent and traffic class 2 at 30 percent of remaining bandwidth. Within each traffic class group (4 and 2 in this example), each priority is serviced based on a strict priority scheduler.

This example shows how to create a CoS transmit queue port group entry named **testTxq** with a port group ID of 2 and a port type ID of 1, assign ETS groups to an 11 queue device, followed by allocation of ETS bandwidth to the assigned groups. Using the **enhanced-groups** option of the **set cos port-config txq** command, ETS group to queue assignment is:

- Group **2** to queues **0**, **1**, and **2**
- Group **4** to queues **3** and **4**

Using the **enhanced-percentage** option of the **set cos port-config txq** command, the assigned ETS bandwidth allocation is:

- **30** percent to group **2**
- **70** percent to group **4**

```
S Chassis(rw)->set cos port-config txq 2.1 name testTxq enhanced-groups
2,2,2,4,4,0,0,0,0,0,0 enhanced-percentage 0,30,0,70,0,0,0,0
```

Use the Enhanced Transmission Selection configuration TLV to advertise this ETS configuration to the peer:

```
S Chassis(rw)->set lldp port tx-tlv enhanced-trans-config
```

Use the Enhanced Transmission Selection recommendation TLV to recommend that the peer use this ETS configuration:

```
S Chassis(rw)->set lldp port tx-tlv enhanced-trans-rec
```

Application Priority Configuration

Application Priority advertises to the peer a preferred priority for frames carrying application-specific traffic. Applications are defined by protocol (Ethertype, TCP, UDP, or Layer 4 port) and protocol ID. Priority tagging is performed by the peer, not by the device advertising the Application Priority. The peer receiving the Application Priority TLV tags its traffic to the advertised priority. Application Priority works with Enhanced Transmission Selection and priority-based flow control in that tagged protocol-specific traffic for the specified priority enforces Enhanced Transmission Selection and priority-based flow control behaviors on the traffic.

For example, Application Priority could advertise to its peer that all iSCSI traffic be tagged with priority 5. In this case, the TCP protocol on well known port 3260, as assigned to iSCSI by IANA, is specified. The peer receiving the Application Priority TLV will tag iSCSI traffic for priority 5 in the Priority Code Point (PCP) tag of its 802.1Q header. Both Enhanced Transmission Selection and priority-based flow control will use this priority 5 setting if each functionality is configured to do so.

Note: The Application Priority feature requires that the peer supports the LLDP willing bit and the willing bit is enabled for priority tagging to occur on the peer. Extreme Networks switches do not currently support the LLDP willing bit. Extreme Networks switches can advertise to the peer a preferred priority for frames carrying application-specific traffic, but an Extreme Networks switch peer will not perform Application Priority tagging.

Use the **set lldp port tx-tlv application-pri** command to enable the advertising of the Application Priority configuration to the link peer using the LLDP-DCB Application Priority TLV.

This example shows how to advertise to the peer priority 4, for the UDP service type dpkeyserv with well known port 1780, for port ge.1.2 and to enable the sending of LLDP-DCB Application Priority TLVs from that port:

```
S Chassis(rw)->set dcb appPri ge.1.2 protocol udp protocol-id 1780 priority 4
S Chassis(rw)->set lldp port tx-tlv application-pri ge.1.2
```

Congestion Notification (CN) Configuration

Note: CN is supported on the S-Series S140 and S180 modules. On non-supported S-Series modules, the Congestion Notification Domain Defense can be configured for either edge or disabled only. When edge configured, flows ingressing non-supported S-Series modules are remapped on ingress. Flows ingressing a supported S-Series module and egressing a non-supported S-Series module, on the same chassis, generate Congestion Notification Messages because congestion notification logic is performed on the ingress module.

Congestion Notification (CN), as defined in IEEE 802.1Q-2011 allows a device to detect congestion at a switch congestion point (egress transmit queue) and transmit a Congestion Notification Message (CNM) PDU back to the reaction point (flow source). The reaction point backs off the traffic rate to alleviate the congestion. Congestion notification supports long lived data flows in a network with delay due to limited bandwidth. It allows for applications that are latency-or-loss-sensitive to run over Ethernet technologies experiencing egress transmit queue congestion.

As the use of Ethernet technologies in the data center expands, prevention of packet loss by some applications becomes more critical. Congestion notification was created to:

- Allow the monitoring of Congestion Controlled Flows (CCFs)
- Detect congestion
- Notify the source to lower the transmit rate for the offending congestion controlled flow.

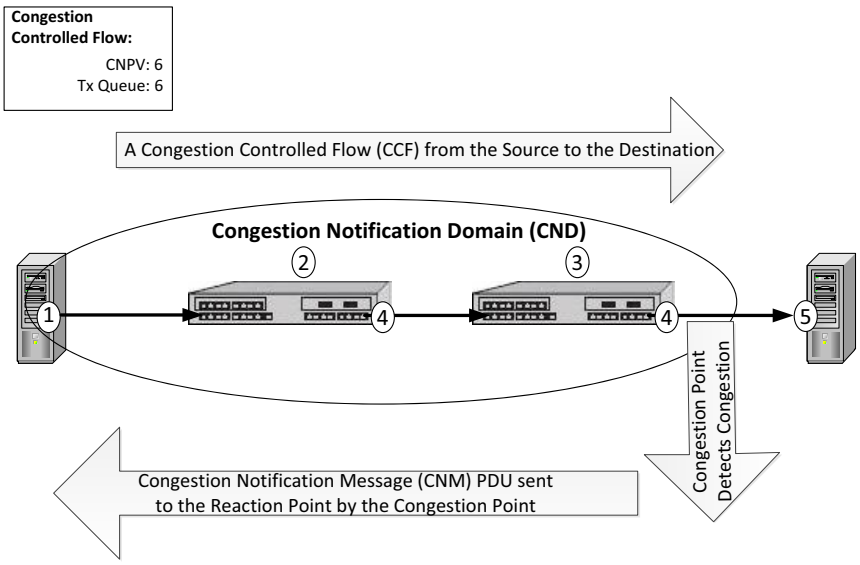
For congestion notification to work, it must be supported at all egress queues for each switch that is in the path from the source to the destination. Congestion notification is applied to an egress queue by configuring an 802.1p value as a Congestion Notification Priority Value (CNPV) that is mapped to the transmit queue using CoS. This collection of egress queues configured for congestion notification make up the Congestion Notification Domain (CND).

Each transmit queue that has been configured for congestion notification is monitored to detect congestion. When congestion is detected, a CNM PDU is generated at the congestion point and sent back to the source with the details of the queue and flow that triggered the message. The source can then use this information to back off the transmission rate for the application that triggered the CNM PDU.

SNMP supports the congestion notification IEEE Standard MIB: IEEE8021-CN-MIB

Figure 19-2 provides a Congestion Notification overview.

Figure 19-2 Congestion Notification Overview



1	Reaction Point (flow source)	**3**	Switch B	**5**	Flow Destination
2	Switch A	**4**	Congestion Point (Tx queue)		

Figure 19-2 identifies the congestion notification reaction point (callout **1**). A reaction point is the source of a congestion controlled flow. A congestion controlled flow consists of frames, all with the same CNPV, and all assigned to a single transmit flow queue in the originating end station. A CNPV is an 802.1p priority mapped to a congestion notification egress queue of each device in the flow.

The reaction point is:

- Capable of optionally adding a CN-TAG to a flow
- Able to process a CNM PDU

- Able to back off the transmission rate of the congestion controlled flow based on information contained in the CNM PDU

The reaction point is connected to a destination device (callout **5**) by traversing switches A and B (callouts **2** and **3**). All egress ports on switches in the path between the reaction point and the destination are configured as congestion points (callouts **4**). All congestion points are configured for a CNPV. In our example, CNPV 6 is mapped to transmit queue 6 for all congestion points.

Up to either four or seven (depending upon the chassis) 802.1p priorities mapped to a port's transmit queues can be configured as CNPVs. At least one 802.1p priority on a port must be a non-congestion aware priority. Any non-congestion aware priority can be used as a congestion notification alternate priority. When a packet that does not belong to a congestion controlled flow has the same priority as a CNPV configured on a congestion notification domain edge ingress port, it must be remapped to an alternate priority to defend against a false triggering of a congestion notification by a non-congestion controlled flow.

In Figure 19-2, the reaction point tags the traffic with the CNPV 6 mapped to queue 6 and transmits it to the destination. The congestion point at switch B identifies congestion and creates a CNM PDU packet that is sent back to the reaction point. When the reaction point receives the CNM PDU, it uses the information contained in the CNM PDU to back off the reaction point transmission rate for the queue associated with the congestion controlled flow that triggered the CNM PDU.

A CN-TAG helps the source identify the flow. The CN-TAG is optional for packets transmitted by the source and, if present, may contain a flow-ID. All CNM PDU packets sent from the congestion point back to the reaction point contain a CN-TAG. The transmit queue that detects the congestion will use the flow-ID from the CN-TAG, if it is present, when generating the CNM PDU back to the source. If the congestion controlled flow does not contain a CN-TAG, congestion notification sets the flow-ID to 0.

The traffic monitored by congestion notification must be isolated to its own CNPV. Traffic that does not support congestion notification must not be placed on a CNPV mapped queue. The source of this non-congestion aware priority traffic would not understand the CNM PDU being transmitted back from the congestion point, defeating the purpose of congestion notification. LLDP may be used to auto-determine the port's capability. The information that LLDP provides ensures that non-supported flows are not placed in the congestion aware traffic classes mapped to a CNPV.

Congestion notification advertises its support and state via a TLV type 127 defined in LLDP. If a non-congestion aware priority frame ingresses a congestion notification domain edge port, with the same priority as a configured CNPV on that port, congestion notification will use an alternate priority value to remap the packets to a non-CNPV queue. This alternate priority is configurable for each CNPV.

Using LLDP and alternate priority settings, devices can be configured to remap a priority for all ports away from a congestion notification configured queue and rely upon LLDP to dynamically determine which ports support congestion notification. The protection of the congestion notification domain from non-CN capable port traffic is referred to as congestion notification domain defense. See "Congestion Notification Domain Defense" on page 19-8 for a detailed discussion of domain defense.

Implementing Congestion Notification

To implement congestion notification:

- Globally enable congestion notification on the device
- Create congestion notification priority values and optionally specify the creation mode for the CNPVs

- Activate the configured congestion notification priority values on all switch ports
- Optionally modify the transmit priority for the CNM PDU sent by the congestion point to the reaction point
- Optionally modify the default domain defense mode for a priority for the device or on a port basis
- Optionally modify the default priority choice globally or on a port basis
- Optionally modify the default alternate priority for a CNPV globally or on a port basis
- Optionally enable congestion notification LLDP for a CNPV for the device or on a port basis
- If priority choice auto is configured on a port, enable the sending of congestion notification TLVs in the LLDP PDUs for each congestion point port on the system.

Enabling Congestion Notification

Congestion notification must be globally enabled on the switch for congestion notification to be operational. Use the **set dcb cn global** command to globally enable congestion notification on the switch. This command globally enables congestion notification on the switch, but it does not affect the state of a created CNPV. See "Congestion Notification Priority Value (CNPV)" on page 19-7 for CNPV creation and activation information. Congestion notification is globally enabled by default.

The transmit priority option of the **set dcb cn global** command allows you to modify the transmit priority value of the CNM PDUs sent from the congestion point to the reaction point when congestion is detected. The default value is priority **6**.

Congestion Notification Priority Value (CNPV)

A congestion notification priority value (CNPV) is an 802.1p value configured for congestion notification and mapped to the same queue on all ports that make up the congestion notification domain for that CNPV. There are eight 802.1p values from 0 - 7. The maximum number of CNPVs configurable on a port depends upon the chassis. The SSA, S3, and S4 chassis support a maximum of seven CNPVs. The S6 and S10 chassis support a maximum of four CNPVs. There must always be at least one alternate (non-CNPV) priority value per port.

Use the **set dcb cn priority** command, specifying the 802.1p priority to set as the CNPV and optionally specifying a creation mode, to create a CNPV for all ports on the device.

By default, a CNPV is created with auto creation enabled when either the **creation enable** option is specified or no option is specified. "Auto creation enabled" specifies that the port-priority choice mode for each port-priority configured for this CNPV is set to **auto** and the domain defense mode default is set to **edge**. See "Priority Choice" on page 19-11 for priority choice details. See "Congestion Notification Domain Defense" on page 19-8 for domain defense mode details.

If **creation disable** is specified when creating the CNPV, the port-priority choice mode for each port-priority is set to **admin** and the domain defense mode default is set to **interior**. Priority choice mode **admin** uses administratively configured settings when determining the domain defense mode for a port.

Use the **set dcb cn priority status** command to activate a CNPV on the switch. Congestion notification will not occur for a CNPV unless the CNPV is activated on the switch. CNPVs are active by default. Priority status defaults to **enable** on creation.

Note: CNPVs do not exist on S-Series hardware bonding ports.

Alternate Priority

The congestion notification alternate priority is a non-CNPV used to protect the congestion notification domain from a non-congestion controlled flow packet with the same priority as a configured CNPV on the port from triggering congestion notification. At least one 802.1p priority on a port must be a non-CNPV. Any non-CNPV can be used as a congestion notification alternate priority.

When a packet ingresses a port at the edge of a congestion notification domain and has the same priority as a CNPV configured on the ingress port, the packet's priority must be remapped to an alternate priority. Should a non-congestion notification packet trigger congestion in a CNPV queue, the source for this packet will not know what to do with the CNM PDU it receives back from the congestion point. The remapping of the priority to a non-CNPV value at the congestion notification domain edge guards against this possibility.

An alternate priority can be set both globally or on a port basis. The global alternate priority is only used if the port-priority choice set using the **set dcb cn port-priority choice** command is set to **default** and the priority choice set using the **set dcb cn priority choice** command is set to **admin**. The port-priority alternate priority is only used if the port-priority choice is set to **admin**. (See "Priority Choice" on page 19-11 for a priority choice discussion.) This administratively set global or port-based alternate priority defaults to **0**. Otherwise, the auto alternate priority is used. The auto alternate priority for a CNPV defaults to either the next lowest non-CNPV priority if a lower one exists on the port or the next highest non-CNPV priority on the port.

Use the **set dcb cn priority alt-pri** command to change the global alternate priority default.

Use the **set dcb cn port-priority alt-pri** command to change the alternate priority default on a port basis. This command overrides any global configuration for the specified port.

Congestion Notification Domain Defense

A congestion notification domain defense provides a means of defending a congestion notification domain against incoming frames from outside of the domain. Domain defense assumes:

- That every bridge along a path between two congestion aware end-stations, using a particular CNPV, is properly configured for congestion notification and therefore belongs to the congestion notification domain
- That every bridge ensures that frames not configured for a CNPV use different queues than the CNPV configured queues for those devices

Domain defense protects the boundaries of a congestion notification domain by preventing frames not in a congestion controlled flow from entering congestion point controlled queues. Domain defense takes advantage of the ability to change the priority value based upon whether or not the port's neighbor is also configured with the same CNPV. If a frame with the same priority as the CNPV is not in the congestion controlled flow, the frame priority is changed to the configured alternate priority for that CNPV.

A default domain defense mode is configured at each congestion point port.

There are four possible domain defense modes depending upon whether the CNPV is configured for the congestion point, whether a given congestion point knows the congestion notification state of its neighbor, and where the congestion point port is located in the congestion notification domain:

- **Disabled** – The domain defense mode state on a port for which congestion notification is disabled. The priority is not a CNPV. Congestion notification does not control priority remapping of input frames on this port. CN-TAGs are neither added by an end station nor removed by a bridge. Disabled mode is only set administratively.

- **Edge** – The domain defense mode configured on a congestion point port that resides at the edge of the congestion notification domain. All frames ingressing the edge of a congestion notification domain by definition do not belong to a congestion controlled flow for this domain. On this port for the given CNPV, congestion notification controls priority remapping. The input frame priority parameters are remapped to an alternate (non-CNPV) value. CN-TAGs are not added by an end station, and are removed from frames before being output by a bridge. This mode is optional for an end station.

  The end result is that ports configured for edge domain defense protect the congestion notification domain by reassigning the 802.1p priorities of non-CNPV ingressing frames to alternate priorities when they are the same as an edge port CNPV.

  When configured for LLDP dynamic congestion notification, a port will be auto configured as edge when the neighbor is not configured for this CNPV.

- **Interior** – The domain defense mode configured on a congestion point port that resides within the congestion notification domain between the flow's source reaction point and the destination end-station. This port does not yet know whether its neighbor is able to receive a CN-TAG in frames sent to it. On this port for the given CNPV, the input frame priority parameters are not remapped. CN-TAGs are not added by an end station, and are removed from frames before being output by a bridge.

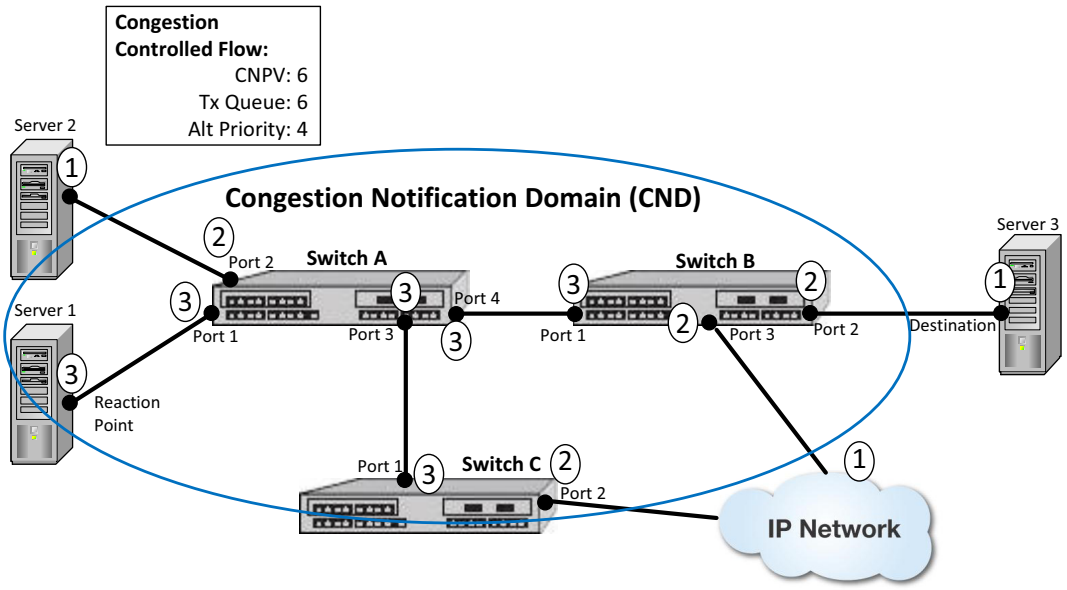
  The Interior defense mode is a transition state within an LLDP dynamic congestion notification configuration. Once the port completes congestion notification negotiation with its neighbor, the defense mode transitions to interior-ready.

- **Interior-Ready** – The domain defense mode configured on an interior congestion port that knows its neighbor is able to receive a CN-TAG in frames sent to it. On this port for the given CNPV, the input frame priority parameters are not remapped. CN-TAGs can be added by an end station, and are not removed from frames by a bridge.

Note: Manually configuring congestion notification domain defense on S-Series bonding ports has no affect. Both hardware and software bonding ports are automatically configured for interior-ready.

Figure 19-3 on page 19-10 provides a dynamic domain defense mode configuration overview.

Figure 19-3 Congestion Notification Domain Defense Mode Overview



1	Non-CN configured port	**2**	Edge defense port	**3**	Interior ready defense port

In Figure 19-3, there are two packet flow sources. One of the flow sources is a reaction point configured for CNPV 6 and mapped to queue 6 (Server 1). The second flow source is not configured for congestion notification (Server 2).

There are two paths between the two packet flow sources and the destination. The first path is from the two flow sources to the destination through switches A and B. The second path is from the flow sources to the destination through switches A and C, an IP network cloud, and switch B.

There are three flow discussions that can be derived from Figure 19-3.

- **Server 2 non-congestion notification flow** – The 802.1p priority 6 frame sourced at Server 2 is a non-CN frame because its source is not a reaction point within a congestion notification domain. As the frame enters port 2 Switch A, because port 2 is a domain edge port, and the frame priority agrees with CNPV for this domain, the frame priority (6) is changed to the alternate priority value (priority 4). The frame transits the remainder of the path to the destination incapable of triggering congestion notification.
- **Server 1 congestion notification flow (Switch A/Switch B path)** – The CNPV 6 frame sourced at Server 1 (reaction point) is a congestion notification frame. As the frame transits to the destination, both ingress ports are configured for the interior-ready defense mode because they have successfully negotiated CNPV 6 with their peers. The CNPV value is not changed to an alternate priority when ingressing interior-ready ports. The frame exits the congestion notification domain for CNPV 6 at port 2, Switch B, and arrives at the destination with its priority unchanged.

  Should congestion occur at port 4 of Switch A or port 2 of Switch B, a CNM PDU will be sent back to the reaction point which will back off the flow transmit rate so long as it receives CNM PDUs from the congestion point.
- **Server 1 congestion notification flow (IP Network path)** – The CNPV 6 frame sourced at Server 1 (reaction point) is a congestion notification frame. As the frame transits to the destination, Switch A and Switch C ingress ports are configured for the interior-ready defense mode because they have successfully negotiated CNPV 6 with their peers. The frame leaves the congestion notification domain at port 2, Switch C. The CN-Tag (if present) is stripped and the priority is unchanged. Assuming there are no policy or other reasons why the priority

would be changed in the IP Network cloud, the frame arrives at port 3, Switch B with a priority of 6. Because port 3, Switch B is configured for edge defense and the frame priority is the same as the congestion notification domain CNPV 6, the frame priority is changed to the alternate priority 4 no longer capable of triggering congestion notification within this domain. The frame transits to the destination with a priority of 4.

Should congestion occur at port 3, Switch A or port 2, Switch C, a CNM PDU will be sent back to the reaction point which will back off the flow transmit rate so long as it receives CNM PDUs from the congestion point. Should congestion occur at port 2, Switch B, congestion notification is not triggered because the priority is now alternate priority 4.

Port defaults for domain defense are determined by priority choice. See "Priority Choice" on page 19-11 for details.

Defaults for domain defense can be administratively configured by priority on a port or for all priorities on a port. A default domain defense can be globally configured per CNPV for all ports using the **set dcb cn priority defense** command. A default domain defense can be set on a port basis for all CNPVs on that port using the **set dcb cn port-priority defense** command.

Priority Choice

There are two priority choice modes that determine how domain defense is configured globally on the switch and three priority choice modes on a port basis. The priority choice mode can be set to:

- **Admin** – Domain defense is administratively configured. When defense choice is set to **admin**, defense mode defaults to **interior** on all ports. Admin can be configured both globally and on a port basis.
- **Auto** – Domain defense is dynamically configured using LLDP. When defense choice is set to **auto**, defense mode defaults to **edge** on all ports. Auto can be configured both globally and on a port basis.
- **Default** – Domain defense is based upon the creation setting (**enable** or **disable**) used when the CNPV is created. If **creation enable** is set, domain defense defaults to **auto**. If **creation disable** is set, domain defense defaults to **admin**.

Table 19-1 cross-references port-priority choice and priority choice settings with the default defense mode and alternate priority settings.

Table 19-1 Choice, Defense Mode, and Alternate Priority Cross-Reference

If Port-Priority Choice is:	And Global Priority Choice is:	Then Defense Mode is:	And Alternate Priority is:
auto	auto or admin	auto defense mode (Default: edge)	auto alternate priority
admin	auto or admin	port-priority admin defense mode (Default: disabled)	port-priority alternate priority
default	auto	auto defense mode (Default: edge)	auto alternate priority
default	admin	priority admin defense mode (Default: interior)	priority alternate priority

For example, if the global priority choice is set to auto and the port-priority choice is set to default (row 3 of Table 19-1), both the defense mode and alternate priority are auto chosen. In this case, the defense mode would default to edge and the alternate priority would default to the next lowest non-CNPV value, or if no lower one exists, the next highest non-CNPV value.

Priority choice on a global basis is configured using the **set dcb cn priority choice** command.

Priority choice on a port basis is configured using the **set dcb cn port-priority choice** command.

LLDP

LLDP can be used to dynamically enable domain defense on a port for a CNPV. Since Congestion Notification requires every node in the path to be capable of congestion notification support, the nearest bridge LLDP address is used in advertising the support. LLDP TLV type 127 is used by congestion notification and includes a:

- Per-priority CNPV indicator – An 8-bit field where each bit represents the priority for the port and if it's capable of congestion notification.
- Per-priority ready indicator – An 8-bit field where each bit represents the priority for the port and if it's ready for congestion notification (set to 1 if the port is enabled for CNPV).

If a congestion notification capable port receives a TLV type 127 signaling that its neighbor is capable on the same CNPV, the ready indicator is set and the port transitions from the edge to the interior ready domain defense mode for that CNPV.

For dynamic configuration of domain defense to take place you must:

- Assure that the global priority choice is set to auto (default setting when creating a CNPV in creation enable mode)
- Enable congestion notification LLDP on the device using the **set dcb cn priority lldp** command (defaults to enabled)
- Enable the sending of congestion notification TLVs on each congestion point port using the **set lldp port tx-tlv congestion-notif** command (defaults to disabled)

Congestion Point Queue

The settings for a subset of parameters associated with a congestion point queue can be modified. Congestion point indexes are assigned to congestion point queues which have one or more CNPVs mapped to them on a given port. These congestion queues can be configured using the assigned congestion point index. The congestion point index is one greater than the lowest CNPV which is mapped to the congestion notification queue. For example: if CNPVs 4 and 7 are configured for a transmit queue, the congestion point index is 5. A queue profile which supports the configuring of a minimum sample option can also be associated with the congestion point configuration.

Congestion point queues are configured with the **set dcb congestion-point** command. You can configure the following:

- **Set point** – A target value for the number of octets in the congestion point queue. CNM PDUs are transmitted to the sources of frames queued in this congestion point's queue in order to keep the total number of octets stored in the queue at the set point value.
- **Weight** – An integer value used for calculating W. W is the weight to be given to the change in queue length when calculating a measure of transmit queue congestion known as quantitized feedback (Fb) as defined in the IEEE 802.1Q-2011 standard.

  The weight option is an integer value from which W is derived. W is equal to two to the power of the weight value specified here. Thus, if weight equals a -1, W = 1/2. W can be between the values specified by Min Weight and Max Weight for this device as displayed by the **show dcb cn q-profile** command. See the IEEE 802.1Q-2011 standard for a detailed discussion for W, weight, and Fb.

A maximum of two congestion notification queue profiles can be associated with a congestion point queue configuration. The default queue profile on the the S-Series has an identifier of **0.1** and can be displayed using the **show dcb cn q-profile** command. This default queue profile can not be modified. A second queue profile, **1.1**, can be configured. The queue profile is configured using the **set dcb cn q-profile** command. See "Congestion Notification Queue Profile" on page 19-13 for a congestion notification queue profile discussion.

Use the **show dcb cn congestion-point** command to display congestion point queue information.

Congestion Notification Queue Profile

Each congestion notification queue belongs to one of two possible queue profiles on the S-Series. A queue profile is a management object containing congestion notification queue configuration. A queue profile is named based upon an index value and the queue type in dotted notation. The S-Series queue type is always 1. For example: the queue profile name for queue profile index 0 would be **0.1**.

The queue profile is configured with the **set dcb cn q-profile** command. You can configure the minimum sample parameter, which specifies the minimum number of octets to enqueue in the congestion point queue between transmissions of CNM PDUs. The default value is 150000 octets.

A queue profile is applied to a congestion queue configuration by specifying its identifier when configuring a congestion point queue using the **set dcb cn congestion-point** command.

Use the **set dcb cn q-profile** command to modify the minimum sample parameter for a group of congestion point queues.

Use the **show dcb cn q-profile** command to display congestion point queue profile information.

Congestion Notification Configuration Example

The following CLI input shows how to configure each switch for the dynamic domain defense as presented in Figure 19-3 on page 19-10.

```
S Chassis(rw)->set dcb cn priority 6
S Chassis(rw)->set dcb cn priority 6 status enable
S Chassis(rw)->set lldp port tx-tlv congestion-notif *.*.*
```

This example assumes:

- This congestion notification configuration is a default configuration on all switches using LLDP to dynamically configure domain defense on all ports.
- That CoS priority 6 is mapped to transmit queue 6 on all ports within the congestion notification domain. Should there be other CNPV values configured, these may use transmit queue 6, but no non-CNPV priority is mapped to transmit queue 6.

Note: Congestion notification and congestion notification LLDP are globally enabled by default. A non-auto alternate priority is not configurable when using auto congestion notification.

Configuring Data Center Bridging

Table 19-2 lists the S-Series device default Data Center Bridging configuration settings.

Table 19-2 Default Data Center Bridging Configuration Settings

Parameter	Description	Default Value
Application Priority	Advertises to the LLDP peer a preferred priority for frames carrying application-specific traffic.	disabled on all ports
CN alternate priority	The priority a packet is mapped to if a non-CNPV packet ingresses a port set for domain defense mode edge.	relative to CNPV, either the next lower non-CNPV priority, if it exists, or the next higher.
CN transmit priority	The priority assigned to the CNM PDUs sent by the congestion point back to the reaction point when congestion is detected.	7
CNPV creation status	Specifies whether auto choice is enabled or disabled on the switch for the created congestion notification priority value (CNPV).	enabled
CNPV status	Specifies whether a configured CNPV is activated (enable) or not activated (disable) on the switch.	activated (enable)
congestion notification domain defense mode	Determines whether the ingressing packet priority is remapped away from a configured CNPV value on the port and whether a CN-TAG can be added or removed from the packet.	Edge if port-priority choice is set to auto; Disabled if port-priority choice is set to admin; Determined by the default choice of the priority, if port-priority is set to default.
congestion notification global status	Specifies whether congestion notification is enabled globally on the switch.	enabled
congestion notification LLDP	Used to send LLDP CN TLVs for all CNPVs on the port. The actual sending of the TLVs is disabled by default.	enabled
congestion queue minimum sample	Specifies the minimum number of octets to enqueue in the congestion point queue between transmissions of CNM PDUs	150000 octets
Enhanced Transmission Selection (ETS)	Provides for the designation of two or more traffic class queues to be allocated for bandwidth that will not be serviced until all non-ETS queues are empty.	No ETS class queues are configured

Table 19-3 lists Data Center Bridging configuration commands.

Table 19-3 Data Center Bridging Configuration

Task	Command
Optionally, map traffic classes to ETS groups in the CoS transmit queue port group configuration and specify the bandwidth allocation for the group.	**set cos port-config txq** *group-type-index* [**name** *name*] [**enhanced-groups** *group-id*] [**enhanced-percentage** *bandwidth*]
Optionally, set the priority to be advertised to the peer of the specified port for the specified application.	**set dcb appPri** *port-string* **protocol** {**ethertype** | **tcp** | **udp** | **l4port**} **protocol-id** *protocol-id* **priority** *priority*

Table 19-3 Data Center Bridging Configuration (continued)

Task	Command
Set Data Center Bridging optional LLDP-DCB TLVs to be transmitted in LLDP PDUs by the specified port or ports.	**set lldp port tx-tlv** {[**enhanced-trans-config**] [**enhanced-trans-rec**] [**application-pri**] [**priority-flowctrl**] [**congestion-notif**]} *port-string*

Table 19-4 lists Congestion Notification global configuration commands. All Congestion Notification commands can be entered from any command mode.

Table 19-4 Congestion Notification Global Configuration

Task	Command
Globally enable congestion notification on the switch.	**set dcb cn global enable**
Optionally, modify the priority for Congestion Notification Messages sent from the congestion point back to the congestion notification reaction point.	**set dcb cn global tx-priority** *tx-priority*
Optionally, modify congestion point queue parameters for queues associated with a congestion point queue configuration or associate a congestion notification queue profile with the configuration.	**set dcb cn congestion-point** *port-string cp-index* [**set-point** *set-point*] [**weight** *weight*] [**qp-index** *qp-index*]
Optionally, modify congestion notification queue profile parameters associated with up to two queue profiles on the S-Series device	**set dcb cn q-profile** *qp-identifier* [**min-sample** *min-sample*]

Table 19-4 lists Congestion Notification global priority configuration commands. All Congestion Notification commands can be entered from any command mode.

Table 19-5 Congestion Notification Global Priority Configuration

Task	Command
Configure up to seven 802.1p priorities as Congestion Notification Priority Value (CNPV) on the switch. Optionally, configure priority choice mode to **admin** using the creation **disable** option. Priority choice defaults to **auto**.	**set dcb cn priority** *priority* [**creation** {**enable** | **disable**}]
Activate configured CNPVs for all ports on the switch.	**set dcb cn priority** *priority* **status enable**
Optionally modify the priority choice mode on all device ports for the specified CNPV. Priority choice globally defaults to **auto**.	**set dcb cn priority** *priority* **choice** {**admin** | **auto**}
Optionally, modify the global default congestion notification alternate priority for the specified CNPV.	**set dcb cn priority** *cnpv* **alt-priority** *alt-priority*
Optionally, administratively configure a global default domain defense mode for all ports on the switch for the specified priority.	**set dcb cn priority** *priority* **defense** {**disabled** | **edge** | **interior** | **interior-ready**}
If using the default priority choice of **auto**, enable auto configuration of domain defense operation using LLDP.	**set dcb cn priority** *priority* **lldp enable**

Table 19-6 lists Congestion Notification port priority configuration commands.

Table 19-6 Congestion Notification Port Priority Configuration

Task	Command
Optionally, for all priorities, modify the default method for determining how the domain defense is selected for the specified port.	**set dcb cn port-priority** *port-string* **choice** {**admin** | **auto** | **default**}
Optionally modify the alternate priority for the specified port and CNPV.	**set dcb cn port-priority** *port-string priority* **alt-priority** *alt-priority*
Optionally, administratively modify the default domain defense mode for the specified priority on the specified port.	**set dcb cn port-priority** *port-string priority* **defense** {**disabled** | **interior** | **interior-ready** | **edge**}
If using the default priority choice of **auto**, enable auto configuration of domain defense operation using LLDP for a specific priority and port.	**set dcb cn port-priority** *port-string priority* **lldp** {**enable** | **disable**}
Optionally, set the method that determines how the domain defense is selected for the specified priority on the specified port.	**set dcb cn port-priority** *port-string* **priority** *priority* **choice** {**admin** | **auto** | **default**}

Table 19-7 lists Data Center Bridging display commands.

Table 19-7 Data Center Bridging Display Commands

Task	Command
Display the ETS group CoS transmit queue port group mappings by the port group and type.	**show cos port-config txq** *group-type-index*
Display Application Priority table entries by port.	**show dcb appPri** *port-string*
Display LLDP port Data Center Bridging, priority-based flow control, Application Priority, or congestion notification transmit TLV support.	**show lldp port tx-tlv** [**data-center-bridging**] [**priority-flowctrl**] [**application-pri**] [**congestion-notif**]
Display the local or remote system information, including ETS information, stored for one or more ports.	**show lldp port** {**local-info** | **remote-info**} [*port-string*]
Display the global status of congestion notification on the switch.	**show dcb cn global**
Display the configuration and status of congestion notification priority values on the switch.	**show dcb cn priority** [*priority*] [**-interesting**]
Display congestion notification port level defense mode configuration.	**show dcb cn port-priority** [*port-string* [*priority*]] [**priority** *priority*] [**-interesting**]
Display the configuration and status of congestion notification congestion points.	**show dcb cn congestion-point** [*port-string* [*cp-id*]] [**stats**]
Display the congestion notification queue profile configuration stats.	**show dcb cn q-profile** [*profile-id*]

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

Terms and Definitions

Table 19-8 lists terms and definitions used in this DCB configuration discussion.

Table 19-8 Data Center Bridging (DCB) Configuration Terms and Definitions

Term	Definition
Application Priority	A DCB feature that provides for the advertisement to the peer of a preferred priority to be applied to frames carrying application-specific traffic. The peer must support the LLDP willing bit.
Congestion Notification (CN)	A DCB feature, as defined in IEEE 802.1Q-2011, that allows a device to detect congestion at a switch congestion point and transmit a Congestion Notification Message back to the reaction point indicating the reaction point should back off the traffic for that flow.
Congestion Notification Message (CNM)	A PDU message sent from the congestion point back to the reaction point to back off on a transmitting the flow when congestion is detected in a congestion notification enabled context.
congestion point	An egress transmit point configured for congestion notification.
Data Center Bridging (DCB)	A group of features that enhance Ethernet technology by enabling the convergence of various applications in data centers, such as Local Area Networks (LAN), Storage Area Networks (SAN), and advanced application High Performance Computing (HPC) onto a single interconnect technology, by providing enhancements to existing 802.1 bridge specifications.
Data Center Bridging Exchange (DCBX)	A protocol that allows Ethernet devices to detect DCB capability on the peer device, as well as DCB configuration between peer devices.
Enhanced Transmission Selection (ETS)	A DCB feature that provides a common management framework for assignment of bandwidth to 802.1p CoS-based traffic classes (IEEE 802.1Qaz).
LLDP willing bit	An LLDP attribute that when enabled instructs the local device to use the peer Application Priority settings contained in received TLVs. The LLDP willing bit is not currently supported on Extreme Networks switches.
reaction point	The flow source device capable of optionally adding a CN-TAG to the flow, has the ability to process a CNM PDU, and is able to throttle its transmission rates based on information contained in the CNM PDU. .

20

Simple Network Management Protocol (SNMP) Configuration

This chapter provides information about configuring and monitoring SNMP on Extreme Networks S-Series devices.

Using SNMP in Your Network

The Simple Network Management Protocol (SNMP) is an application layer protocol that facilitates the exchange of management information between network devices. The most widely used management protocol on Internet Protocol (IP) networks, it helps you monitor network performance, troubleshoot problems, and plan for network growth.

SNMP's simplicity lies in the fact that it uses a basic set of command messages to relay notifications of events and error conditions over a connectionless communication link.

Most network devices support the three versions of the protocol: SNMPv1, SNMPv2c, and SNMPv3. The latest version, SNMPv3, provides enhanced security and administrative features as described in this document.

SNMP is a simple, cost-effective tool for monitoring your network devices for conditions that warrant administrative attention. It is widely used because it is:

- Easily integrated into your existing LAN topology
- Based on an open standard, making it non-proprietary and well documented
- Flexible enough to communicate the specific conditions you need monitored in your network
- A common management platform supported by many network devices

High-Level Configuration Process

You can implement SNMP on Extreme Networks switching devices using simple CLI commands as described in this chapter. The configuration process involves the following tasks:

1. Creating users and groups allowed to manage the network through SNMP
2. Setting security access rights
3. Setting SNMP Management Information Base (MIB) view attributes
4. Setting target parameters to control the formatting of SNMP notification messages
5. Setting target addresses to control where SNMP notifications are sent
6. Setting SNMP notification parameters (filters)
7. Reviewing SNMP statistics

SNMP Concepts

It is helpful to understand the following SNMP concepts:

Manager/Agent Model Components

SNMP provides a message format for communication between managers and agents, which use a MIB and a relatively small set of commands to exchange information. The SNMP manager can be part of a network management system, such as Extreme Networks NetSight, while the agent and MIB reside on the switch.

The SNMP agent acts upon requests from the manager to either collect data from the MIB or to set data into the MIB. A repository for information about device parameters and network data, the MIB is organized in a tree structure in which individual variables are represented as leaves on the branches. A unique object identifier (OID) distinguishes each variable in the MIB and is the means by which the manager and agent specify which managed elements are changed.

An agent can send unsolicited notification messages (also known as traps or informs) alerting the SNMP manager to a condition on the network. These conditions include such things as improper user authentication, restarts, link status (up or down), MAC address tracking, closing of a TCP connection, loss of connection to a neighbor, or other significant events.

Message Functions

SNMP uses five basic message types (Get, Get Next, Get Response, Set, and Trap) to communicate between the manager and the agent. The Get and Get Next messages allow the manager to request information for a specific variable. The agent, upon receiving a Get or Get Next message, will issue a Get Response message to the manager with either the information requested or an error indication about why the request cannot be processed.

A Set message allows the manager to request a change to a specific variable. The agent then responds with a Get Response message indicating the change has been made or an error indication about why the change cannot be made.

A trap or inform message allows the agent to spontaneously inform the manager of an "important" event in the network.

The SNMP manager and agent use information in the MIB to perform the operations described in Table 20-1.

Table 20-1 SNMP Message Functions

Operation	Function
get-request	Retrieves a value from a specific variable.
get-next-request	Retrieves a value from a variable within a table.[1]
get-bulk-request[2]	Retrieves large blocks of data, such as multiple rows in a table, that would otherwise require the transmission of many small blocks of data.
get-response	Replies to a get-request, get-next-request, and set-request sent by a management station.
set-request	Stores a value in a specific variable.
trap | inform[3]	Unsolicited message sent by an SNMP agent to an SNMP manager when an event has occurred.

1. With this operation, an SNMP manager does not need to know the exact variable name. A sequential search is performed to find the needed variable from within a table.
2. The get-bulk operation is only supported in SNMPv2c or later.
3. Inform notifications are only supported in SNMPv3.

Trap Versus Inform Messages

As compared to earlier versions, SNMPv3 provides a higher degree of reliability for notifying management stations when critical events occur. Traditionally, SNMP agents communicated events to SNMP managers via "traps." However, if a temporary network problem prevented the manager from receiving the trap, then the trap would be lost. SNMPv3 provides "informs", which are a more reliable form of traps. The SNMP agent initiates the inform process by sending an inform request to the manager. The manager responds to the inform request to acknowledge receipt of the message. If the inform is not received by the manager, the inform request will timeout and a new inform request will be sent. Subsequent inform requests will be sent as previous requests time-out until either an acknowledgement is received from the manager, or until a pre-specified retry-count is reached.

Access to MIB Objects

SNMP uses the following authentication methods to grant user access to MIB objects and functions.

Community Name Strings

Earlier SNMP versions (v1 and v2c) rely on community name strings for authentication. In order for the network management station (NMS) to access the switch, the community string definitions on the NMS must match at least one of the three community string definitions on the switch. A community string can have one of these attributes:

- Read-only (**ro)**—Gives read access to authorized management stations to all objects in the MIB except the community strings, but does not allow write access.

- Read-write (**rw)**—Gives read and write access to authorized management stations to all objects in the MIB, but does not allow access to the community strings.

User-Based

SNMPv3 provides a User-Based Security Model (USM) which relies on a user name match for authenticated access to network management components.

Refer to "Security Models and Levels" on page 20-6 for more information.

SNMP Support on S-Series Devices

By default, SNMP Version 1 (SNMPv1) is configured on Extreme Networks switches. The default configuration includes a single community name - public - which grants read-write access to the whole MIB tree for both SNMPv1 and SNMPv2c.

This section provides the following information about SNMP support on Extreme Networks devices:

Versions Supported

Extreme Networks devices support three versions of SNMP:

- Version 1 (SNMPv1) — This is the initial implementation of SNMP. Refer to RFC 1157 for a full description of functionality.
- Version 2 (SNMPv2c) — The second release of SNMP, described in RFC 1907, has additions and enhancements to data types, counter size, and protocol operations.
- Version 3 (SNMPv3) — This is the most recent version of SNMP, and includes significant enhancements to administration and security. The major difference between SNMPv3 and earlier versions is that v3 provides a User-Based Security Model (USM) to associate users with managed access to security information. In addition to better security and better access control, SNMPv3 also provides a higher degree of reliability for notifying management stations when critical events occur.

  SNMPv3 is fully described in RFC 2571, RFC 2572, RFC 2573, RFC 2574, and RFC 2575.

SNMPv1 and v2c Network Management Components

The Extreme Networks implementation of SNMPv1 and v2c network management components fall into the following three categories:

- Managed devices (such as a switch).
- SNMP agents and MIBs, including SNMP traps, community strings, and Remote Monitoring (RMON) MIBs, which run on managed devices.

- SNMP network management applications, such as the Extreme Networks NetSight application, which communicate with agents to get statistics and alerts from the managed devices.

SNMPv3 User-Based Security Model (USM) Enhancements

SNMPv3 adds to v1 and v2c components by providing secure access to devices by authenticating and encrypting frames over the network. The Extreme Networks supported advanced security features provided in SNMPv3's User-Based Security Model are:

- Message integrity — Collects data securely without being tampered with or corrupted.
- Authentication — Determines the message is from a valid source.
- Encryption — Scrambles the contents of a frame to prevent it from being seen by an unauthorized source.

Unlike SNMPv1 and SNMPv2c, in SNMPv3, the concept of SNMP agents and SNMP managers no longer apply. These concepts have been combined into an SNMP entity. An SNMP entity consists of an SNMP engine and SNMP applications. An SNMP engine consists of the following four components:

  - Dispatcher — Sends and receives messages.
  - Message processing subsystem — Accepts outgoing PDUs from the dispatcher and prepares them for transmission by wrapping them in a message header and returning them to the dispatcher. Also accepts incoming messages from the dispatcher, processes each message header, and returns the enclosed PDU to the dispatcher.
  - Security subsystem — Authenticates and encrypts messages.
  - Access control subsystem — This component determines which users and which operations are allowed access to managed objects.

Terms and Definitions

Table 20-2 lists common SNMP terms and defines their use on Extreme Networks devices.

Table 20-2 SNMP Terms and Definitions

Term	Definition
community	A name string used to authenticate SNMPv1 and v2c users.
context	A subset of MIB information to which associated users have access rights.
engine ID	A value used by both the SNMPv3 sender and receiver to propagate inform notifications.
group	A collection of SNMP users who share the same access privileges.
inform	A notification message sent by an SNMPv3 agent to a network management station, a console, or a terminal to indicate the occurrence of a significant event, such as when a port or device goes up or down, when there are authentication failures, and when power supply errors occur.
MIB	Management Information Base, a repository for information about device parameters and network data organized in a tree structure.
notify profile	Associates target parameters to an SNMP notify filter to determine who should not receive SNMP notifications. This is useful for fine-tuning the amount of SNMP traffic generated.

Table 20-2 SNMP Terms and Definitions (continued)

Term	Definition
OID	Object Identifier, a unique ID distinguishing each variable in the MIB and is the means by which the SNMP manager and agent specify which managed elements are changed.
security level	The permitted level of security within a security model. The three levels of SNMP security are: • no authentication required (NoAuthNoPriv) • authentication required (AuthNoPriv) • privacy (authPriv)
security model	An authentication strategy that is set up for an SNMP user and the group in which the user resides. A combination of a security model and a security level determines which security mechanism is employed when handling an SNMP frame.
storage type	Specifies whether an SNMP user entry will be stored in volatile or nonvolatile memory.
taglist	A list of SNMP notify values that link a target (management station IP) address to specific SNMP notifications.
target address	A unique identifier and a specific IP address that will receive SNMP notification messages.
target parameters	A named set of security/authentication criteria used to generate a message to a target.
trap	A notification message sent by an SNMPv1 or v2c agent to a network management station, a console, or a terminal to indicate the occurrence of a significant event, such as when a port or device goes up or down, when there are authentication failures, and when power supply errors occur.
user	A person registered in SNMPv3 to access management information. In v1 and v2c, a user is set with the community name string.
USM	User-Based Security Model, the SNMPv3 authentication model which relies on a user name match for access to network management components.
VACM	View-based Access Control Model, which determines remote access to SNMP managed objects, allowing subsets of management information to be organized into user views.
view	Specifies permission for accessing SNMP MIB objects granted to a particular SNMP user group. View types and associated access rights are: • read - view-only access • write - allowed to configure MIB agent contents • notify - send trap messages

Security Models and Levels

An SNMP security model is an authentication strategy that is set up for a user and the group in which the user resides. A security level is the permitted level of security within a security model. The three levels of SNMP security on Extreme Networks devices are:

- No authentication required (NoAuthNoPriv)
- Authentication required (AuthNoPriv)
- Privacy (authPriv)

A combination of a security model and a security level determines which security mechanism is employed when handling an SNMP frame. Table 20-3 identifies the levels of SNMP security available on Extreme Networks devices and authentication required within each model.

Table 20-3 SNMP Security Models and Levels

Model	Security Level	Authentication	Encryption	How It Works
v1	NoAuthNoPriv	Community string	None	Uses a community string match for authentication.
v2c	NoAuthNoPriv	Community string	None	Uses a community string match for authentication.
v3 / USM	NoAuthNoPriv	User name	None	Uses a user name match for authentication.
	AuthNoPriv	MD5 or SHA	None	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms.
	authPriv	MD5 or SHA	DES	Provides authentication based on the HMAC-MD5 or HMAC-SHA algorithms. Provides DES 56-bit encryption in addition to authentication based on the CBC-DES (DES-56) standard.

Access Control

In addition to the Security Models and Levels described above, the Extreme Networks implementation of SNMP also provides a View-based Access Control Model (VACM), which determines remote access to managed objects. VACM allows you to organize subsets of management information into “views.” Management information that is in a user's view gives the user the corresponding access level to that management information: either read, write, or notify. Individual users can be organized into groups for whom you can pre-define what views are available based on the security model and security level used to request access. In this way, VACM allows you to permit or deny access to any individual item of management information depending on a user's group membership and the level of security provided by the communications channel.

Configuring SNMP

This section provides the following information about configuring SNMP on Extreme Networks devices:

Configuration Basics

Completing an SNMP configuration on an Extreme Networks device involves defining users who will be authorized to receive SNMP notifications about network events, associating security (target) parameters, access rights and MIB views to those users, and specifying an IP address where they will receive notifications. The basic steps in this process are:

1. Creating a name that will act as an SNMP user password:
   - This will be a **community** name for an SNMPv1 or v2c configuration, or
   - A **user** name for an SNMPv3 configuration.
2. Creating a group for the user named in Step 1.
3. Creating access rights for the user group named in Step 2.
4. Defining MIB view(s) for the user group.
5. Creating a target parameters entry to associate security and authorization criteria to the users created in Step 1.
6. Verifying if any applicable SNMP notification entries exist, or creating a new one. You will use this entry to send SNMP notification messages to the appropriate targets configured in Step 5.
7. Creating a target address entry to bind a management IP address to:
   - The notification entry and tag name created in Step 6, and
   - The target parameters entry created in Step 5.

Note: Commands for configuring SNMP on Extreme Networks devices are independent during the SNMP setup process. For instance, target parameters can be specified when setting up optional notification filters — even though these parameters have not yet been created with the **set snmp targetparams** command. The steps in this section are a guideline to configuring SNMP and do not necessarily need to be executed in this order.

How SNMP Processes a Notification Configuration

In order to send a trap or inform notification requested by a MIB code, the SNMP agent requires the equivalent of a trap “door”, a “key” to unlock the door, and a “procedure” for crossing the doorstep. To determine if all these elements are in place, the SNMP agent processes a device configuration as follows:

1. Determines if the “keys” for trap “doors” do exist. The key that SNMP is looking for is the notification entry created with the **set snmp notify** command.
2. Searches for the doors matching such a key and verifies that the door is available. If so, this door is tagged or bound to the notification entry. It was built using the **set snmp targetaddr** command, which specifies the management station IP address to which this door leads, and the “procedure” (**targetparams**) to cross the doorstep
3. Verifies that the description of how to step through the door is, in fact, there. The agent checks **targetparams** entries and determines this description was made with the **set snmp targetparams** command, which tells exactly which SNMP protocol to use and what community or user name to provide.
4. Verifies that the specified name, configured using either the **set snmp community** or **set snmp user** command is available.
5. Sends the notification message to the target address.

SNMP Defaults

Device Start Up Configuration

By default, SNMPv1 is configured on Extreme Networks switches. Table 20-4 lists the default configuration parameters, which include a single community name - public - granting read-write access to the whole MIB tree for both SNMPv1 and SNMPv2c.

Table 20-4 Default Extreme Networks SNMP Configuration

Parameter	Default Value
Community name	public
Group access privileges	rw (read-write)
Group user name	public
Security model	v1
Security access rights	all (for read, write, and notify access)
MIB view	all (entire MIB tree)

You can revise this default configuration by following the steps described in "Adding to or Modifying the Default Configuration" on page 20-10.

To take advantage of the advanced security and other features available in SNMPv3, it is recommended that you add to the Extreme Networks default configuration by configuring SNMPv3 as described in "Configuring SNMPv3" on page 20-11.

Refer also to "Configuring Secure SNMP Community Names" on page 20-18 for a description of a recommended configuration that will prevent unsecured access to SNMP information.

Configuring SNMPv1/SNMPv2c

Creating a New Configuration

Procedure 20-1 shows how to create a new SNMPv1 or SNMPv2c configuration. This example assumes that you haven't any preconfigured community names or access rights.

Note: The **v1** parameter in this example can be replaced with **v2** for SNMPv2c configuration.

Procedure 20-1 New SNMPv1/v2c Configuration

Step	Task	Command(s)
1.	Create a community name.	**set snmp community** *community* [**securityname** *securityname*] [**context** *context*] [**transport** *transport*] [**volatile** | **nonvolatile**]
2.	Create a security model (VACM) group using the *community name* you assigned in step 1.	**set snmp group** *groupname* **user** *communityname* **security-model v1**
3.	Set security access rights for the VACM group.	**set snmp access** *groupname* **security-model v1 read** *viewname* **write** *viewname* **notify** *viewname*
4.	Set MIB view attributes.	**set snmp view viewname** *viewname* **subtree** *subtree*

Procedure 20-1 New SNMPv1/v2c Configuration (continued)

Step	Task	Command(s)
5.	Specify the target parameters for SNMP notification message generation.	**set snmp targetparams** *paramset_name* **user** *community name* **security-model v1 message processing v1**
6.	Specify the target address to which SNMP notification messages generated using the specified target parameters will be sent.	**set snmp targetaddr** *targetaddr_name ipaddr* **param** *paramset_name* **taglist** *taglist*
7.	Specify a name for this notification entry and bind it to the target address.	**set snmp notify** *notify* **tag** *taglist*

Example

The following example displays an S-Series device configuration using the steps in Procedure 20-1. It shows how to:

- Create the community name **public**.
- Assign the **public** user to the group named **groupRW** and the SNMPv1 security model.
- Specify that, if SNMP messages are received with the **public** name string, the view **RW for** read requests, write requests, and notify requests will be applied to this user.
- For the view **RW**, include the MIB subtree denoted with OID **1** and **0.0,** and exclude view access to subtree denoted with OID **1.3.6.1.6.3.13.1** (which is the notification MIB).
- Assign a target parameters entry**, TVv1public**, for security level processing to the **public** community name.
- Create a target address entry named **TVTrap** at IP address **10.42.1.10,** which will use security and authorization criteria contained in the target parameters entry called **TVv1public,** and bind these parameters together with a tag entry called **TVTrapTag**.

```
S Chassis(su)->set snmp community public
S Chassis(su)->set snmp group groupRW user public security model v1
S Chassis(su)->set snmp access groupRW security-model v1 read RW write RW notify RW
S Chassis(su)->set snmp view viewname RW subtree 1
S Chassis(su)->set snmp view viewname RW subtree 0.0
S Chassis(su)->set snmp view viewname RW subtree 1.3.6.1.6.3.13.1 excluded
S Chassis(su)->set snmp targetparams TVv1public user public security-model v1
message             processing v1
S Chassis(su)->set snmp targetaddr TVTrap 10.42.1.10 param TVv1public taglist
           TVTraptag
S Chassis(su)->set snmp notify TVTrap tag TVTrapTag
```

Adding to or Modifying the Default Configuration

By default, SNMPv1 is configured on Extreme Networks switches. A single community name - public - is configured, which grants read-write access to the whole MIB tree for both SNMPv1 and SNMPv2c.

The beginning command sequence in the default configuration is similar to the first part of the previous example. It looks like this:

```
S Chassis(su)->set snmp community public
S Chassis(su)->set snmp group groupRW user public security-model v1
```

```
S Chassis(su)->set snmp access groupRW security-model v1 read All write All notify
All
S Chassis(su)->set snmp view viewname All subtree 1
```

Note: Any use of the parameter 'All' must be exactly as shown in this example. Any other variation (including, but not limited to, values such as 'all' or 'ALL') will not be valid.

You can modify this default configuration as shown in the following examples.

Adding a New Community Name

Use these commands to add a new SNMPv1 community name called **newname with the same** permissions as the default configuration:

```
S Chassis(su)->set snmp community newname
S Chassis(su)->set snmp group groupRW user newname security-model v1
```

Use this command to remove the **public** community name from the default configuration:

```
S Chassis(su)->clear snmp community public
```

Note: You can leave the **set snmp group groupRW user public security-model v1** statement in the default configuration in case you want to re-activate the **public** community name at some point, or can clear it as well.

Refer to "Configuring Secure SNMP Community Names" on page 20-18 for a description of a recommended configuration that will prevent unsecured access to SNMP information.

Configuring SNMPv3

Procedure 20-2 shows how to complete a basic SNMPv3 configuration.

Procedure 20-2 SNMPv3 Configuration

Step	Task	Command(s)
1.	Create an SNMPv3 user and specify authentication, encryption, and security credentials. • If **remote** is not specified, the user will be registered for the local SNMP engine. • If **authentication** is not specified, no authentication will be applied. • If **privacy** is not specified, no encryption will be applied.	**set snmp user** *user* [**remote** *remoteid*] [**authentication** {**md5** | **sha**}] [*authpassword*] [**privacy** *privpassword*]
2.	Create a user group and add the user created in Step 1. • If storage type is not specified, **nonvolatile** will be applied.	**set snmp group** *groupname* **user** *user* **security-model usm** [**volatile** | **nonvolatile**]

Procedure 20-2 SNMPv3 Configuration (continued)

Step	Task	Command(s)
3.	Set security access rights for the group. • If security level is not specified, no authentication will be applied. • Only one context, the “default context”, is supported in this release. There is no need to configure this parameter. • If **read** view is not specified none will be applied. • If **write** view is not specified, none will be applied. • If **notify** view is not specified, none will be applied. • If storage type is not specified, entries will be stored as permanent and will be held through device reboot.	**set snmp access** *groupname* **security-model usm** [**noauthentication** | **authentication** | **privacy**] [**exact** | **prefix**] [**read** *readviewname*] [**write** *writeviewname*] [**notify** *notifyviewname*] [**volatile** | **nonvolatile**]
4.	Define views created in Step 3. • If not specified, **mask** will be set to empty. • If not specified, subtree use will be **included**. • If storage type is not specified, **nonvolatile** (permanent) will be applied.	**set snmp view viewname** *viewname* **subtree** *subtree* [**mask** *mask*] [**included** | **excluded**] [**volatile** | **nonvolatile**]
5.	Set SNMP target parameters. • If not specified, security level will be set to **noauthentication.** • If not specified, storage type will be set to **nonvolatile**.	**set snmp targetparams** *paramset_name* **user** *user* **security-model usm message-processing v3** [**noauthentication** | **authentication** | **privacy**] [**volatile** | **nonvolatile**]
6.	Set the SNMP target address for notification message generation. • If not specified, *udpport* will be set to **162**. • If not specified, *mask* will be set to **255.255.255.255**. • If not specified, *timeout* will be set to **1500** (15 seconds). • If not specified, number of *retries* will be set to **3**. • If **taglist** is not specified, none will be set. • If not specified, storage type will be **nonvolatile**.	**set snmp targetaddr** *targetaddr_name ipaddr* **param** *paramset_name* [**udpport** *udpport*] [**mask** *mask*] [**timeout** *timeout*] [**retries** *retries*] [**taglist** *taglist*] [**volatile** | **nonvolatile**]
7.	Set SNMP notification parameters. • If not specified, message type will be set to **trap**. • If not specified, storage type will be set to **nonvolatile**.	**set snmp notify** *notify* **tag** *tag* [**trap** | **inform**] [**volatile** | **nonvolatile**]

The following example is an S-Series device configuration using the steps in Procedure 20-2. It shows how to:

- Create the user **Extremenetworks_user**, specifying authentication, encryption, and security credentials.
- Assign **Extremenetworks_user** to the **Extremenetworks** group and associate it to the SNMPv3 security model, **usm**.
- Specify that, if SNMP messages are received with authentication and encryption, the view, **readView** for read requests, and the view writeView for write requests will be applied to this user group based on the USM security model.
- For the view **writeView**, include the MIB subtree denoted with OID **1,** and exclude the subtree denoted by OID **1.3.6.1.4.1.5624.1.2.16**.
- Assign an SNMPv3 target parameters entry named **matrixn** to the **Extremenetworks_user** using the USM security model.
- Create a target address entry named **Extreme_Networks** at IP address **172.29.10.1 which will** use security and authorization criteria contained in a target parameters entry called **matrixn,** and bind these parameters together with a tag entry called **v3TrapTag**.

```
S Chassis(su)->set snmp user Extremenetworks_user authentication md5
my_authentication             privacy my_privacy

S Chassis(su)->set snmp group Extremenetworks user Extremenetworks_user
security-model usm

S Chassis(su)->set snmp access Extremenetworks security-model usm privacy read
readView             write writeView

S Chassis(su)->set snmp view viewname readView subtree 1

S Chassis(su)-> set snmp view viewname writeView subtree 1

S Chassis(su)-> set snmp view viewname writeView subtree 1.3.6.1.4.1.5624.1.2.16
             excluded

S Chassis(su)-> set snmp targetparams matrixn user Extremenetworks_user
security-model usm              message-processing v3

S Chassis(su)-> set snmp targetaddr Extreme_Networks 172.29.10.1 param matrixn
             taglist v3TrapTag

S Chassis(su)->set snmp notify SNMPv3TrapGen tag v3TrapTag inform
```

How SNMP Will Process This Configuration

As described in "How SNMP Processes a Notification Configuration" on page 20-8, if the SNMP agent on the device needs to send an inform message, it looks to see if there is a notification entry that says what to do with inform messages. Then, it looks to see if the tag list (**v3TrapTag**) specified in the notification entry exists. If it exists, then the inform message is sent to the target addresses specified by the tag list, (**Extreme_Networks**) using the parameters specified for each address (**matrixn**).

Configuring an SNMPv3 Inform or Trap Engine ID

This section provides additional information for configuring SNMPv3 inform or trap notifications. The steps in Procedure 20-3 on page 20-14 add to the following configuration example:

```
S Chassis(su)->set snmp view viewname All subtree 1
S Chassis(su)->set snmp user v3user authentication md5 md5passwd privacy despasswd
S Chassis(su)->set snmp group v3group user v3user security-model usm
S Chassis(su)->set snmp access v3group security-model usm privacy exact read All
write All notify All
S Chassis(su)->set snmp notify v3notify tag v3tag inform
```

```
S Chassis(su)->set snmp targetaddr v3TA 134.141.209.73 param v3TP taglist v3tag
S Chassis(su)->set snmp targetparams v3TP user v3user security-model usm
message-processing v3 privacy
```

Inform EngineIDs

In the Extreme Networks SNMP implementation, the receiver's EngineID value is used by both the sender and receiver to propagate inform notifications. In order to send and receive SNMP v3 informs in their most secure form (with authentication and privacy enabled), you must configure a user ID and corresponding receiver EngineID on the sender as shown in the example in Procedure 20-3. This example assumes that NetSight Console is the receiver, and an S-Series switch is the sender.

Note: The following file location and EngineID are provided as examples. Your settings will vary.

Procedure 20-3 adds to the configuration example shown in "Configuring an SNMPv3 Inform or Trap Engine ID" on page 20-13.

Procedure 20-3 Configuring an EngineID

Step	Task	Command(s)
1.	If necessary, create an SNMP3 configuration.	Refer to "Configuring an SNMPv3 Inform or Trap Engine ID" on page 20-13.
2.	On the management station, navigate to and display the Netsight Console SNMP trap configuration file.	**C:\Program Files\Extreme Networks\NetSight Shared\snmptrapd.conf**
3.	Determine the EngineID from this line in the configuration file.	**oldEngineID 0x800007e5804f190000d232aa40**
4.	On the Matrix N, define the same user as in the above example (**v3user**) with this EngineID and with the same Auth/Priv passwords you used previously.	**set snmp user v3user remote 800007e5804f190000d232aa40 authentication md5 md5passwd privacy despasswd**  **Note:** You can omit the **0x** from the EngineID. You can also use the colon notation like this: 80:00:07:e5:80:4f:19:00:00:d2:32:aa:40
5.	Navigate to and display the user configuration on the management station. (This assumes that you have already created the user in Netsight Console, so you will only need to add it to the configuration file of the trap daemon.)	**C:\Program Files\Extreme Networks\NetSight Console\Bin\snmptrapd.conf**
6.	Using any plain text editor, add this line to the configuration file.	**createuser v3user MD5 md5passwd DES despasswd**

Trap EngineID

To use traps instead of inform notifications, you would change the preceding configuration as follows:

1. Use this command to specify trap notifications:

   ```
   set snmp notify v3notify tag v3tag trap
   ```

2. Verify that the "createuser" entry in the NetSight Console SNMP trap configuration looks like this:

```
createuser -e 0x800015f80300e06314d79c v3user MD5 md5passwd DES
despasswd
```

When you are finished modifying the configuration, save the file and restart the SNMP Trap Service using Netsight Services Manager.

Note: When installed on a Unix platform, the NetSight server must be manually restarted.

Configuring an SNMP View

It is possible to include certain OIDs and exclude certain other OIDs within one SNMP MIB view. You do this by stacking different set snmp view includes and excludes which specify a single view name. This allows the user to view all of the “included” OID strings for their associated view name, minus all of the “excluded” OID strings for their view name. If no such parameter is specified, “included” is assumed.

Though it is possible to create and use multiple view names as desired, for demonstration purposes it is simplest to modify the default view, since it is already being referenced by the remainder of the SNMP command set.

The following example removes the default view specifications, and inserts one which permits access to branch MIB **1.3.6.1.2.1** with the exception of branch interfaces **1.3.6.1.2.1.2**.:

```
S Chassis(su)->clear snmp view All 1
S Chassis(su)->clear snmp view All 0.0
S Chassis(su)->set snmp view viewname All subtree 1.3.6.1.2.1
S Chassis(su)->set snmp view viewname All subtree 1.3.6.1.2.1.2 excluded
S Chassis(su)->show snmp view
 View Name       = All
 Subtree OID     = 1.3.6.1.2.1
 Subtree mask    =
 View Type       = included
 Storage type    = nonVolatile
 Row status      = active

 View Name       = All
 Subtree OID     = 1.3.6.1.2.1.2
 Subtree mask    =
 View Type       = excluded
 Storage type    = nonVolatile
 Row status      = active
```

You can test this configuration using any MIB browser directed to the IP of the configured device and using the default community name **public** associated with the view **All.** If configured correctly, only your specified sections of the MIBs will be visible.

Configuring the Optional Mask Parameter

Note: The mechanics of determining exactly how to configure the optional mask parameter make for an inefficient use of time if you will only be using the query once. However, for data retrieved repeatedly, using the method described in the following examples can prevent the unnecessary transfer of much SNMP data over your network.

As defined in RFC2575, an SNMP mask is an optional parameter of the set snmp view command. You can use a mask to modify a view inclusion, designating certain octets of an OID string as wild-card “don't care” values. Once defined, you can view within a MIB branch (using a MIB browser such as that offered within the NetSight suite of products) only those leaves associated with specific items, such as designated port numbers, MAC addresses, and IP addresses.

For example, the RMON Statistics MIB branch is defined as follows, with the leaves defined within that branch each having multiple iterations, one for each port.

```
                etherStatsEntry=1.3.6.1.2.1.16.1.1.1
                etherStatsIndex=1.3.6.1.2.1.16.1.1.1.1.<port>
           etherStatsDataSource=1.3.6.1.2.1.16.1.1.1.2.<port>
           etherStatsDropEvents=1.3.6.1.2.1.16.1.1.1.3.<port>
               etherStatsOctets=1.3.6.1.2.1.16.1.1.1.4.<port>
                 etherStatsPkts=1.3.6.1.2.1.16.1.1.1.5.<port>
       etherStatsBroadcastPkts=1.3.6.1.2.1.16.1.1.1.6.<port>
       etherStatsMulticastPkts=1.3.6.1.2.1.16.1.1.1.7.<port>
      etherStatsCRCAlignErrors=1.3.6.1.2.1.16.1.1.1.8.<port>
       etherStatsUndersizePkts=1.3.6.1.2.1.16.1.1.1.9.<port>
        etherStatsOversizePkts=1.3.6.1.2.1.16.1.1.1.10.<port>
           etherStatsFragments=1.3.6.1.2.1.16.1.1.1.11.<port>
             etherStatsJabbers=1.3.6.1.2.1.16.1.1.1.12.<port>
           etherStatsCollisions=1.3.6.1.2.1.16.1.1.1.13.<port>
         etherStatsPkts64Octets=1.3.6.1.2.1.16.1.1.1.14.<port>
   etherStatsPkts65to127Octets=1.3.6.1.2.1.16.1.1.1.15.<port>
  etherStatsPkts128to255Octets=1.3.6.1.2.1.16.1.1.1.16.<port>
  etherStatsPkts256to511Octets=1.3.6.1.2.1.16.1.1.1.17.<port>
 etherStatsPkts512to1023Octets=1.3.6.1.2.1.16.1.1.1.18.<port>
etherStatsPkts1024to1518Octets=1.3.6.1.2.1.16.1.1.1.19.<port>
               etherStatsOwner=1.3.6.1.2.1.16.1.1.1.20.<port>
              etherStatsStatus=1.3.6.1.2.1.16.1.1.1.21.<port>
```

As shown in the example output above, when displaying the etherStatsEntry branch, all ports are listed for each leaf before moving on to the ports of the next leaf as the result of listing all of the data in numeric OID order.

Here is an abbreviated example of one such SNMP query.

```
Object                          Instance  Type          Value
etherStatsIndex                 1001      INTEGER       1001
etherStatsIndex                 1518      INTEGER       1518
etherStatsDataSource            1001      OBJECT ID     1.3.6.1...11001
etherStatsDataSource            1518      OBJECT ID     1.3.6.1...12006
etherStatsStatus                1001      INTEGER       valid(1)
etherStatsStatus                1518      INTEGER       valid(1)
```

Example

This example shows you how to use the mask parameter to significantly refine your query output, so that only data for specified ports is returned. For this example, assume that S-Series slot 1 port 12 is of interest.

The first ten octets of the etherStatsEntry (1.3.6.1.2.1.16.1.1.1) must match exactly as specified. The next octet, representing each of the 21 possible leaves within that branch, need not match exactly. The remainder, representing the port number, must match exactly as specified.

The bit representations for this would be 11111111-11011111, or 0xffdf. If the actual OID string being masked is longer than the specified bits, the missing bits to the right are assumed to be 1's. It is thus only necessary to make the mask long enough (in increments of 8-bit bytes) to designate, with a 0 bit, any desired “wild-card” OID string octets.

The following is an SNMP View using these specifications, starting with a default configuration.

```
S Chassis(su)->show snmp view
 View Name       = All
 Subtree OID     = 1
 Subtree mask    =
 View Type       = included
 Storage type    = nonVolatile
```

```
 Row status      = active

 View Name       = All
 Subtree OID     = 0.0
 Subtree mask    =
 View Type       = included
 Storage type    = nonVolatile
 Row status      = active

S Chassis(su)->clear snmp view All 1
S Chassis(su)->set snmp view viewname All subtree 1.3.6.1.2.1.16.1.1.1.0.1012 mask
ff:df
S Chassis(su)->show snmp view
 View Name       = All
 Subtree OID     = 0.0
 Subtree mask    =
 View Type       = included
 Storage type    = nonVolatile
 Row status      = active

 View Name       = All
 Subtree OID     = 1.3.6.1.2.1.1.1.0.244
 Subtree mask    = ff:df
 View Type       = included
 Storage type    = nonVolatile
 Row status      = active
```

You can see by the unexpected Subtree OID value that this view actually accommodates only the right-most 8 bits of the entered decimal value 1012. The hexadecimal equivalent is 0x3f4, and the decimal equivalent of 0xf4 is 244. It is therefore true that this defined subtree will get a “hit” on multiple port values (244, 500, 756, 1012, etc), should they exist. This has nothing to do with the mask, and everything to do with the reasonable limitations of MIB design.

Note: Any use of the **mask** parameter assumes the View Type is configured as i**ncluded**. Parameters **included** or **excluded** cannot be specified along with the **mask** parameter.

An SNMP query of the etherStatsEntry branch using the community name associated with this defined view would display a result similar to the following.

```
 Object                          Instance  Type          Value
 etherStatsIndex                 1012      INTEGER       1012
 etherStatsDataSource            1012      OBJECT ID     1.3.6.1...11012
 etherStatsDropEvents            1012      Counter       54323
 etherStatsOctets                1012      Counter       302877211
 etherStatsPkts                  1012      Counter       1592774
 etherStatsBroadcastPkts         1012      Counter       793487
 etherStatsMulticastPkts         1012      Counter       729406
 etherStatsCRCAlignErrors        1012      Counter       0
 etherStatsUndersizePkts         1012      Counter       0
 etherStatsOversizePkts          1012      Counter       0
 etherStatsFragments             1012      Counter       0
 etherStatsJabbers               1012      Counter       0
 etherStatsCollisions            1012      Counter       0
 etherStatsPkts64Octets          1012      Counter       0
 etherStatsPkts65to127Octets     1012      Counter       458931
 etherStatsPkts128to255Octets    1012      Counter       55190
 etherStatsPkts256to511Octets    1012      Counter       656909
 etherStatsPkts512to1023Octets   1012      Counter       57
 etherStatsPkts1024to1518Octets  1012      Counter       1
```

```
etherStatsOwner                 1012      OCTET STRING  monitor
etherStatsStatus                1012      INTEGER       valid(1)
```

Configuring Secure SNMP Community Names

Procedure 20-4 provides an example of a recommended configuration that will prevent unsecured SNMPv1/v2c access of potentially security compromising information.

As discussed previously in this document, SNMP v1 and v2c are inherently insecure device management protocols. Community names used to define access levels are passed in clear text in all protocol frames sent to the managed entity and may be visible by read-only SNMP users when querying certain SNMP configuration-related objects. In addition, you may be further exposing your network due to configuration conventions which reuse the community names in other aspects of entity management, such as CLI login passwords, and SNMP security names.

Extreme Networks recommends that you "secure" all SNMP community names. You do this by creating a configuration that hides, through the use of "views" sensitive information from SNMP v1/v2c users as follows:

Procedure 20-4 Configuring Secure Community Names

Step	Task	Command(s)
1.	Create the following SNMP view group configurations. • An admin (v3) view group with secure read, write, and notify access • A read-only view group with unsecure (v1 and v2c) access • A read-write view group with unsecure (v1 and v2c) access	**set snmp access** *admin-groupname* **security-model usm privacy exact read** *secured-viewname* **write** *secure-viewname* **notify** *secured-viewname* **set snmp access** *read-only-groupname* **security-model v1 exact read** *unsecured-viewname* **set snmp access** *read-only-groupname* **security-model v2c exact read** *unsecured-viewname* **set snmp access** *read-write-groupname* **security-model v1 exact read** *unsecure-viewname* **write** *unsecured-viewname* **set snmp access** *read-write-groupname* **security-model v2c exact read** *unsecured-viewname* **write** *unsecured-viewname*
2.	Create v1/v2c "public" and "private" community names and security names.	**set snmp community** *private-communityname* **securityname** *read-write-securityname* **set snmp community** *public-communityname* **securityname** *read-only-securityname*
3.	Create user groups and bind them to the security names created in Step 2.	**set snmp group** *admin-groupname* **user** *admin-username* **set snmp group** *read-only-groupname* **user** r*ead-only-securityname* **security-model v1** **set snmp group** *read-write-groupname* **user** *read-write-securityname* **security-model v1** **set snmp group** *read-only-groupname* **user** *read-only-securityname* **security-model v2c** **set snmp group** *read-write-groupname* **user** *read-write-securityname* **security-model v2c**

Procedure 20-4 Configuring Secure Community Names (continued)

Step	Task	Command(s)
4.	Using the *admin-username* assigned in Step 3, create the v3 user and define authentication keys.	**set snmp user** *admin-username* **authentication sha** *auth-key* **privacy** *priv-key*
5.	Using the viewnames assigned in Step 1, create restricted views for v1/v2c users, and unrestricted views for v3 users.	**set snmp view viewname** *secured-viewname* **subtree 1** **set snmp view viewname** *secured-viewname* **subtree 0.0** **set snmp view viewname** *unsecured-viewname* **subtree 1** **set snmp view viewname** *unsecured-viewname* **subtree 0.0**
6.	Exclude the following from the restricted view • snmpUsmMIB (which contains v3 user names, but no passwords) • snmpVacmMIB (which contains SNMP view configurations) • snmpCommunityTable (which contains community names)	**set snmp view viewname** *unsecured-viewname* **subtree 1.3.6.1.6.3.15 excluded** **set snmp view viewname** *unsecured-viewname* **subtree 1.3.6.1.6.3.16 excluded** **set snmp view viewname** *unsecured-viewname* **subtree 1.3.6.1.6.3.18.1.1 excluded**

Example

The following example shows an S-Series device configuration using the steps in Procedure 20-4.

```
S Chassis(su)->set snmp access gAdmin security-model usm privacy exact read
vSecured write vSecured notify vSecured
S Chassis(su)->set snmp access gReadOnlyV1V2C security-model v1   exact read
vUnsecured
S Chassis(su)->set snmp access gReadOnlyV1V2C security-model v2c  exact read
vUnsecured
S Chassis(su)->set snmp access gReadWriteV1V2C security-model v1  exact read
vUnsecured write vUnsecured
S Chassis(su)->set snmp access gReadWriteV1V2C security-model v2c exact read
vUnsecured write vUnsecured
S Chassis(su)->set snmp community cnPrivate securityname sn_v1v2c_rw
S Chassis(su)->set snmp community cnPublic securityname sn_v1v2c_ro
S Chassis(su)->set snmp group gReadOnlyV1V2C user sn_v1v2c_ro security-model v1
S Chassis(su)->set snmp group gReadWriteV1V2C user sn_v1v2c_rw security-model v1
S Chassis(su)->set snmp group gReadOnlyV1V2C user sn_v1v2c_ro security-model v2c
S Chassis(su)->set snmp group gReadWriteV1V2C user sn_v1v2c_rw security-model v2c
S Chassis(su)->set snmp group gAdmin user it-admin security-model usm
S Chassis(su)->set snmp user it-admin authentication sha auth_key privacy priv_key
S Chassis(su)->set snmp view viewname vSecured subtree 1
S Chassis(su)->set snmp view viewname vSecured subtree 0.0
S Chassis(su)->set snmp view viewname vUnsecured subtree 1
S Chassis(su)->set snmp view viewname vUnsecured subtree 0.0
S Chassis(su)->set snmp view viewname vUnsecured subtree 1.3.6.1.6.3.15 excluded
```

```
S Chassis(su)->set snmp view viewname vUnsecured subtree 1.3.6.1.6.3.16 excluded
S Chassis(su)->set snmp view viewname vUnsecured subtree 1.3.6.1.6.3.18.1.1
excluded
```

Reviewing SNMP Settings

Use the **show** commands described in this section to review SNMP settings.

Community

Use this command to display SNMPv1/SNMPv2c community names and status. In SNMPv1 and v2, community names act as passwords to remote management.

show snmp community [*name*]

Example

```
S Chassis(su)->show snmp community public
Name            = public
Security name   = public
Context         =
Transport tag   =
Storage type    = nonVolatile
Status          = active
```

Context

Use this command to display the context list configuration for SNMP view-based access control:

show snmp context

Example

```
S Chassis(su)->show snmp context
--- Configured contexts:
default context (all MIBs)
router
```

Counters

Use this command to display SNMP traffic counter values:

show snmp counters

Example

```
S Chassis(su)->show snmp counters

--- mib2 SNMP group counters:
snmpInPkts              = 396601
snmpOutPkts             = 396601
snmpInBadVersions       = 0
snmpInBadCommunityNames = 0
snmpInBadCommunityUses  = 0
snmpInASNParseErrs      = 0
snmpInTooBigs           = 0
snmpInNoSuchNames       = 0
snmpInBadValues         = 0
snmpInReadOnlys         = 0
snmpInGenErrs           = 0
snmpInTotalReqVars      = 403661
snmpInTotalSetVars      = 534
snmpInGetRequests       = 290
snmpInGetNexts          = 396279
snmpInSetRequests       = 32
snmpInGetResponses      = 0
snmpInTraps             = 0
snmpOutTooBigs          = 0
snmpOutNoSuchNames      = 11
snmpOutBadValues        = 0
snmpOutGenErrs          = 0
snmpOutGetRequests      = 0
snmpOutGetNexts         = 0
snmpOutSetRequests      = 0
snmpOutGetResponses     = 396601
snmpOutTraps            = 0
snmpSilentDrops         = 0
snmpProxyDrops          = 0
--- USM Stats counters:
```

```
usmStatsUnsupportedSecLevels = 0
usmStatsNotInTimeWindows     = 0
usmStatsUnknownUserNames     = 0
usmStatsUnknownEngineIDs     = 0
usmStatsWrongDigests         = 0
usmStatsDecryptionErrors     = 0
```

Engineid

Use this command to display SNMP engine properties:

show snmp engineid

Example

```
S Chassis(su)->show snmp engineid

EngineId: 80:00:15:f8:03:00:e0:63:9d:b5:87
Engine Boots    = 12
Engine Time     = 162181
Max Msg Size    = 2048
```

Groups

Use this command to display SNMP group information. If no parameters are specified, all information about all groups is displayed.

show snmp group [**groupname** *groupname*] [**user** *user*] [**security-model** {**v1** | **v2c** | **usm**}] [**volatile** | **nonvolatile** | **read-only**]

Example

```
S Chassis(su)->show snmp group
Security model          = SNMPv1
Security/user name      = public
Group name              = groupRW
Storage type            = nonVolatile
Row status              = active

Security model          = SNMPv2c
Security/user name      = public
Group name              = groupRW
Storage type            = nonVolatile
Row status              = active

Security model          = USM
Security/user name      = admin1
Group name              = alladmin
Storage type            = nonVolatile
Row status              = active
```

```
Security model          = USM
Security/user name      = admin2
Group name              = alladmin
Storage type            = nonVolatile
Row status              = active
```

Group Access Rights

Use this command to display an SNMP group's access rights. If no parameters are entered, access information about all groups is displayed.

show snmp access [*groupname*] [**security-model** {**v1** | **v2c** | **usm**}] [**noauthentication** | **authentication** | **privacy**] [**context** *context*] [**volatile** | **nonvolatile** | **read-only**]

Example

```
S Chassis(su)->show snmp access
Group           = groupRW
Security model  = SNMPv1
Security level  = noAuthNoPriv
Read View       = All
Write View      = All
Notify View     = All
Context match   = "default context" (exact)
Storage type    = nonVolatile
Row status      = active

Group           = groupRW
Security model  = SNMPv2c
Security level  = noAuthNoPriv
Read View       = All
Write View      = All
Notify View     = All
Context match   = "default context" (exact)
Storage type    = nonVolatile
Row status      = active
```

Target Parameter Profiles

Use this command to display SNMP target parameter profiles. If no parameters are specified, information for all target parameter profiles is displayed.

show snmp targetparams [*targetParams*] [**volatile** | **nonvolatile** | **read-only**]

Example

```
S Chassis(su)-> show snmp targetparams matrix
```

```
Target Parameter Name = matrix
Security Name         = Extremenetworks_user
Message Proc. Model   = USM
Security Level        = authNoPriv
Storage type          = nonVolatile
Rox status            = active
```

Target Address Profiles

Use this command to display SNMP target address information. If no parameters are entered, information about all target address profiles is displayed.

show snmp targetaddr [*targetAddr*] [**volatile** | **nonvolatile** | **read-only**]

Example

```
S Chassis(su)-> show snmp targetaddr
Target Address Name = Extremenetworks_user
Tag List            =
IP Address          = 172.29.10.1
UDP Port#           = 162
Target Mask         = 255.255.255.255
Timeout             = 1500
Retry count         = 3
Parameters          = matrix
Storage type        = nonVolatile
Row status          = active
```

Notify

Use this command to display the SNMP notify configuration, which determines which management targets will receive SNMP notifications. If no parameters are entered, information about all notify configurations is displayed.

show snmp notify [*notify*] [**volatile** | **nonvolatile** | **read-only**]

Example

```
S Chassis(su)->show snmp notify
Notify name     = 1
Notify Tag      = Console
Notify Type     = trap
Storage type    = nonVolatile
Status          = active

Notify name     = 2
Notify Tag      = TrapSink
Notify Type     = trap
Storage type    = nonVolatile
Status          = active
```

Notify Filter

Use this command to display SNMP notify filter information, identifying which profiles will not receive SNMP notifications:

show snmp notifyfilter [*profile*] [**subtree** *oid-or-mibobject*] [**volatile** | **nonvolatile** | **read-only**]

Example

```
S Chassis(su)->show snmp notifyfilter
Profile         = pilot1
Subtree         = 1.3.6
Subtree mask
Filter type     = included
Storage type    = nonVolatile
Row status      = active
```

Notify Profile

Use this command to display SNMP notify profile information:

show snmp notifyprofile [*profile*] [**targetparam** *targetparam*] [**volatile** | **nonvolatile** | **read-only**]

Example

```
S Chassis(su)->show snmp notifyprofile area51
Notify Profile  = area51
TargetParam     = v3ExampleParams
Storage type    = nonVolatile
Row status      = active
```

Users

Use this command to display SNMPv3 users:

show snmp user [**list**] | [*user*] | [**remote** *remote*] [**volatile** | **nonvolatile** | **read-only**]

Example

```
S Chassis(su)->show snmp user Extremenetworks_user

EngineId                = 80:00:15:f8:03:00:e0:63:9d:cb:89
Username                = Extremenetworks_user
Auth protocol           = usmHMACMD5AuthProtocol
Privacy protocol        = usmDESPrivProtocol
Storage type            = nonVolatile
Row status              = active
```

Views

Use this command to display SNMP views. If no parameters are entered, all view information is displayed.

show snmp view [*viewname*] [**subtree** *oid-or-mibobject*] [**volatile** | **nonvolatile** | **read-only**]

Example

```
S Chassis(su)->show snmp view readView


View Name       = readView
Subtree OID     = 1
Subtree mask    =
View Type       = included
Storage type    = nonVolatile
Row status          = active
```

21

Spanning Tree Configuration

This chapter provides the following information about configuring and monitoring Spanning Tree protocols on Extreme Networks S-Series devices:

What Is the Spanning Tree Protocol?

The Spanning Tree Protocol (STP) resolves the problem of physical loops in a network by establishing one primary path between any two devices. Duplicate paths are barred from use and become standby or “blocked” paths until the primary path fails, at which point the redundant path can be brought into service.

STP operates by forming a fully connected tree of data loop free LAN connected bridges (switches) through the exchange of Bridge Protocol Data Units (BPDUs). Each bridge port transmits BPDUs on a periodic basis. The information contained in the BPDU is used by the receiving bridge to calculate a port role for each bridge port. There is one bridge in the network chosen to be the root bridge, based on its bridge ID. Ports that directly connect bridges to the root bridge or are connected through another bridge are assigned one of four roles:

- Root Port – The best path to the root
- Designated Port – Ports which either provide a path to the root for other bridges or connect end users
- Backup Port – A port attached to a LAN where another port of the same bridge is a designated port. This backup port takes over the designated role should the LAN’s designated port become disabled

- Alternate Port – Port providing a path to the root that is not root, designated, or backup

For a summary of port roles, see Table 21-1 on page 21-14.

While the network is in a steady state, alternate and backup ports are in blocking state; root and designated ports are in forwarding state.

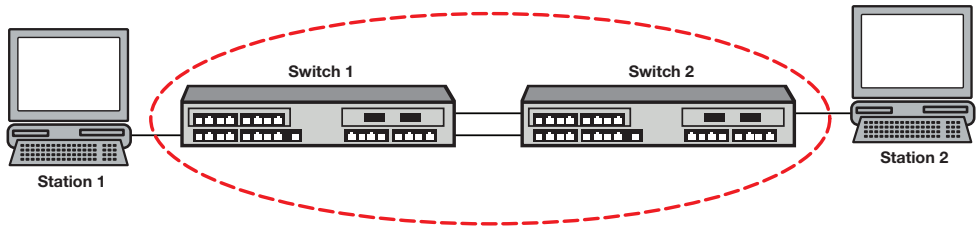
STP allows for the automatic reconfiguration of the network. When bridges are added to or removed from the network, root election takes place and port roles are recalculated.

Why Would I Use Spanning Trees in My Network?

Redundant links must be factored into even the simplest of topologies to protect against data loss and downtime due to any single point of failure. STP prevents redundant links from forming data loops which would consume all available network bandwidth. STP manages redundant links by keeping them in a blocking state and automatically unblocking them when changes in topology require that they be used. See Table 21-2 on page 21-14 for a summary of Spanning Tree port states.

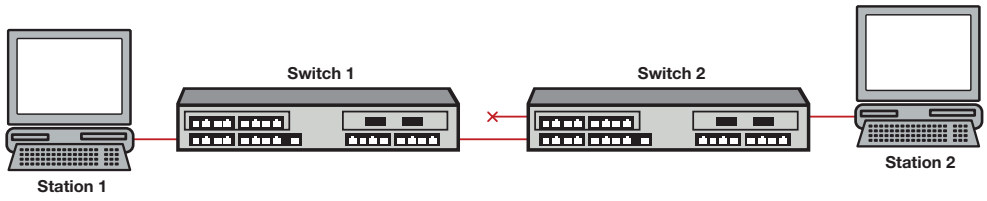
As shown in Figure 21-1, a planned redundant link between Switch 1 and Switch 2 makes it possible for a bridging loop to occur. If Station 1 transmits a multicast or broadcast packet to Station 2 in this scenario, the packet would continue to circulate endlessly between both switching devices. Without Spanning Tree blocking one of the links, there would be nothing at layer 2 to stop this loop from happening and unnecessarily consuming network resources. As administrator, you would be forced to manually disable one of the links between Switch 1 and 2 for the Figure 21-1 network to operate.

Figure 21-1 Redundant Link Causes a Loop in a Non-STP Network



STP automatically blocks redundant paths, as shown in Figure 21-2. In the event that the primary (unblocked) path fails, STP places the blocked path into service. If the original primary path recovers, the redundant path will once again block and the primary path will be used.

Figure 21-2 Loop Avoided When STP Blocks a Duplicate Path



How Do I Implement Spanning Trees?

By default, Spanning Tree is both enabled globally and on all ports. The design of the Spanning Tree protocol and the default configuration values on these devices make user configuration unnecessary in order to add redundant ports to your network. You will want to make configuration changes to select a root bridge, take advantage of Multiple Spanning Tree, or use any of the advanced features described below. Before configuring STP it is important to understand how it works.

STP Overview

Extreme Networks switch devices support the Spanning Tree Protocol (STP), Rapid Spanning Tree Protocol (RSTP), and Multiple Spanning Tree Protocol (MSTP) as defined in the following standards and described in IEEE 802.1Q:

- IEEE 802.1D (Spanning Tree Protocol)
- IEEE 802.1w (Rapid Spanning Tree Protocol)
- IEEE 802.1s (Multiple Spanning Tree Protocol)
- IEEE 802.1t (Update to 802.1D)

STP forms a network of bridges connected by LANs into a tree that is:

- Predictable – A given set of configured bridges always yields the same topology when the network reaches steady state
- Optimized – STP selects the best path to the root bridge
- Fully connected – Each bridge communicates with every other bridge in the network
- Free from data loops – One root port is chosen in a bridge and the remaining ports with paths to the root bridge are put into blocking mode

The root bridge is the bridge with the lowest bridge ID in the network and functions as the logical center of the STP network. Each bridge calculates its best path to the root using the information contained in BPDUs received from its neighbor bridges. Non-root bridges select the root port among all the ports receiving BPDUs. BPDUs advertise a bridge's cost to the root bridge. The root port is chosen from the ports with received BPDUs indicating a path to the root. The root port will have the lowest cost path to the root. In the case of multiple ports offering identical costs, tie breaking is based upon the transmitting bridge ID, transmitting port ID, and receive port ID. For MSTP there are additional fields to consider – internal path cost and regional root ID. These are all discussed in more detail below.

Once the root port has been established, STP determines the other port roles. Ports providing a path to root but are not the root port become alternate ports because they provide an alternate path to the root. Other operational ports that provide a path to the root for attached bridges have the designated role. There is another type of port known as a backup port. A backup port attaches to a LAN where another port of the same bridge is a designated port. A backup port does not become part of the active topology unless the LAN's designated port is disabled and the backup port takes over the designated role.

The alternate and backup ports are set to blocking state while the root and designated ports move to the forwarding state.

Bridge priority, port path cost, and port priority are configurable parameters that are part of the port role calculation and may be modified to create the desired topology.

Bridge Priority – A typical network configuration would place two or more bridges in the core. To preserve root in the core, the core bridges would each have their bridge priority set to a lower value than bridges you do not desire to be root. The default bridge priority value is 32768. If you

desire a particular bridge to be root, set its bridge priority to a lower value than bridges that should not be root. Otherwise the bridge with the lowest MAC address is set to root.

Port Path Cost – If it is desired for a bridge to use one link over another, the administrative port path cost may be modified. The default of zero ensures that the link with the highest speed gets chosen.

Port Priority – Port Priority may be set but is not typically modified, as Link Aggregation is usually run on multiple links between two bridges.

Rapid Spanning Tree

Rapid Spanning Tree (RSTP) optimizes convergence in a properly configured network by significantly reducing the time to reconfigure the network's active topology when physical topology or configuration parameter changes occur. RSTP is defined in the IEEE 802.1w standard. Spanning Tree's primary goal is to ensure a fully connected, loop-free topology. A secondary goal, realized with the introduction of RSTP, is to move root and designated ports to the forwarding state as quickly as possible.

In a stable topology all the root and designated ports will be forwarding and the alternate and backup ports will be blocking. When there is a network topology change, Spanning Tree recalculates port roles. Ports which are no longer part of the active topology will be put into blocking state. New designated ports will only forward after receiving an acknowledgement or, in the case of a port being connected to a non-RSTP device (802.1d), after a sufficient amount of time has passed.

When a topology change occurs, a change in port operational status or new information contained in BPDUs is immediately acted upon. A new root port moves to forwarding state as soon as any recent former root port is put into blocking state. A designated port moves to forwarding state once the connected device acknowledges agreement with the new topology information. This is typically an exchange of two BPDUs. These rules ensure an orderly transition from the old topology to the new topology by preventing transient loops.

Multiple Spanning Tree

The Multiple Spanning Tree Protocol (MSTP) provides for traffic forwarding on multiple ports for each bridge. A single Spanning Tree only allows for single root port forwarding per bridge. MSTP provides for a number of common network requirements that cannot be configured on a single Spanning Tree (for example, the segregation of traffic over multiple VLANs or optimizing the utilization of redundant links between switching devices in a network).

An MSTP configuration is made up of one or more:

- Multiple Spanning Tree (MST) Regions – A set of connected bridges that share the same MST configuration ID
- MST configuration IDs – A unique identifier for each MST region
- Spanning Tree Identifiers (SIDs) – A unique identifier for each Spanning Tree

An MSTP configuration is made up of zero or more Multiple Spanning Tree Instances (MSTIs). An MSTI is an SID that exists within an MST region other than the default SID 0.

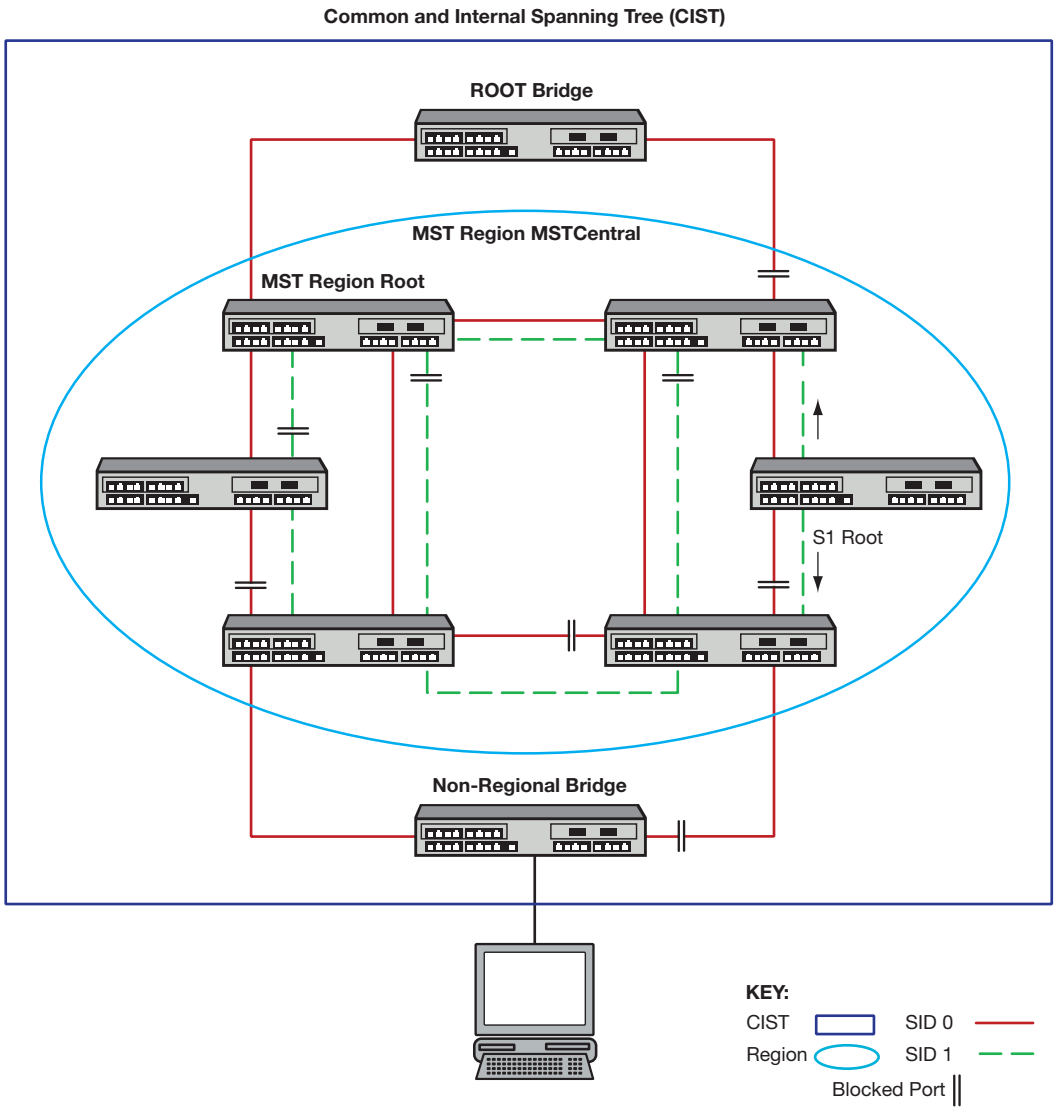
All bridges in the Spanning Tree network are inter-connected by SID 0 and can belong to:

- The Common Spanning Tree (CST) – A Spanning Tree defined in the IEEE 802.1q standard that assumes one Spanning Tree instance for the entire bridged network, regardless of the number of VLANs

- An Internal Spanning Tree (IST) instance – A Spanning Tree instance that extends the CST inside the MST region and represents the entire MST region as a single CST virtual bridge to the outside world.

One or more MSTs can be part of the Common and Internal Spanning Tree (CIST). The CIST represents the connectivity of the entire network. Figure 21-3 provides an overview of an MST configuration with one MST region within the CIST. The MST region's configuration ID name is **MSTCentral**.

Figure 21-3 Multiple Spanning Tree Overview



SID 0 is the default Spanning Tree and interconnects all bridges to the Root Bridge. SID 0 within the MST is the Internal Spanning Tree (IST) and provides connectivity out to the CST as well as functioning as another Spanning Tree instance within the MST region. SID 1 is an MSTI configured within the MST region.

Each SID has a root bridge. In Figure 21-3 the SID 0 root bridge belongs to the CST. The SID 0 root bridge functions as root for SID 0 Spanning Tree instance in both the CST and MST. SID 1 only exists within the **MSTCentral** region. The root for SID 1 is a bridge within the MSTCentral region. SID 1 can provide traffic segmentation by forwarding traffic on a second VLAN within the

MSTCentral region or provide for optimization of redundant links by forwarding traffic within the MSTCentral region on the same VLAN.

See Configuring MSTP on page 21-27 for examples of MSTP traffic segregation and optimization of redundant links.

Note: MSTP and RSTP are fully compatible and interoperable with each other and with legacy STP.

Functions and Features Supported on the S-Series Device

Spanning Tree Versions

MSTP and RSTP automatically detect the version of Spanning Tree being used on a LAN. RSTP bridges receiving MSTP BPDUs interpret them as RSTP BPDUs. MSTP and RSTP bridges receiving STP BPDUs will switch to use STP BPDUs when sending on the port connected to the STP bridge. MSTP incorporates a force version feature that allows you to administratively force MSTP to behave as STP or RSTP. This will cause all ports of the bridge to transmit STP or RSTP BPDUs. Use the force version feature when the MSTP bridge is attached to a device that cannot properly handle a non-STP BPDU.

Note: Forcing a bridge to STP will prevent it from joining a region and will disable rapid reconfiguration.

Maximum SID Capacities

By default, Multiple Spanning Tree mode is globally enabled on Extreme Networks switching devices and a single Spanning Tree is configured as SID 0.

Maximum device SID capacities in addition to SID 0 are 64 instances.

Network Diameter

Extreme Networks switching devices support a default 20-bridge span from and including the root bridge. You can configure support for a maximum span of up to 40 bridges from the Spanning Tree root in the Common Spanning Tree (CST) or the Common and Internal Spanning Tree (CIST) regional root within an MST region. Max age defines the diameter for the CST and Maxhops defines the diameter within a region. See Defining the Maximum Age Time on page 21-25.

Port Forwarding

MSTP and RSTP use rapid forwarding mechanisms to get ports to the forwarding state. However, there is a difference in forwarding time between user ports and inter-switch links (ISLs). If a user port is defined as adminedge TRUE using the **set spantree adminedge** command, it will forward as soon as the port becomes operational. An ISL will forward based on an exchange of BPDUs. By default, autoedge is set to TRUE and adminedge is set to FALSE. These settings satisfy most requirements. Autoedge allows a port defined as adminedge FALSE to discover in a short period of time that it is an edge port. The only time it is necessary to set adminedge to TRUE is when the attached user device cannot tolerate the several seconds required for auto-detection to detect the

port as a user port and move it to forwarding. Setting an ISL to adminedge TRUE should be avoided because it can lead to transient data loops.

Disabling Spanning Tree

Spanning Tree may be disabled globally or on a per port basis. If Spanning Tree is disabled globally all linked ports will be in a forwarding state and the Spanning Tree Protocol will not run. Additionally, a received BPDU will be treated as any multicast packet and flooded out all ports.

If Spanning Tree is disabled on a port by setting portadmin to disabled using the **set spantree portadmin** command, the port will be in a forwarding state and the protocol will not run for that port. A received BPDU will be consumed. The intention is that the port terminates the Spanning Tree domain. For instance, the port may be attached to a router. If this port were accidentally attached to another switching port, a data loop may result.

STP Features

Extreme Networks switching devices provide seamless Spanning Tree functionality by:

- Creating a single Spanning Tree from any arrangement of switching or bridging elements.
- Compensating automatically for the failure, removal, or addition of any switching device in an active data path.
- Achieving port changes in short time intervals, which establishes a stable active topology quickly with minimal network disturbance.
- Using a minimum amount of communications bandwidth to accomplish the operation of the Spanning Tree Protocol.
- Reconfiguring the active topology in a manner that is transparent to stations transmitting and receiving data packets.
- Managing the topology in a consistent and reproducible manner through the use of Spanning Tree Protocol parameters.
- Increasing security and reliability with SpanGuard, as described below and in Understanding and Configuring SpanGuard on page 21-32.
- Further protecting your network from loop formation with Loop Protect, as described below and in Understanding and Configuring Loop Protect on page 21-34.
- Supporting more port density and faster port speeds as described in Updated 802.1t on page 21-8.
- Supporting the Restricted Topology Change Notice (TCN) feature as described in Restricted Topology Change Notification (TCN) on page 21-8.
- Supporting the Restricted Role feature as described in Restricted Role on page 21-9

SpanGuard

The Extreme Networks SpanGuard feature helps protect your network from two situations that can cause a Denial of Service (DoS) condition: repeated topology change notifications and an unwanted bridge being inserted into and forcing traffic through the topology. SpanGuard increases security and reliability by preventing Spanning Tree respans that can occur when BPDUs are received on user ports and notifies network management that they were attempted.

If a SpanGuard enabled port receives a BPDU, it becomes locked and transitions to the blocking state. It will only transition out of the blocking state after a globally specified time or when it is manually unlocked. By default, SpanGuard is globally disabled on the S-Series device and must be

globally enabled to operate on all user ports. For a more detailed discussion of the SpanGuard feature, refer to Understanding and Configuring SpanGuard on page 21-32.

Loop Protect

The Loop Protect feature prevents or short circuits loop formation caused by redundant paths in your network by requiring ports to receive BPDUs (RSTP/MSTP only) on point-to-point ISLs before their states are allowed to become forwarding. Further, if a BPDU timeout occurs on a port, its state becomes listening until a new BPDU is received. In this way, both upstream and downstream facing ports are protected.

When a root or alternate port loses its path to the root bridge, due to message age expiration, it takes on the role of designated port and will not forward traffic until a BPDU is received. When a port is intended to be the designated port in an ISL, it constantly proposes and will not forward until a BPDU is received. It will revert to listening if it stops getting a response. Loop Protect also overrides the port admin setting. This protects against misconfiguration (such as disabling STP on a port using the **set spantree portadmin** *port-string* **disable** command) and protocol failure by the connected bridge. By default, the Loop Protect feature is globally disabled on Extreme Networks switch devices and must be globally enabled to operate on all ports. For configuration information, refer to Understanding and Configuring Loop Protect on page 21-34.

Updated 802.1t

IEEE 802.1t is enabled by default on Extreme Networks switch devices. This updated Spanning Tree protocol supports multiple Spanning Trees, more switch port density, and faster port speeds.

802.1t includes the following updates:

- New bridge identifier encoding (4-bit priority, 12-bit system ID extension, 48-bit bridge address)
- New port identifier encoding (4-bit priority, 12-bit port number)
- Bridge detection state machine (for edge port identification)
- Path cost default values (the ability to switch between 802.1t and 802.1d mode and cost values)

Restricted Topology Change Notification (TCN)

Restricted Topology Change Notification (TCN) is a Spanning Tree protocol feature that allows or disallows TCN propagation on specified ports. When Restricted TCN is disabled, TCN propagation is allowed. The port propagates received TCNs and topology changes to other ports. Restricted TCN is disabled by default. When Restricted TCN is enabled, the port does not propagate received TCNs and topology changes to other ports. Enable Restricted TCN to prevent unnecessary address flushing in the core region of the network caused by activation of bridges external to the core network.

A possible reason for not allowing TCN propagation is when bridges are not under the full control of the administrator or because MAC operational state for the attached or downstream LANs transitions frequently, causing disruption throughout the network.

Rapid Spanning Tree responds to TCNs by selectively flushing the filter database. Persistent TCNs are disruptive, causing persistent address flushing, which in turn causes increased flooding in the network. Restricted TCN is a useful tool when it is not possible to remove the source of the TCNs. Be cautioned that when enabled, temporary loss of connectivity can occur after changes in a Spanning Tree's active topology, due to persistent, incorrectly learned, station location information. This would be the case where the part of the topology that is the source of the unwanted TCNs is redundantly connected to other parts of the network.

Restricted Role

Restricted Role is a Spanning Tree protocol feature that allows or disallows the root role on specified ports. When Restricted Role is enabled, the port will not be selected as the root port for the CIST or any MSTI, even if it has the best Spanning Tree priority. A port with Restricted Role enabled is selected as an alternate port after the root port has been selected.

If enabled, Restricted Role can cause lack of Spanning Tree connectivity. Setting Restricted Role to enabled prevents bridges, external to a core region of the network, from influencing the Spanning Tree active topology. You may wish to use Restricted Role when bridges are not under your full control. You may also wish to enable Restricted Role on ports where the bridge is external to the core and where the port faces away from the root, in cases where the port role would normally be designated. This can speed network reconvergence, particularly after loss of the root bridge. Restricted role is disabled by default.

Multisource Detection

Multisource detection is a feature that prevents network disruption due to excessive topology changes caused by a full duplex port transmitting multiple BPDUs with different source MAC addresses, and hence different BPDU information.

When a port is point-to-point, the received priority information comes from the most recently received BPDU. When a port is non-point-to-point, the received information reflects the best priority information out of all the received BPDUs. Typical scenarios for multisource detection are when a switch is connected to a device which

- has been improperly configured to forward received BPDUs out other ports, or
- has been configured to not run the Spanning Tree protocol and treats BPDUs as multicast packets by transmitting them out all other forwarding ports.

In these situations, the connected port is effectively acting as a shared media device. The way to detect shared media is the duplex setting. Since the port is full duplex, it treats the connection as point-to-point. Multisource detection, which is always enabled, recognizes the multiple source MAC addresses and sets the port's operational point-to-point status to false, treating the port as a shared media device. The port is constantly monitored. If the situation is resolved, as determined by receiving a unique address for a sufficient amount of time, the port's operational point-to-point status will revert to true.

A syslog message is issued when multiple source addresses are detected.

Note: When loop protect is configured for the port, if multisource detection is triggered, the port will go to the listening state and no longer be part of the active topology. Loop protect does not operate on shared media ports.

Understanding How Spanning Tree Operates

This section provides you with a more detailed understanding of how the Spanning Tree operates in a typical network environment.

Spanning Tree Basics

The most elemental task of a Spanning Tree Bridge is to control the forwarding state of each port. The bridge evaluates the information received from its immediate neighbors in the form of BPDUs, along with its own configured information. From this information a root is elected and then port roles may be selected for each port. For the root port and designated ports the desired state is forwarding. These ports will become forwarding by subsequent exchange of BPDUs or through the expiration of protocol timers according to the state machines defined by the Spanning Tree Protocol. The remaining ports will become discarding (shorthand for the states of blocking, listening, and learning).

To facilitate this process, the bridge transmits BPDUs out each port on a periodic basis as well as in response to events such as changes in port operational status, configuration changes, timer expiration, and changes in topology derived from received BPDUs.

Electing the Root Bridge

The network topology is determined by the selection of the root bridge. The topology is based on each bridge's best path to root. Root election occurs on each bridge when new information is received from a neighboring bridge in a BPDU, when link is lost on a port connecting a neighboring bridge, or when the bridge's priority is administratively changed.

The root is elected by comparing the root IDs received in BPDUs as well as the bridge's own bridge ID. The bridge with the lowest ID is chosen as root. The bridge ID is an 8-byte value with the 2 most significant bytes being the bridge priority and the 6 least significant bytes being the bridge MAC address. Root may be forced to a particular bridge by the configuration of bridge priority. Among bridges with the same bridge priority, the one with the lowest MAC address is elected root. If a bridge receives no BPDUs indicating a better bridge ID than its own, it becomes the root bridge.

Assigning Path Costs

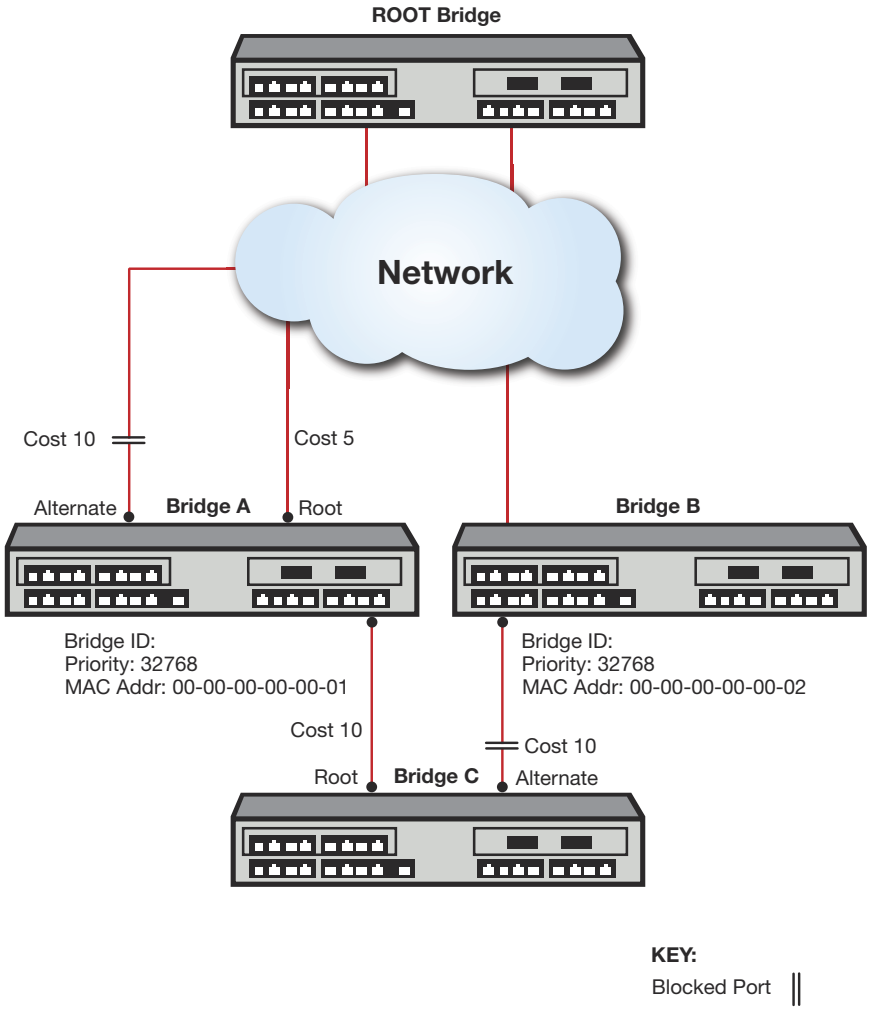
Path costs are one factor in determining port roles. Each LAN segment has an operational path cost associated with it. The cost is based on the port speed, by default. The higher the speed, the lower the cost. Port costs for link aggregations are based on the aggregate speed of all the underlying physical ports. The port cost value may also be administratively assigned using the **set spantree adminpathcost** command. This may be done to choose a particular path.

Paths to Root

If the bridge is not elected as root, one or more ports provide a path back to the root bridge. The port with the best path is selected as the root port. The best path is the one that has the lowest designated cost. The lowest cost is the aggregate cost of all the LANs traversed between the port and the root bridge. Figure 21-4 on page 21-11 displays root port configuration based upon lowest

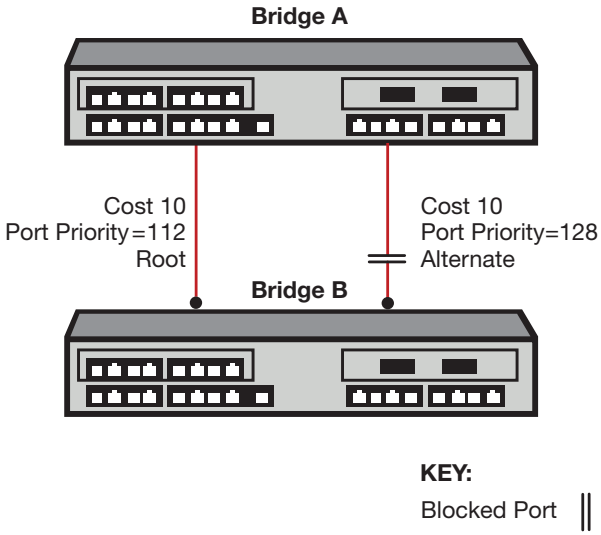
cost for Bridge A. If multiple ports have the same lowest cost, the one with the lowest bridge ID becomes the root port. The bridge ID is the ID of the transmitting bridge. Figure 21-4 displays root port configuration based upon lowest bridge ID for Bridge C.

Figure 21-4 Root Port Selection Based Upon Lowest Cost or Bridge ID



If there are ports with the same bridge ID, the port ID is used as a tie breaker. The port with the lowest port ID is chosen as root port. The port ID is a 2-byte value with the 4 most significant bits being the port priority and the 12 least significant bits being the bridge port number. Because the port priority occupies the most significant bits in the port ID, setting a lower port priority assures that port will be selected as root. In the case of no single port having a lowest port priority, the root port is selected based upon the overall port ID value. Figure 21-5 on page 12 presents a root port configuration for Bridge B determined by the port priority setting. If there is still a tie, these ports are connected via a shared medium. The final tie breaker is the receiving port ID.

Figure 21-5 Root Port Selection Base Upon Lowest Port ID

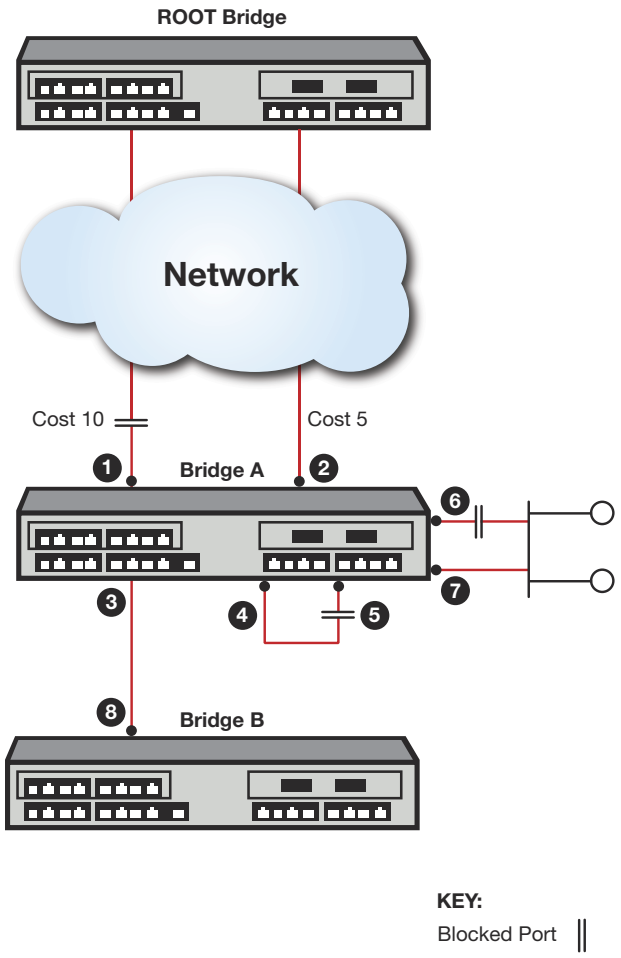


After selecting the root port, the bridge's cost to root is the total of the root port's designated cost as advertised in the received BPDU, plus the path cost associated with that port. In a hierarchically designed network, the designated cost for ports attached to the next higher level will be less than the bridge's cost to the root. Ports attached to bridges on the same level will have designated costs equal to the bridge's cost to the root. Likewise, this bridge's port will advertise the bridge's cost to the root. Thus one port connected to the LAN will be the designated port and the other(s) will be alternate. The port on the bridge with the lower ID will be the designated port.

Identifying Designated, Alternate, and Backup Port Roles

Ports in a Spanning Tree configuration are assigned one of four roles: root, designated, alternate, or backup. Figure 21-6 on page 21-13 presents an overview of Spanning Tree port roles.

Figure 21-6 Spanning Tree Port Role Overview



1	Port 1, Bridge A, Alternate Port	**5**	Port 5, Bridge A, Backup Port
2	Port 2, Bridge A, Root Port	**6**	Port 6, Bridge A, Backup Port
3	Port 3, Bridge A, Designated Port	**7**	Port 7, Bridge A, Designated Port
4	Port 4, Bridge A, Designated Port	**8**	Port 1, Bridge B, Root Port

All ports which act as edge ports take on the designated port role. If the bridge has been elected root, all ports connected to ports on other bridges are also designated ports.

On non-root bridges, Spanning Tree identifies ports which provide a path to the root bridge and selects the best path among these as the root port as described in Paths to Root on page 21-10 (Figure 21-6, callout2). There may be only a single port providing a path to root, in which case that is the root port and the remaining ports are designated. If there are other ports providing a path to root, these ports are selected as alternate paths. Should the root port become disabled, one of the alternate ports will be selected as the new root port (Figure 21-6, callout 1).

A port which is not a designated port, but is connected to another port on the same bridge (Figure 21-6, callout 5) or connected to a shared LAN on which this bridge already provides a designated port (Figure 21-6, callout 6), takes the role of backup port. In the shared LAN example it may take over as designated port if the original designated port is disabled.

All operational ports which are not root, alternate or backup are designated ports. These ports provide a path to the root for attached devices.

Table 21-1 on page 21-14 provides a summary of STP port roles.

Table 21-1 Spanning Tree Port Roles

Port Role	Description
Root	The one port that is used to connect to the root bridge. It is elected based on its least “path-cost” to the root bridge and is forwarding traffic.
Alternate	Any redundant upstream port that provides an alternate path to the root bridge (other than the root port). Alternate ports are set to blocking.
Designated	Any downstream port that provides a path back to the root bridge for a downstream bridge. This port is forwarding traffic.
Backup	A port that acts as a redundant designated port on a shared LAN. Backup ports are set to blocking.

Assigning Port States

All ports are blocking when the operational status switches from disabled to enabled. By default, automatic edge detection is enabled and ports are configured as non-edge ports. In this scenario a user port will become forwarding in several seconds. A port configured as an edge port will forward immediately.

Ports which are selected as alternate or backup ports are immediately put into the discarding state and remain discarding until a new port role is selected. The root port may go to the forwarding state as long as any recent former root ports are synchronized with the new root information. Designated ports may forward as soon as the attached port signals agreement as specified by RSTP. In the absence of the above conditions, root and designated ports get to the forwarding state through the use of timers. The value of the timers is dependent on the value of ForceVersion. The default value is MSTP. If the value is StpCompatible, the timer values are derived from forward delay. Otherwise the values are derived from hello time.

Table 21-2 provides a summary of STP port states.

Table 21-2 Spanning Tree Port States

Port State	Behavior
Blocking	Actively preventing traffic from using this path. Still receiving BPDUs, so continuing to monitor for management and STA information.
Listening	Continuing to block traffic while waiting for protocol information to determine whether to go back to the blocking state or continue to the learning state. Listens to BPDUs to ensure no loops occur on the network.
Learning	Learning station location information but continuing to block traffic.
Forwarding	Forwarding traffic and continuing to learn station location information.
Disabled	Disabled administratively or by failure.
Discarding	Used as shorthand for blocking, listening, or learning state.

RSTP Operation

RSTP optimizes convergence by significantly reducing the time to reconfigure the network’s active topology when physical topology or configuration parameter changes occur. RSTP provides rapid connectivity following the failure of a switching device, switch port, or the addition of a switch into the network.

A new root port may forward as soon as any recent root ports are put into blocking.

A designated port may forward with the exchange of two BPDUs in rapid succession. The designated port presents new BPDU information with a proposal request. The attached port processes the BPDU and may respond immediately with an agreement. Upon reception of that agreement BPDU, the designated port may move to forwarding. Another feature of RSTP is that designated ports transmit periodic BPDUs regardless of reception of BPDUs at the root port. This insulates the network from jitter in receiving BPDUs, particularly at the edge.

Important STP timers are max age, hello time, and forward delay. The default values for the timers are:

- Hello time – 2 seconds
- Forward delay – 15 seconds
- Max age – 20 seconds

The operational values from these timers are derived from the root bridge. The current IEEE standard for Spanning Tree fixes hello time at 2 seconds. The Enterasys switches covered in this document do not enforce this restriction to allow existing configurations to remain compatible. It is not recommended that a value other than 2 seconds be used. Other values may not interact well with other non-variable protocol times such as edgeDelayWhile or mDelayWhile. The max age timer may be adjusted to change the network diameter. Take care to consider that failure in the network may cause the topology to "unravel" causing the diameter to become larger than anticipated. An insufficient value could cause devices near or at the edge of the network to become unreachable. For example, in a ring topology of 10 bridges, no bridge is more than 5 hops from the root. A max age that accounts for 6 hops would be sufficient. A failure of ports immediately interconnecting a bridge with the root would break the ring topology and change the furthest hop from the root from 5 to 9. Any bridges beyond the configured network diameter of 6 would cause the Spanning Tree topology not to converge.

MSTP Operation

MSTP makes it possible for VLAN switching devices to use multiple Spanning Trees, allowing traffic belonging to different VLANs to flow over potentially different paths within the LAN. It builds upon the advancements of RSTP with its decreased time for network re-spans. MSTP's principle objective is to increase bandwidth utilization by allowing:

- Frames assigned to different VLANs to follow different data routes
- Ports to block for some Spanning Trees and forward for others
- Every inter-switch link in the topology to be forwarding for at least one Spanning Tree

MSTP is the default Spanning Tree mode on all Extreme Networks switch devices.

Common and Internal Spanning Tree (CIST)

MSTP uses all Spanning Tree region information to create a single Common and Internal Spanning Tree (CIST) that represents the connectivity of the entire network. This is equivalent to the single Spanning Tree used for STP and RSTP.

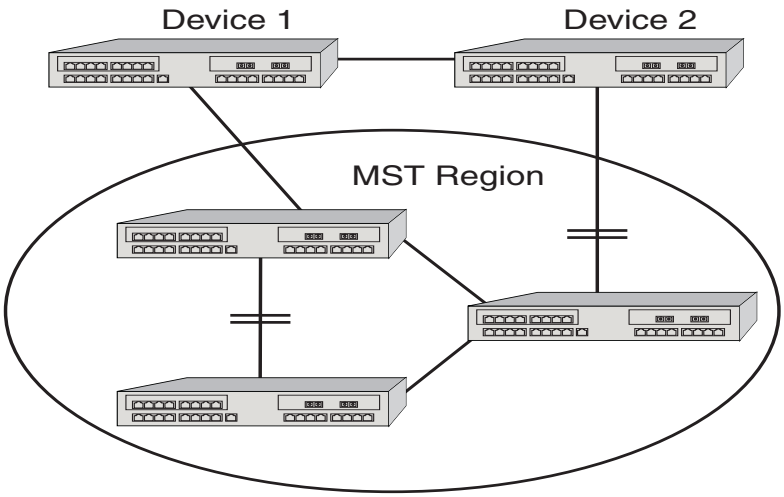
The MSTP enabled network may contain any combination of Single Spanning Tree (SST) regions and Multiple Spanning Tree (MST) regions. A typical network may contain multiple MST regions as well as separate LAN segments running legacy STP and RSTP Spanning Tree protocols. The CIST contains a root bridge, which is the root of the Spanning Tree for the network. The CIST root may be, but is not necessarily, located inside an MST region. Each MST region contains a CIST regional root which may be the CIST root if the CIST root is internal to the region. If the CIST root is external to the region, the CIST regional root provides the connectivity to the CIST root. Bridges in an MSTP topology compare their received BPDUs to calculate their shortest path to the CIST root, CIST regional root, and MSTI regional root.

Ideally, there should be one all-encompassing region. This is not always possible, for example, when non-MSTP bridges exist such as those shown in Figure 21-3 on page 5. From the outside, the region appears as a single Spanning Tree bridge which is part of the Common Spanning Tree (CST). A port which connects to a bridge not having the same MST configuration ID, or which is not running MSTP, forms part of the boundary of the region. The region attaches to the CST at the root port of the CIST regional root. All other region boundary ports which provide paths to the root are alternate ports and remain blocking until the topology changes, causing a new regional root port to be chosen. Ports which provide a path to the root for other bridges at the region boundary are designated ports. At boundary ports, port states for MSTIs follow the states of the CIST for the port.

MST Region

An MST region is a group of devices that are configured together to form a logical region. The MST region presents itself to the rest of the network as a single switching device, which simplifies administration. Path cost is only incremented when traffic enters or leaves the region, regardless of the number of devices within the region. Each LAN can only be a member of one region. Figure 21-7 shows that the MST region appears as a single switching device to devices 1 and 2, but really consists of three devices.

Figure 21-7 Example of an MST Region



For a switching device to be considered as part of an MST region, it must be administratively configured with the same configuration identifier information as all other devices in the MST region. The configuration identifier consists of four parts:

- Format Selector – One octet in length and is always 0. It cannot be administratively changed.
- Configuration Name – A user-assigned, case sensitive name given to the region. The maximum length of the name is 32 octets. A bridge's default configuration name is a character string corresponding to the bridge MAC address. This guarantees that the default behavior of a bridge is to not be part of an MST region.
- Revision Level – Two octets in length. The default value of 0 may be administratively changed.
- Configuration Digest – 16-octet HMAC-MD5 signature created from the configured VLAN Identification (VID)/Filtering Identification (FID) to Multiple Spanning Tree Instances (MSTI) mappings. All devices must have identical mappings to have identical configuration digests.

By default, each bridge is in its own MST region and has a default configuration name derived from the bridge MAC address. For example, if the bridge MAC address is **00-1f-45-9a-6c-b7**, the

default MSTP configuration name is "**00:1f:45:9a:6c:b7**". When grouping two or more bridges into a single MST region, you must assign the same configuration name to each member of the region. MD5 digests are derived from a mapping of a Filtering Database ID (FID) to a Spanning Tree ID (SID), referred to as a FID-to-SID mapping (see Multiple Spanning Tree Instances (MSTI) on page 21-17 for more information). Since there is a small probability of different mappings resulting in the same digest, the addition of administratively assigned name and version configuration ID parameters guarantee the uniqueness of the region.

SIDs exist within an MST region, each having a separate topology. Within an MST region there always exists the Internal Spanning Tree (IST) which is SID 0. There are zero or more Multiple Spanning Tree Instances (MSTIs). Each MSTI corresponds to a set of VIDs. One or more VIDs may be mapped to an SID using a FID-to-SID mapping. The IST and each MSTI may have different root bridges. Port path costs and bridge priorities may be different for each port/instance. Each bridge port has a unique port state per instance. With proper configuration, redundant links may be utilized to their maximum extent by each forwarding for one or more instances. See Configuring MSTP on page 21-27 for more detail on how to do this.

Multiple Spanning Tree Instances (MSTI)

Inside the MST region, a wholly contained set of topologies is maintained separate from the outside world. For example, MSTI 1 in MST region A has no correspondence to MSTI 1 in MST region B. The S-Series supports 64 MST instances.

The Extreme Networks switch device by default maps VLAN IDs (VIDs) to Filtering IDs (FIDs) in a one-to-one correlation for bridges with the VLAN learning mode set to individual VLAN learning (IVL). VIDs to FIDs can also be mapped in a many-to-one correlation for bridges with the VLAN learning mode set to shared VLAN learning (SVL). The VLAN learning mode and shared VLAN learning VID to FID mapping are set by configuring VLAN constraint using the **set vlan constraint** command.

For example, in an IVL bridge, FID 3 may contain VID 3 and FID 4 may contain VID 4. In an SVL bridge, FID 3 may contain VID 3 and FID 4 may contain VIDs 4 and 5. Regardless of the type of VLAN learning taking place, one or more FIDs may be mapped to a Spanning Tree Instance (SID). The end result is a mapping of VIDs to SIDs. SID topologies may then be configured to provide a type of load balancing. Note that without further configuration, each SID will have the same topology as the IST. Typically, load balancing will be achieved by choosing different root bridges in the core for the different instances.

See "Learning Modes and Filtering Databases" on page 24-3 for a learning mode and filtering database discussion.

Determining FID-to-SID Mappings

VLANs are mapped to MSTIs through a FID-to-SID mapping which is the key element in an MSTP configuration. Each VLAN is associated to a FID and is mapped to Spanning Tree IDs using their FID association. The mapping is performed by the **set spantree mstmap** command. This mapping is represented within the MST configuration digest described in the previous section and displayed in the following example. By default, every bridge will have a FID-to-SID mapping that equals VLAN FID 1/SID 0.

Use the **show spantree mstcfgid** command to determine MSTI configuration identifier information, and whether or not there is a misconfiguration due to non-matching configuration identifier components:

This example shows how to display MSTI configuration identifier information. In this case, this bridge belongs to "Region1":

```
Extremenetworks->show spantree mstcfgid
MST Configuration Identifier:
```

```
 Format Selector:      0
 Configuration Name:   Region1
 Revision Level:       88
 Configuration Digest: 6d:d7:93:10:91:c9:69:ff:48:f2:ef:bf:cd:8b:cc:de
```

In order for other bridges to belong to Region1, all four elements of those bridges' configuration id output must match. The default value that must be changed for this to happen is the configuration name setting. Also, the MSTIs must be created and the FIDs mapped to them.

Use the **set spantree mstcfgid** command to change the configuration name from the default bridge MAC address value.

This example changes the default bridge configuration name to **Region1**:

```
Extremenetworks->set spantree mstcfgid cfgname Region1
```

For the configuration digest to match, the mapping of VIDs to SIDs must match. Use these commands to configure the SIDs, map the FIDs to the SIDs and display the VID-SID and FID-SID mappings:

```
Extremenetworks->set spantree msti sid 3 create
Extremenetworks->set spantree msti sid 4 create
Extremenetworks->set spantree mstmap 3 sid 3
Extremenetworks->set spantree mstmap 4 sid 4
Extremenetworks->show spantree mstilist
Configured Multiple Spanning Tree Instances:
 3     4
Extremenetworks->show spantree mstmap
Fid 3 is mapped to Sid 3
Fid 4 is mapped to Sid 4
Extremenetworks->show spantree vlanlist
Vlan 3 is mapped to Sid 3
Vlan 4 is mapped to Sid 4
```

Since an MSTI is a separate Spanning Tree, each MSTI has its own root inside the MST region. Figure 21-8 and Figure 21-9 show two MSTIs in a single region. Switching device 3 is the root for MSTI 1, switching device 2 is the root for MSTI 2, and switching device 5 is the CIST regional root. Traffic for all the VLANs attached to an MSTI follow the MSTI's spanned topology.

Various options may be configured on a per-MSTI basis to allow for differing topologies between MSTIs. To reduce network complexity and processing overhead needed to maintain MSTIs, you should only create as many MSTIs as needed.

Figure 21-8 MSTI 1 in a Region

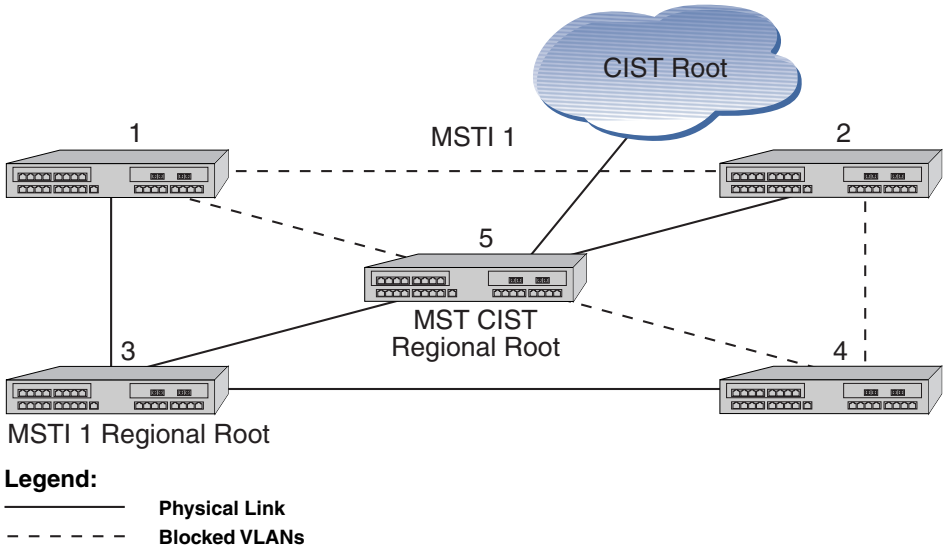


Figure 21-9 MSTI 2 in the Same Region

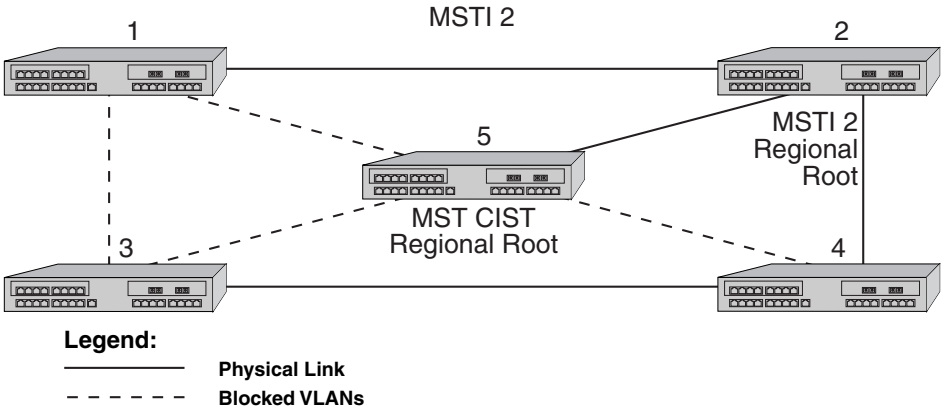


Figure 21-10 shows 3 regions with five MSTIs. Table 21-3 defines the characteristics of each MSTI. Ports connected to PCs from devices 1, 3, 9, and 11 will be automatically detected as edge ports. Devices 4 and 10 are the CIST regional roots. Each MSTI can be configured to forward and block various VLANs.

Figure 21-10 Example of Multiple Regions and MSTIs

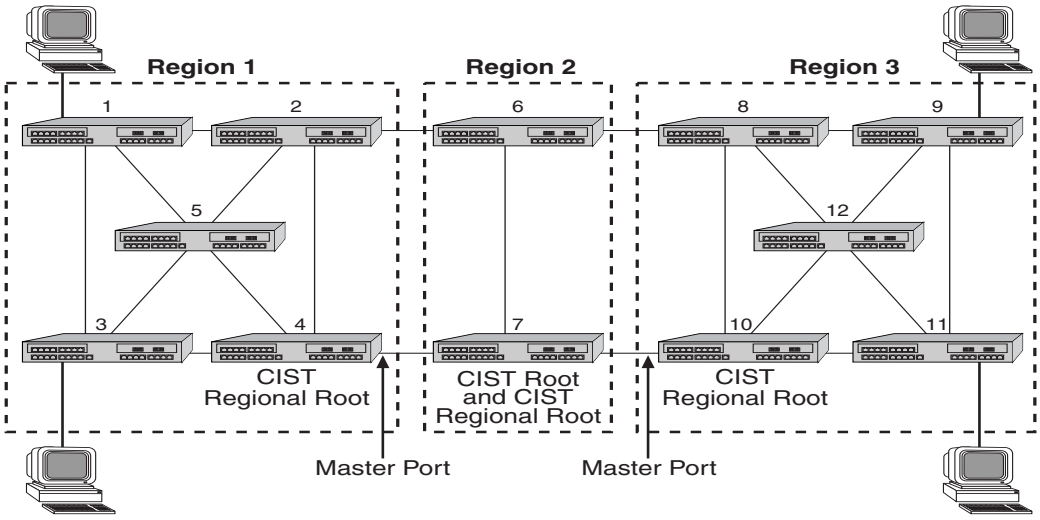


Table 21-3 MSTI Characteristics for Figure 21-10

MSTI / Region	Characteristics
MSTI 1 in Region 1	Root is switching device 4, which is also the CIST regional root
MSTI 2 in Region 1	Root is switching device 5
MSTI 1 in Region 2	Root is switching device 7, which is also the CIST root
MSTI 1 in Region 3	Root is switching device 11
MSTI 2 in Region 3	Root is switching device 12
	Switching device 10 is the CIST regional root

Multisource Detection

Multisource Detection is a feature that prevents network disruption due to excessive topology changes caused by a full duplex port transmitting multiple BPDUs with different source MAC addresses, and hence different BPDU information. When a port is point-to-point, the received priority information comes from the most recently received BPDU. When a port is non-point-to-point, the received information reflects the best priority information out of all the received BPDUs.

Typical scenarios for multisource detection are when a switch is connected to a device which:

- Has been improperly configured to forward received BPDUs out other ports
- Or has been configured to not run the Spanning Tree protocol and treats BPDUs as multicast packets by transmitting them out all other forwarding ports.

In these situations, the connected port is effectively acting as a shared media device. Shared media is detected using the duplex setting. Since the port is full duplex it treats the connection as point-to-point.

One way of preventing the disruption of this situation is to configure the receiving port's adminpoint value to false. This causes the operpoint value to always be false and to be treated as non-point-to-point. Multisource Detection, which is always enabled, recognizes the multiple source MAC addresses and automatically sets the operpoint value to false when the adminpoint value is auto. The port is constantly monitored. If the situation is resolved, as determined by

receiving a unique address for a sufficient amount of time, the operpoint value will be restored to true.

A syslog message is issued when multiple source addresses are detected:

```
Receive Event: Multiple BPDU sources received on Port = ge.1.1 BPDU Source MAC =
00:00:00:01:02:03 Prior Source MAC = 00:00:00:0f:0e:0d
```

Note: When loop protect is configured for the port, if multisource detection is triggered, the port will go to the listening state and no longer be part of the active topology. Loop protect does not operate on shared media ports.

Configuring STP and RSTP

Caution: Spanning Tree configuration should be performed only by personnel who are very knowledgeable about Spanning Trees and the configuration of the Spanning Tree Algorithms. Otherwise, the proper operation of the network could be at risk.

Reviewing and Enabling Spanning Tree

By default, Spanning Tree is enabled globally on Extreme Networks switch devices and enabled on all ports. On all switching devices, the default Spanning Tree version is set to MSTP (802.1s) mode. Since MSTP mode is fully compatible and interoperable with legacy STP and RSTP bridges, in most networks, this default should not be changed.

Use the following commands to review, re-enable, and reset the Spanning Tree mode.

1. Review the current configuration on one or more SIDs, ports, or both:

   show spantree stats [**port** *port-string*] [**sid** *sid*] [**active**]

   Specifying **active** will display information for port(s) that have received BPDUs since boot.

2. If necessary, globally enable Spanning Tree:

   set spantree stpmode ieee8021

3. Review the status of Spanning Tree on one or more ports:

   show spantree portadmin [**port** *port-string*]

4. If necessary, re-enable Spanning Tree on one or more ports:

   set spantree portadmin *port-string* **enable**

Example

This example shows how to display the device's Spanning Tree configuration:

```
Extremenetworks->show spantree stats
SID                          - 1
Spanning tree mode           - enabled
```

```
Designated Root              - 00-e0-63-6c-9b-6d
Designated Root Priority     - 0
Designated Root Cost         - 1
Designated Root Port         - ge.5.1
Root Max Age                 - 20 sec
Root Hello Time              - 2  sec
Root Forward Delay           - 15 sec
Bridge ID MAC Address        - 00-e0-63-9d-b5-87
Bridge priority              - 32768
Bridge Max Age               - 20 sec
Bridge Hello Time            - 2  sec
Bridge Forward Delay         - 15 sec
Topology Change Count        - 6539
Time Since Top Change        - 00 days 00:00:00
```

Note: By default, Spanning Tree is enabled both globally and on all ports.

Adjusting Spanning Tree Parameters

You may need to adjust certain Spanning Tree parameters if the default values are not suitable for your bridge configuration. Parameters affecting the entire Spanning Tree are configured with variations of the global bridge configuration commands. Interface-specific parameters are configured with variations of the Spanning Tree port configuration commands. Default settings are listed in Table 21-4:

Table 21-4 Spanning Tree Port Default Settings

Setting	Default Value
Bridge priority mode	802.1t
Bridge priority	32768
Port priority	128
Port cost	0 (automatically calculated based on port speed)
Hello time (bridge and ports)	2 seconds
Bridge forward delay	15 seconds
Bridge maximum aging time	20 seconds

Use the commands in the following sections to adjust these defaults.

Note: Poorly chosen adjustments to these parameters can have a negative impact on network performance. Please refer to the IEEE 802.1D specification for guidance.

Setting Bridge Priority Mode and Priority

Bridge priority mode affects the range of priority values used to determine which device is selected as the Spanning Tree root. By default, switching devices are set to 802.1t mode as described in Updated 802.1t on page 21-8.

Use this command to set the bridge priority mode:

```
set spantree bridgepriortymode 802.1t | 802.1d
```

In addition to setting priority mode, you can globally configure the priority of an individual bridge. When two bridges tie for position as the root bridge, this setting affects the likelihood that a bridge will be selected. The lower the bridge's priority, the more likely the bridge will be selected as the root bridge.

Use this command to set the bridge priority:

```
set spantree priority priority [sid]
```

Valid *priority* values are:

– For 802.1t priority mode: **0–61440** (in increments of 4096), with 0 indicating high priority and 61440 low priority. Values will automatically be rounded up or down, depending on the 802.1t value to which the entered value is closest.

– For 802.1D priority mode: **0–65535** (in increments of 1), with 0 indicating high priority and 65535 low priority.

Valid *sid* values are **0–4094**. If not specified, SID 0 will be assumed.

Setting a Port Priority

You can set a Spanning Tree port priority. Port priority is used to break a tie when choosing the root port for a bridge, in a case where the choice is between ports connected to the same bridge. The port with the lowest value will be elected.

Use this command to set a port priority:

```
set spantree portpri port-string priority [sid sid]
```

Valid *priority* values are **0–240** (in increments of 16) with 0 indicating high priority.

Valid *sid* values are **0–4094**. If not specified, SID 0 will be assumed.

Assigning Port Costs

Each interface has a Spanning Tree port cost associated with it, which helps to determine the quickest path between the root bridge and a specified destination. By convention, the higher the port speed, the lower the port cost. By default, this value is set to 0, which forces the port to recalculate Spanning Tree port cost based on the speed of the port and whether or not legacy (802.1D) path cost is enabled.

Use this command to assign different Spanning Tree port costs:

```
set spantree adminpathcost port-string cost [sid sid]
```

Valid *cost* values are:

– **0–65535** if legacy path cost is enabled.

– **0–200000000** if legacy path cost is disabled.

Valid *sid* values are **0–4094**. If not specified, SID 0 will be assumed.

Notes: Please refer to the IEEE 802.1D specification for guidance in setting appropriate cost values for your port speeds.

By default, legacy path cost is disabled. Enabling the device to calculate legacy path costs affects the range of valid values that can be administratively assigned.

To check the status of legacy path cost, use **show spantree legacypathcost**.

To disable legacy path cost, if necessary use **set spantree legacypathcost disable**.

Adjusting Bridge Protocol Data Unit (BPDU) Intervals

Use the commands in this section to adjust default BPDU interval values.

Table 21-5 BPDU Interval Defaults

BPDU Interval	Default Value
Hello time (bridge and ports)	2 seconds
Forward delay	15 seconds
Maximum age time	20 seconds

Adjusting the Bridge Hello Time

Caution: Poorly chosen adjustments to bridge and port hello time parameters can have a negative impact on network performance. It is recommended that you do not change these parameters unless you are familiar with Spanning Tree configuration and have determined that adjustments are necessary. Please refer to the IEEE 802.1D specification for guidance.

Hello time is the interval, in seconds, at which the bridge or individual ports send BPDU messages. By default, bridge hello mode is enabled, meaning the device uses a single bridge administrative hello time.

Adjust the bridge hello time as follows:

1. Check the status of bridge hello mode:

   `show spantree bridgehellomode`

2. If necessary, re-enable bridge hello mode:

   `set spantree bridgehellomode enable`

3. Set a new hello time interval:

   `set spantree hello` *`interval`*

   Valid *interval* values are 1–10.

Adjusting Port Hello Times

You can set the device to use per-port administrative hello times by disabling bridge hello mode and adjusting the hello time interval for one or more ports as follows:

1. Check the status of bridge hello mode:

   `show spantree bridgehellomode`

2. If necessary, disable bridge hello mode:

   `set spantree bridgehellomode disable`

3. Set a new hello time interval for one or more ports:

   `set spantree porthello` *`port-string interval`*

   Valid *interval* values are 10–100

Adjusting the Forward Delay Interval

When rapid transitioning is not possible, forward delay is used to synchronize BPDU forwarding. The forward delay interval is the amount of time spent listening for topology change information after an interface has been activated for bridging and before forwarding actually begins. This delay is required because every device must receive information about topology changes before it

starts to forward frames. In addition, each port needs time to listen for conflicting information that would make it return to a blocking state. Otherwise, temporary data loops might result.

Use this command to adjust the forward delay interval setting:

```
set spantree fwddelay delay
```

Valid *delay* values are 4–30.

Defining the Maximum Age Time

If a bridge does not hear BPDUs from the root bridge within the interval (number of seconds) specified as maximum age time, it assumes that the network has changed and recomputes the Spanning Tree topology. By adjusting this value, you can configure support for a maximum diameter from the STP root of up to 40 bridges. By default, Extreme Networks switching devices are set with a maximum age time of 20 seconds, supporting a 20-bridge span from the root bridge.

Use this command to adjust the maximum age setting:

```
set spantree maxage agingtime
```

Valid *agingtime* values are 6–40 (seconds).

Setting the Maximum Configurable STPs

By default, Multiple Spanning Tree mode is globally enabled on Extreme Networks switching devices and one Spanning Tree is configured as Spanning Tree ID (SID) 0. As described in "Maximum SID Capacities" on page 21-6, devices support different numbers of Spanning Tree instances (including SID 0), depending on their model type and memory installed. SID values are from 1 to 4094.

The S-Series allows you to set the maximum number of user configured Spanning Trees allowed on the device:

```
set spantree maxconfigurablestps numstps
```

Valid *numstps* values are 1 – 64.

Setting Restricted TCN and Restricted Role

As described in Restricted Topology Change Notification (TCN) on page 21-8, Restricted Topology Change Notice (TCN) allows or disallows TCN propagation on specified ports.

The S-Series device allows you to restrict the propagation of TCNs on the specified port by setting Restricted TCN to true:

```
set spantree restrictedtcn port-string {true | false}
```

By default, Restricted TCN is set to false, allowing the propagation of TCNs on the specified port.

As described in Restricted Role on page 21-9, Restricted Role allows or disallows the root role on specified ports.

The S-Series device allows you to restrict the root role on the specified port by setting Restricted Role to true:

```
set spantree restrictedrole port-string {true | false}
```

By default, Restricted Role is set to false, allowing the root role on the specified port.

Enabling the Backup Root Function

Disabled by default, the backup root function works only when the backup root-enabled bridge is directly connected to the root bridge. The backup root function prevents stale Spanning Tree

information from circulating throughout the network in the event that the link between the root bridge and the backup root-enabled bridge is lost. If this happens, the backup root will dynamically lower its bridge priority relative to the existing root bridge's priority, causing it to immediately be selected as the new root bridge.

Use this command to enable the backup root function on an SID:

```
set spantree backuproot sid enable
```

When SNMP trap messaging is configured and the backup root function is enabled, a trap message will be generated when the backup becomes the new root of the network.

Adjusting RSTP Parameters

Since rapid link reconfiguration can happen only on a point-to-point link or an edge port (a port that is known to be on the edge of a bridged LAN), in some cases you may want to define them administratively. However, since edge port and point-to-point links are automatically detected on Extreme Networks switching devices, in most cases you will not need to change these default port designations.

Defining Point-to-Point Links

By default, the administrative point-to-point status is set to auto on all Spanning Tree ports, allowing the Extreme Networks firmware to determine each port's point-to-point status. In most cases, this setting will not need to be changed and will provide optimal RSTP functionality. You can, however, use the following commands to review and, if necessary, change the point-to-point status of a Spanning Tree link.

Review and define the point-to-point status of an RSTP link as follows:

1. Display the point-to-point operating status of a LAN segment attached to a port:

   ```
   show spantree operpoint [port port-string]
   ```

   A status of "true" indicates the LAN segment is operating as a point-to-point link.

   A status of "false" indicates it is not.

   If *port-string* is not specified, point-to-point operating status will be displayed for all Spanning Tree ports.

2. Display the point-to-point administrative status of a LAN segment attached to a port:

   ```
   show spantree adminpoint [port port-string]
   ```

   A status of "true" indicates the port is administratively set to be considered point-to-point.

   A status of "false" indicates the port is administratively set to be considered non point-to-point.

   A status of "auto" (the default setting) indicates that the firmware is allowed to determine the port's point-to-point status.

   If *port-string* is not specified, point-to-point administrative status will be displayed for all Spanning Tree ports.

3. If necessary, change the point-to-point administrative status of a LAN segment attached to a port:

   ```
   set spantree adminpoint port-string {auto | true | false}
   ```

Defining Edge Port Status

By default, edge port status is disabled on all ports. When enabled, this indicates that a port is on the edge of a bridged LAN. You can use the following commands to review and, if necessary, change the edge port detection status on the device and the edge port status of Spanning Tree ports.

Review and define edge port status as follows:

1. Display the status of edge port detection:

   show spantree autoedge

2. If desired, enable edge port detection:

   set spantree autoedge enable

3. Display the edge port operating status of one or more port(s):

   show spantree operedge [**port** *port-string*]

   A status of "true" or "Edge-Port" indicates the port is operating as an edge port.

   A status of "false" or "Non-Edge-Port" indicates it is not.

   If *port-string* is not specified, edge port status will be displayed for all Spanning Tree ports.

4. Display the edge port administrative status of one or more port(s):

   show spantree adminedge [**port** *port-string*]

   A status of "true" or "Edge-Port" indicates the port is administratively set to be considered an edge port.

   A status of "false" or "Non-Edge-Port" indicates the port is administratively set to be considered a non edge port.

   If *port-string* is not specified, edge port administrative status will be displayed for all Spanning Tree ports.

5. If necessary, change the edge port administrative status of one or more port(s):

   set spantree adminedge *port-string* **true**

Configuring MSTP

In order for MSTP to provide multiple forwarding paths, the following must happen:

- The configuration identifier must match on all bridges within the region.
- All bridges must be within the same region.
- All bridges must be connected to MSTP-aware bridges. (They can be connected using a shared media such as a repeater provided that a single Spanning Tree device does not reside on that LAN).

Note: A single Spanning Tree device between two MSTP bridges will terminate the ability to have multiple forwarding paths.

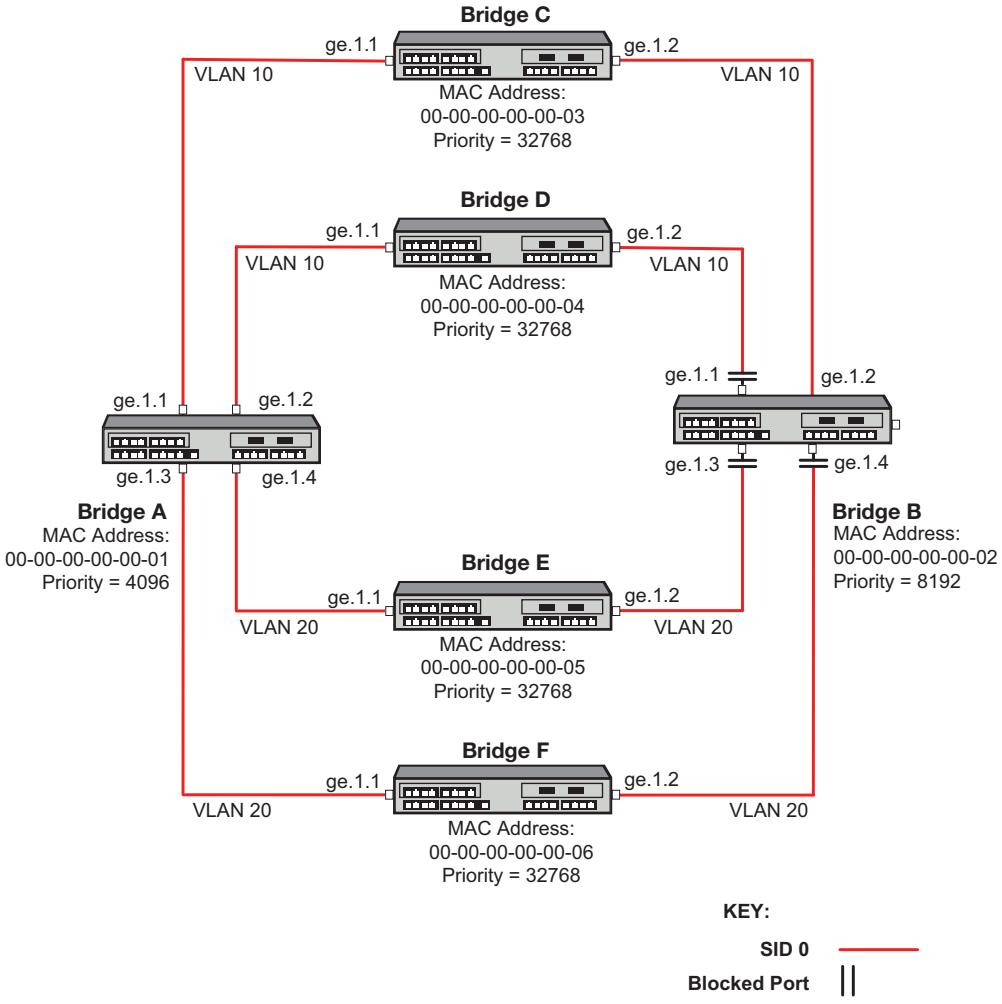
For information about...	Refer to page...
Example 1: Configuring MSTP for Traffic Segregation	21-28
Example 2: Configuring MSTP for Maximum Bandwidth Utilization	21-30
Adjusting MSTP Parameters	21-31

For information about...	Refer to page...
Monitoring MSTP	21-32

Example 1: Configuring MSTP for Traffic Segregation

This example illustrates the use of MSTP for traffic segregation by VLAN and SID. Bridges A, B, C and D participate in VLAN 10. Bridges A, B, E and F participate in VLAN 20. Figure 21-11 shows the problem that arises when using a single Spanning Tree configuration for traffic segregation with redundancy.

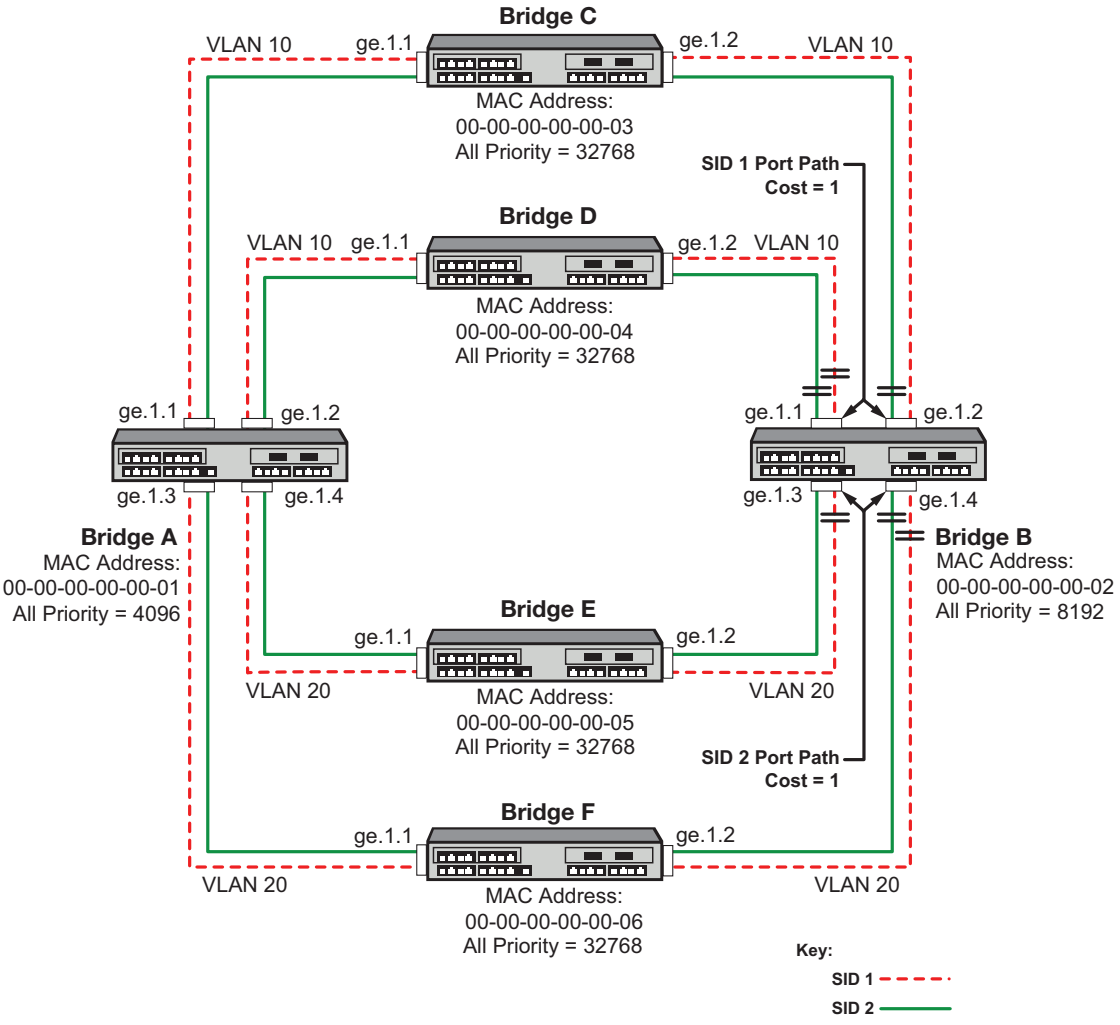
Figure 21-11 Traffic Segregation in a Single STP Network Configuration



In a single Spanning Tree configuration a bridge can only have one port forwarding towards the root for all traffic. Bridge A has the lowest priority and is the root. Bridge B forwards traffic towards the root on port ge.1.2. All other ports are blocked. For this configuration, Bridge B will not have any active links forwarding for VLAN 20.

Figure 21-12 shows the solution using MSTP. By configuring separate Spanning Tree instances to overlay the two VLAN topologies, Bridge B port ge.1.2 forwards on VLAN 10 for SID 1 and port ge.1.3 forwards on VLAN 20 for SID 2.

Figure 21-12 Traffic Segregation in an MSTP Network Configuration



To configure the traffic segregation MSTP example on all bridges:

- Configure the MST configuration ID with the same name

  set spantree mstcfgid cfgname *name*

- Create SIDs 1 and 2

  set spantree msti sid *sid* **create**

- Create the FID to SID mappings VLAN 10 to SID 1 and VLAN 20 to SID 2

  set spantree mstmap *vlan-id* **sid** *sid*

To configure Bridge A as root, set the priority to 4096 for both SID 1 and SID 2.

set spantree priority *priority sid*

To configure Bridge B as the backup should Bridge A fail:

- Set the Spanning Tree priority to 8192 for both SID 1 and SID 2

  set spantree priority *priority sid*

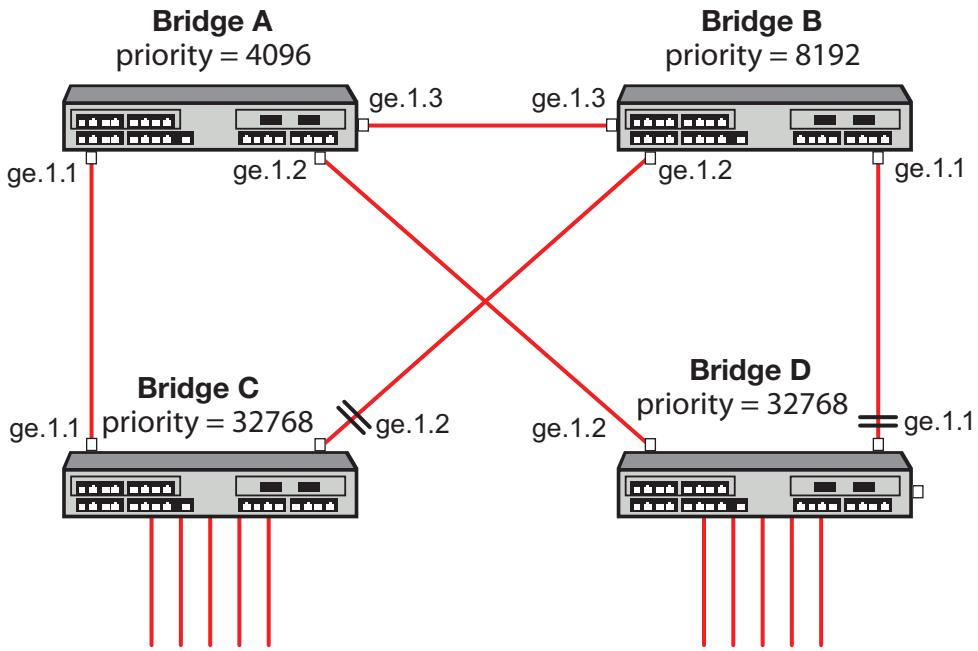
- Set the admin path cost on ports ge.1.1-2 to 1 for SID 1
- Set the admin path cost on ports ge.1.3-4 to 1 for SID 2

```
set spantree adminpathcost port-id cost sid
```

Example 2: Configuring MSTP for Maximum Bandwidth Utilization

This example illustrates the use of MSTP for maximum bandwidth utilization. Maximum bandwidth utilization takes place when all bridges participate on all VLANs. Figure 21-13 shows that with a single Spanning Tree configuration, only a single link towards the root forwards on a bridge. The alternate ports are blocking.

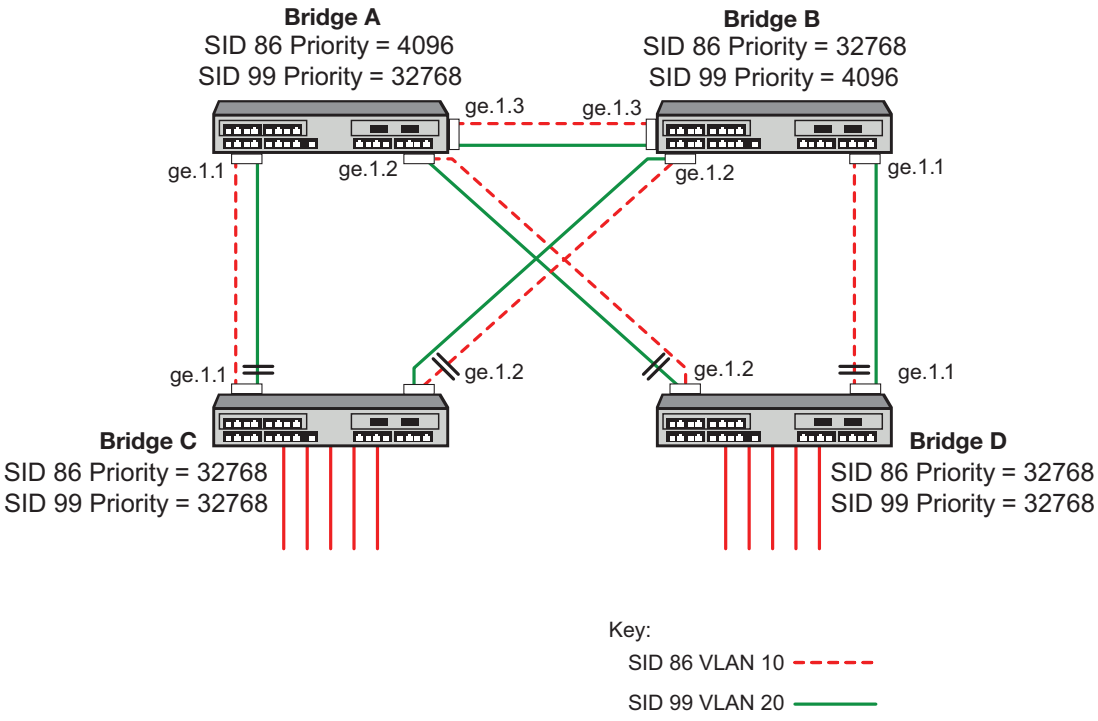
Figure 21-13 Maximum Bandwidth Utilization in a Single STP Network Configuration



In Figure 21-13, Bridge A is the root of the Spanning Tree because it has the lowest priority. Bridge D port ge.1.2 forwards traffic to Bridge A. Bridge D port ge.1.1 is blocking. Bridge C port ge.1.1 forwards traffic to Bridge A. Bridge C port ge.1.2 is blocking. This single Spanning Tree configuration prevents maximum bandwidth utilization for this network.

Figure 21-14 on page 21-31 shows that with an MSTP configuration each link can be forwarding for some VLAN and each VLAN has a path to the root bridge.

Figure 21-14 Maximum Bandwidth Utilization in an MSTP Network Configuration



To configure the MSTP maximum bandwidth utilization example on all bridges:

- Create VLANs 10 and 20

  set vlan create *vlan-id*

- Configure the MST configuration ID with the same name

  set spantree mstcfgid cfgname *name*

- Create SIDs 86 and 99

  set spantree msti sid *sid* **create**

- Create the FID to SID mappings VLAN 10 to SID 86 and VLAN 20 to SID 99

  set spantree mstmap *vlan-id* **sid** *sid*

Additionally, the root of each SID is chosen to be in a different bridge. This will spread out the traffic. The bridges on the next level down have a link to each of the root bridges.

To configure Bridge A as root for SID 86, set the priority to 4096 for SID 86.

set spantree priority *priority sid*

To configure Bridge B as the root for SID 99, set the priority to 4096 for SID 99.

Adjusting MSTP Parameters

You may need to adjust certain Spanning Tree parameters if the default values are not suitable for your bridge configuration. Refer back to Adjusting Spanning Tree Parameters on page 21-22 and Adjusting RSTP Parameters on page 21-26 for information on adjusting Spanning Tree defaults. Changes made to global and port-related Spanning Tree defaults will take affect if the device is running in STP, RSTP, or MSTP.

Monitoring MSTP

Use the commands in Table 21-6 to monitor MSTP statistics and configurations. You can also use the show commands described in Reviewing and Enabling Spanning Tree on page 21-21 to review information related to all Spanning Tree protocol activity.

Table 21-6 Commands for Monitoring MSTP

Task	Command
Verify that MSTP is running on the device.	**show spantree version**
Display the maximum configurable MSTIs allowed on the device.	**show spantree maxconfigurablestps**
Display a list of MSTIs configured on the device.	**show spantree mstilist**
Display the mapping of one or more filtering database IDs (FIDs) to Spanning Trees. Since VLANs are mapped to FIDs, this shows to which SID a VLAN is mapped.	**show spantree mstmap** [**fid** *fid*]
Display the Spanning Tree ID(s) assigned to one or more VLANs.	**show spantree vlanlist** [*vlan-list*]
Display MST configuration identifier elements, including format selector, configuration name, revision level, and configuration digest.	**show spantree mstcfgid**
Display protocol-specific MSTP counter information.	**show spantree debug** [**port** *port-string*] [**sid** *sid*] [**active**]

Understanding and Configuring SpanGuard

What Is SpanGuard?

As described previously in the overview of SpanGuard on page 21-7, this feature enables Extreme Networks switching devices to detect unauthorized bridges in your network, resolving the threat of repeated topology change notifications or new root bridge announcements causing a Denial of Service (DoS) condition. It prevents Spanning Tree respans that can occur when BPDUs are received on user ports and notifies you (network management) they were attempted.

If a SpanGuard enabled port receives a BPDU, it becomes locked and transitions to the blocking state. It will only transition out of the blocking state after a globally specified time or when it is manually unlocked.

By default, SpanGuard is globally disabled and must be globally enabled to operate on all user ports. For configuration information, refer to Configuring SpanGuard on page 21-33.

How Does It Operate?

SpanGuard helps protect against Spanning Tree Denial of Service (DoS) SpanGuard attacks as well as unintentional or unauthorized connected bridges, by intercepting received BPDUs on configured ports and locking these ports so they do not process any received packets.

When enabled, reception of a BPDU on a port that is administratively configured as a Spanning Tree edge port (adminedge = True) will cause the port to become locked and the state set to blocking. When this condition is met, packets received on that port will not be processed for a specified timeout period. The port will become unlocked when:

- the timeout expires,
- the port is manually unlocked,
- the port is no longer administratively configured as adminedge = True, or
- the SpanGuard function is disabled.

The port will become locked again if it receives another offending BPDU after the timeout expires or it is manually unlocked.

In the event of a DoS attack with SpanGuard enabled and configured, no Spanning Tree topology changes or topology reconfigurations will be seen in your network. The state of your Spanning Tree will be completely unaffected by the reception of any spoofed BPDUs, regardless of the BPDU type, rate received or duration of the attack.

By default, when SNMP and SpanGuard are enabled, a trap message will be generated when SpanGuard detects that an unauthorized port has tried to join a Spanning Tree.

Configuring SpanGuard

Use the following commands to configure device ports for SpanGuard, to enable the SpanGuard function, and to review SpanGuard status on the device.

Reviewing and Setting Edge Port Status

Note: To use the SpanGuard function, you must know which ports are connected between switching devices as ISLs (inter-switch links). Also, you must configure edge port status (adminedge = true or false) on the entire switch, as described in “Defining Edge Port Status” on page 21-27, before SpanGuard will work properly.

Review and set edge port status as follows:

1. Use the show commands described in “Defining Edge Port Status” on page 21-27 to determine edge port administrative status on the device.
2. Set edge port administrative status to false on all known ISLs.
3. Set edge port administrative status to true on any remaining ports where SpanGuard protection is desired. This indicates to SpanGuard that these ports are not expecting to receive any BPDUs. If these ports do receive BPDUs, they will become locked.

Enabling and Adjusting SpanGuard

Use this command to enable SpanGuard on the device:

```
set spantree spanguard enable
```

Use this command to adjust the SpanGuard timeout value. This sets the length of time that a SpanGuard-affected port will remain locked:

```
set spantree spanguardtimeout timeout
```

Valid values are 0–65535 seconds. Default is 300 seconds. Setting the value to 0 will set the timeout to forever.

Use this command to manually unlock a port that was locked by the SpanGuard function. This overrides the specified timeout variable:

set spantree spanguardlock *port-string*

Monitoring SpanGuard Status and Settings

Use the commands in Table 21-7 to review SpanGuard status and settings.

Table 21-7 Commands for Monitoring SpanGuard

Task	Command
Display the status of SpanGuard on the device.	**show spantree spanguard**
Display the status of the SpanGuard lock function on one or more ports.	**show spantree spanguardlock** [**port** *port-string*]
Display the SpanGuard timeout setting.	**show spantree spanguardtimeout**
Display the status of the SpanGuard trap function.	**show spantree spanguardtrapenable**

Understanding and Configuring Loop Protect

What Is Loop Protect?

As described previously in the overview of Loop Protect on page 8, this feature prevents or short circuits loop formation in your network. It does this by requiring ports to receive type 2 BPDUs (RSTP/MSTP) on point-to-point inter-switch links (ISLs) before their states are allowed to become forwarding. Further, if a BPDU timeout occurs on a port, its state becomes non-forwarding until a BPDU is received.

In this way, both upstream and downstream facing ports are protected. When a root or alternate port loses its path to the root bridge due to a message age expiration, it takes on the role of designated port and will not forward traffic until a BPDU is received.

When a port is intended to be the designated port in an ISL, it constantly proposes and will not forward until a BPDU is received. This protects against misconfiguration and protocol failure by the connected bridge.

How Does It Operate?

Loop Protect operates as a per port, per MST instance feature and should be set on ISLs. It comprises several related functions, including:

- Controlling port forwarding state based on reception of agreement BPDUs
- Controlling port forwarding state based on reception of disputed BPDUs

- Communicating port non-forwarding status through traps and syslog messages
- Disabling a port based on frequency of failure events

Port Modes and Event Triggers

Ports work in two Loop Protect operational modes. If the port is configured so that it is connected to a switching device known to implement Loop Protect, it uses full functional (enhanced) mode. Otherwise, it operates in limited functional (standard) mode.

Connection to a Loop Protect switching device guarantees that the alternate agreement mechanism is implemented and, therefore, the designated port can rely on receiving a response to its proposal regardless of the role of the connected port. This has two important implications. First, the designated port connected to a non-root port may transition to forwarding. Second, there is no ambiguity when a timeout happens; a Loop Protect event has occurred.

In full mode, when a type 2 BPDU is received and the port is designated and point-to-point, the timer is set to 3 times hello time. Limited mode adds a further requirement that the flags field in the BPDU indicates a root role. If the port is a boundary port, the MSTIs for that port follow the CIST (for example if the MSTI port timers are set according to the CIST port timer). If the port is internal to the region, the MSTI port timers are set independently using the particular MSTI message.

Loop Protect initializes the MSTI timer to zero and does not allow the designated port to transition from listening to learning until the timer becomes non-zero. If the port is not designated, the timer does not apply. Its state is controlled through normal protocol behavior.

A disputed BPDU is one in which the flags field indicates a designated role, a learning state, and the priority vector is worse than that already held by the port. If a disputed BPDU is received, the port is forced to the listening state.

Message age expiration and the expiration of the Loop Protect timer are both events for which Loop Protect generates a notice level syslog message. You can also configure traps to report these events, as well as a syslog message and trap for disputed BPDUs.

In addition, you can configure Loop Protect to force the locking of an SID/port when one or more events occur. When the configured number of events happen within a given window of time, the port will be forced into blocking and held there until you manually unlock it.

Example: Basic Loop Protect Configuration

The following sample configuration shows how Loop Protect functions in a basic Spanning Tree topology.

In the example in Figure 21-15, Switch 1 is the root bridge with BPDUs being sent to both Switch 2 and 3. (Designated ports are labeled D and root ports are labeled R.) Switch 3 has placed the port that connects to Switch 2 in a blocking state.

Figure 21-15 Basic Loop Protect Scenario

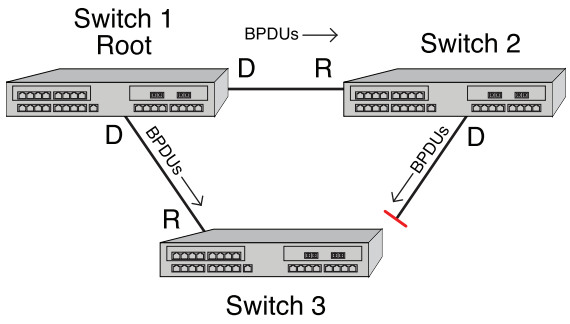


Figure 21-16 shows that, without Loop Protect, a failure could be as simple as someone accidentally disabling Spanning Tree on the port between Switch 2 and 3. Switch 3's blocking port eventually transitions to a forwarding state which leads to a looped condition.

Figure 21-16 Spanning Tree Without Loop Protect

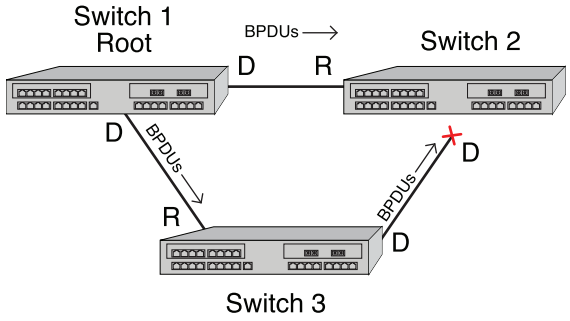
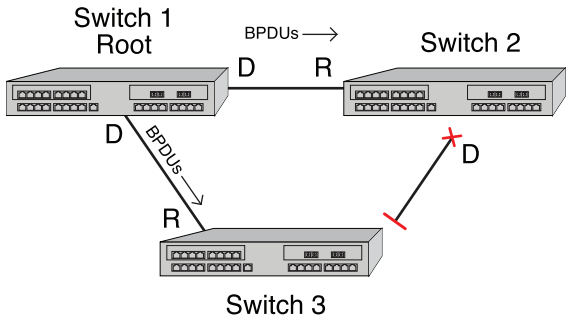


Figure 21-17 shows that, with Loop Protect enabled, Switch 3 will not go to a forwarding state until it has received a BPDU from Switch 2.

Figure 21-17 Spanning Tree with Loop Protect



Configuring Loop Protect

Enabling or Disabling Loop Protect

By default, Loop Protect is disabled on all ports. Use this command to enable (or, if desired, disable) the feature on one or more ports:

```
set spantree lp port-string {enable | disable} [sid sid]
```

If no SID is specified, SID 0 is assumed.

This command takes precedence over per port STP enable/disable state (portAdmin). Normally, portAdmin disabled would cause a port to go immediately to forwarding. If Loop Protect is enabled, that port should go to listening and remain there.

Note: The Loop Protect enable/disable settings for an MSTI port should match those for the CIST port.

Specifying Loop Protect Partners

By default, each port is not set as a Loop Protect capable partner. If the port is set as a Loop Protect capable partner (true), the full functionality of the Loop Protect feature is used. If the value is false, then there is some ambiguity as to whether an Active Partner timeout is due to a loop protection event or is a normal situation due to the fact that the partner port does not transmit Alternate Agreement BPDUs. Therefore, a conservative approach is taken in that designated ports will not be allowed to forward unless receiving agreements from a port with root role. This type of timeout will not be considered a loop protection event. Loop protection is maintained by keeping the port from forwarding, but since this is not considered a loop event, it will not be factored into locking the port.

Use this command to set the Loop Protect partner state on one or more ports:

```
set spantree lpcapablepartner port-string {true | false}
```

Setting the Loop Protect Event Threshold and Window

The Loop Protect event threshold is a global integer variable that provides protection in the case of intermittent failures. The default value is 3. If the event counter reaches the threshold within a given period (the event window), the port for the given SID becomes locked (that is, held indefinitely in the blocking state). If the threshold is 0, the ports are never locked.

Use this command to set the Loop Protect event threshold:

```
set spantree lpthreshold value
```

The Loop Protect window is a timer value, in seconds, that defines a period during which Loop Protect events are counted. The default value is 180 seconds. If the timer is set to 0, the event counter is not reset until the Loop Protect event threshold is reached.

Use this command to set the Loop Protect event window value in seconds:

```
set spantree lpwindow value
```

Enabling or Disabling Loop Protect Event Notifications

Loop Protect traps are sent when a Loop Protect event occurs, that is, when a port goes to listening due to not receiving BPDUs. The trap indicates port, SID and loop protection status.

Use this command to enable or disable Loop Protect event notification. By default, this is disabled:

```
set spantree lptrapenable {enable | disable}
```

Setting the Disputed BPDU Threshold

A disputed BPDU is one in which the flags field indicates a designated role and a learning state, and the priority vector is worse than that already held by the port. If a disputed BPDU is received, the port is forced to the listening state. Refer to the 802.1Q-2005 standard, *IEEE Standard for Local and Metropolitan Area Networks – Virtual Bridged Local Area Networks,* for a full description of the dispute mechanism, which prevents looping in cases of one-way communication.

The disputed BPDU threshold is an integer variable that represents the number of disputed BPDUs that must be received on a given port and SID before a disputed BPDU trap is sent and a syslog message is issued. For example, if the threshold is 10, a trap is issued when 10, 20, 30 (and so on) disputed BPDUs have been received. The trap indicates port, SID and total Disputed BPDU count.

Use this command to set the disputed BPDU threshold:

```
set spantree disputedbpduthreshold value
```

Default value is 0, which means that traps are not sent.

Monitoring Loop Protect Status and Settings

Use the commands in Table 21-8 to monitor Loop Protect settings.

Table 21-8 Commands for Monitoring Loop Protect

Task	Command
Display the Loop Protect status per port, per SID, or both.	**show spantree lp** [**port** *port-string*] [**sid** *sid*]
Display the Loop Protect lock status per port, per SID, or both. **Note:** A port can become locked if a configured number of Loop Protect events occur during the configured window of time. Once a port is forced into blocking (locked), it remains locked until manually unlocked with the **clear spantree lplock** command.	**show spantree lplock** [**port** *port-string*] [**sid** *sid*]
Display the Loop Protect capability of a link partner for one or more ports.	**show spantree lpcapablepartner** [**port** *port-string*]
Display the reason for placing a port in a non-forwarding state due to an exceptional condition.	**show spantree nonforwardingreason** [**port** *port-string*] [**sid** *sid*]

Example

The following example shows a switching device with Loop Protect enabled on port lag.0.2, SID 56:

```
Extremenetworks->show spantree lp port lag.0.2 sid 56
  LoopProtect is enabled on port lag.0.2, SID 56
Extremenetworks->show spantree lplock port lag.0.2 sid 56
  LoopProtect Lock status for port lag.0.2, SID 56_ is UNLOCKED
Extremenetworks->show spantree lpcapablepartner port lag.0.2
  Link partner of port lag.0.2_is LoopProtect-capable.
Extremenetworks->show spantree nonforwardingreason port lag.0.2
  Port lag.0.2 has been placed in listening or blocking state on SID 0 by the
  LoopProtect feature.
```

Terms and Definitions

Table 21-9 lists terms and definitions used in Spanning Tree configuration.

Table 21-9 Spanning Tree Terms and Definitions

Term	Definition
Alternate port	Acts as an alternate path to the root bridge than that provided by the root port.
Backup port	Acts as an backup for the path provided by a designated port toward the leaves of the Spanning Tree. Backup ports can exist only where two ports are connected together in a loopback mode or bridge with two or more connections to a shared LAN segment.
BID	Bridge identification, which is derived from the bridge's MAC address and bridge priority. The bridge with the lowest BID becomes the root bridge.
BPDU	Bridge Protocol Data Unit messages. Used by STP to exchange information, including designating a bridge for each switched LAN segment, and one root bridge for the Spanning Tree.
Bridge	Switching device.
Bridge priority	Assigns the bridge's relative priority compared to other bridges.
CIST	Common and Internal Spanning Tree created by MSTP to represent the connectivity of the entire network. This is equivalent to the single Spanning Tree used for STP and RSTP. Communications between MST regions occurs using the CIST.
CST	A Spanning Tree defined in the IEEE 802.1q standard that assumes one Spanning Tree instance for the entire bridged network, regardless of the number of VLANs.
Designated port	A forwarding port within an active topology elected for every switched LAN segment.
Edge port	Port on the edge of a bridged LAN.
FID	Filter Identifier. Each VLAN is associated to a FID. VLANs are mapped to SIDs using their FID association.
Forward delay	Time interval (in seconds) the bridge spends in listening or learning mode before it begins forwarding BPDUs.
Hello time	Time interval (in seconds) at which the bridge sends BPDUs.
ISL	Inter-Switch Link.

Table 21-9 Spanning Tree Terms and Definitions (continued)

Term	Definition
IST	A Spanning Tree instance that extends the CST inside the MST region and represents the entire MST region as a single CST virtual bridge to the outside world.
Loop Protect	Prevents or short circuits loop formation in a network with redundant paths by requiring ports to receive type 2 BPDUs (RSTP/MSTP) on point-to-point inter-switch links (ISLs) before their states are allowed to become forwarding.
Master port	The MSTI port whose connecting CIST port is root port for an entire MST region.
Max age	Maximum time (in seconds) the bridge can wait without receiving a configuration message (bridge “hello”) before attempting to reconfigure.
MST region	An MSTP group of devices configured together to form a logical region. The MST region presents itself to the rest of the network as a single device, which simplifies administration.
MSTI	Multiple Spanning Tree Instance. See Table 21-4 on page 22 for MSTI support per platform.
Path cost	Sum of the port costs in the best path to the root bridge.
Port cost	Value assigned to a port based on the speed of the port. The faster the speed, the lower the cost. This helps to determine the quickest path between the root bridge and a specified destination. The segment attached to the root bridge normally has a path cost of zero.
Port priority	Assigns a port’s priority in relation to the other ports on the same bridge.
Restricted TCN	Restricts the propagation of Topology Change Notices on a specified port when set to true.
Restricted Role	Disallows root role on a specified port when set to true.
Root bridge	Logical center of the Spanning Tree, used by STP to determine which paths to block and which to open.
Root port	Port in an active topology through which the root bridge can be reached.
SID	Spanning tree identifier. By default, SID 0 is assumed. VLANs are mapped to SIDs using their FID association.
SpanGuard	Prevents Spanning Tree respans that can occur when BPDUs are received on user ports and notifies network management that they were attempted.
TCN	Topology Change Notification.

22

Shortest Path Bridging (SPB) Configuration

This chapter provides information about configuring and monitoring Shortest Path Bridging (SPB) on S-Series devices.

Using Shortest Path Bridging (SPB) in Your Network

Shortest Path Bridging (SPB), IEEE 802.1aq, is a protocol that provides data traffic a shortest cost path between any pair of switches in the SPB network. SPB features dynamic route calculation in a loop-free Layer-2 network and fast convergence time using IS-IS. The S-Series supports Shortest Path Bridging VLAN (SPBV).

SPB is administratively enabled by default, but requires that Spanning Tree be configured with the Spanning Tree version set to SPT, and the MST configuration name must be the same on all devices for a given SPB region. You can configure multiple SPB regions by assuring that each device within a given SPB region has the same MST configuration name. Spanning Tree defaults to version 3 (MSTP).

SPB uses IS-IS link state to populate the network topology information and calculate the Shortest Path Trees (SPTs). No IS-IS configuration is required. IS-IS is automatically enabled with a default NET as part of SPB region setup. The SPB capable bridge provides user access to an SPB region. A bridge joins an SPB region by forming an adjacency with a neighboring bridge through the exchange of IS-IS PDUs.

The base-VLAN is the customer VLAN that ingresses the SPBV network. It is used to manage operations in the SPT and provides access to the SPBV network. The Shortest Path VID (SPVID) is a VLAN used to identify a base-VLAN and SPT within the SPB network. Once a packet ingresses the SPB network, the SPVID is used. When the packet egresses the SPB network, the SPVID is translated back to the base-VLAN.

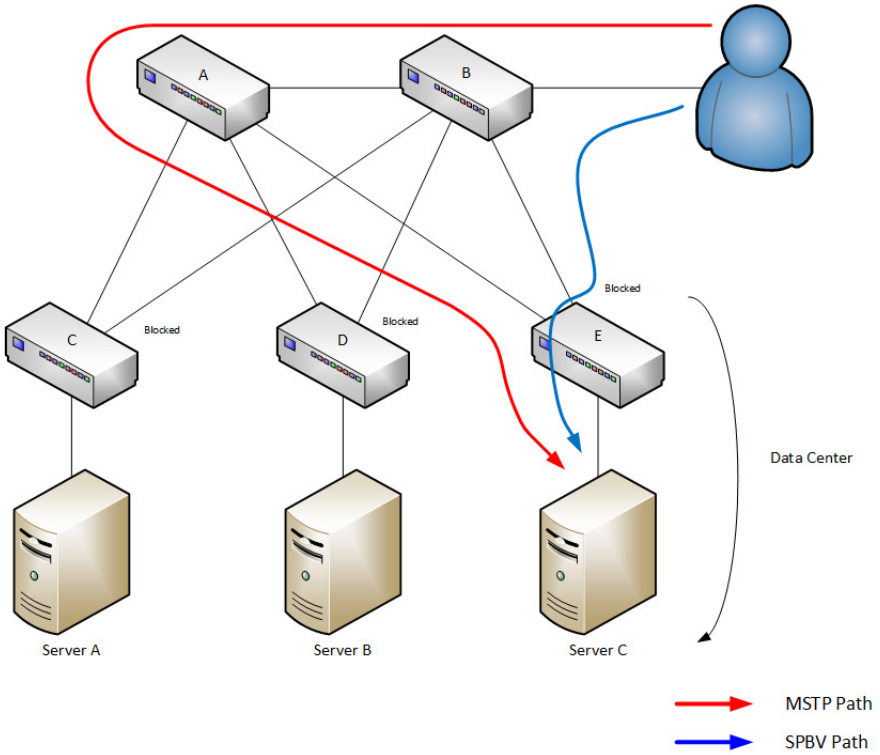
There are two types of ports in an SPBV region. Packets ingress and egress the SPB region on boundary ports. The boundary port is a member of the base-VLAN. Internal ports face into the SPB region and are members of the SPVID. An SPVID pool is administratively configured. The VLANs in the SPVID pool can only be used as SPVIDs. All SPB region ports must be administratively enabled for SPB.

Each base-VLAN at the boundary port is administratively assigned one of 16 Equal Cost Tree (ECT) algorithms as defined in IEEE 802.1Qaq. Each base-VLAN within the same SPB region must be assigned the same ECT algorithm. When traffic ingresses the SPB region on the base-VLAN, the packet VLAN (base-VLAN) is translated to an SPVID. By default this translation is dynamic, but you can optionally administratively set the base-VLAN to SPVID mapping.

Each VLAN on a bridge is associated with an SPVID whose tree is rooted at that bridge. This unique mapping is enforced by ISIS. As the traffic passes through the SPBV network, the SPVID is used. The base-VLAN and SPVID follow normal MAC and FID learning by making use of shared VLAN learning, but it's FID will be defined by the mapped base-VLAN. Since the SPVID is unique to a device and base-VLAN, the SPVID defines the source bridge of traffic received within the SPBV region. The SPVID pool must be a size equal to or greater than the number of base-VLANs times the number of nodes in the SPB region that switch base-VLAN traffic. The ECT determines the next switch in the path through the SPB network. When the packet egresses the SPB network at a boundary port, the SPVID is translated back to the appropriate VLAN before being forwarded.

Figure 22-1 on page 22-2 shows an example where a user is trying to access server C in the Data Center through use of a spanning tree switched network. Spanning tree determines bridge A is the root of the CIST, and is blocking ports on bridges C, D, and E. The red line would be a normal MSTP switched traffic path, which makes an extra hop through router A. The shortest path is skipped due to the blocked port. The SPB network comprises a set of trees, each tree rooted at a switch where traffic ingresses the region. No ports are blocked and ISIS will create a shortest path through B and E to server C on the blue line.

Figure 22-1 SPB Overview

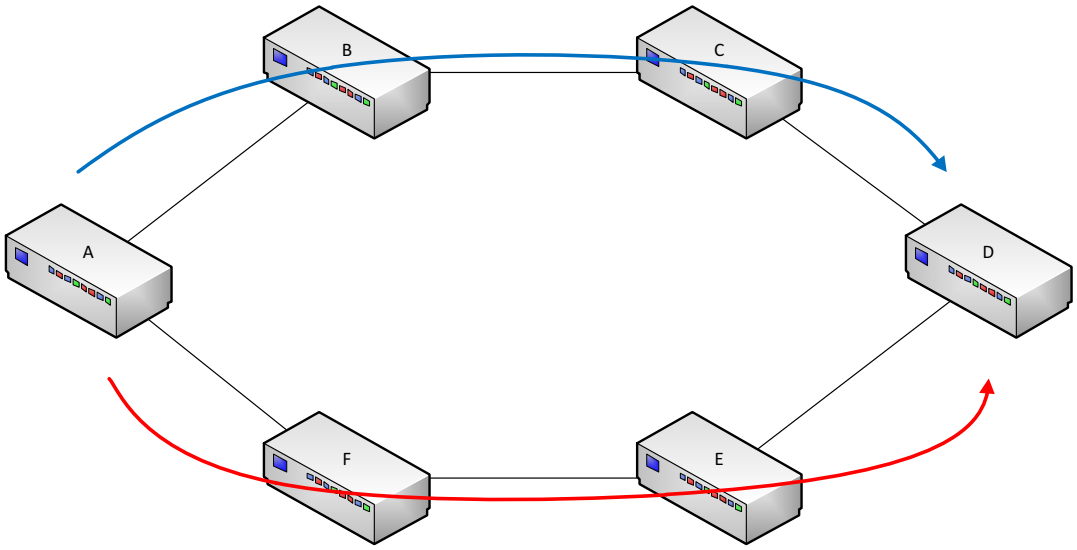


SPB provides benefits in addition to lower latency due to fewer hops. Use of IS-IS allows for quicker network recovery times than traditional Spanning Tree. SPB also provides for a more even distribution of traffic so switches, such as bridge A in Figure 22-1, do not become bottlenecks. SPB makes use of Equal Cost Trees (ECTs) to further refine traffic distribution.

Traffic can be mapped to access the SPBV backplane with different base-VLANs. A base-VLAN with one ECT algorithm may make a different path choice than another base-VLAN with a different ECT algorithm when faced with paths of equal cost. A network can be set up where traffic received on specific VLANs outside the SPBV network will all map to a single base-VLAN and will use the same ECT path, for example the blue line in Figure 22-2 on page 22-3. Other VLANs can be mapped to a different base-VLAN and use a different ECT path, as indicated by the red line. All devices that are associated with a particular base-VLAN must be configured with the same ECT algorithm.

SPBV provides shortest paths without requiring changes in MTU size because it performs VID translation rather than adding its own header. It also fully supports congestion notification.

Figure 22-2 SPBV Using Equal Cost Trees



Implementing Shortest Path Bridging

To implement SPBV:

1. Configure Spanning Tree on all devices in the region:
   - Set the Spanning Tree version to SPT on all devices in the SPB region
   - Configure the same MST configuration name on all devices in the SPB region
2. Configure an SPVID pool for this SPB region (same VLAN range for all devices in the region).
3. Assign the base-VLANs that will be used to ingress and egress the SPB region to SID **4093** or SID **spbv**.
4. Enable SPB on all ports that will take part in the SPB region.
5. Optionaly, assign the desired ECT algorithm to each configured base-VLAN (unless the default algorithm is desired).
6. Optionally, administratively assign the base-VLAN to SPVID mapping for the base-VLAN on each device in the SPB region. When administratively assigning the base-VLAN to SPVID mapping, change the SPB VLAN mode to manual.

Shortest Path Bridging VLAN Configuration Overview

SPBV Spanning Tree Configuration

Spanning Tree must be configured for SPBV to be operational. Configure Spanning Tree as documented in the *Extreme Networks S-Series Configuration Guide*. There are two important considerations when configuring Spanning Tree for SPBV:

- Spanning Tree version
- MST configuration name

Spanning Tree version defaults to MSPT. SPBV requires Spanning Tree version SPT. Use the **set spantree version** command in any configuration mode to set the Spanning Tree version to SPT. The following example sets the Spanning Tree version to SPT:

```
S Chassis(rw)->set spantree version spt
S Chassis(rw)->
```

The Spanning Tree MST configuration name must be the same for all devices in the SPBV region. The MST configuration name is set using the set spantree mstcfgid command in any configuration mode. The following example sets the MST configuration name to **spbv1** for this device:

```
S Chassis(rw)->set spantree mstcfgid cfgname spbv1
S Chassis(rw)->
```

SPVID Pool

Shortest Path VLANs (SPVIDs) are the VLANs used by the SPBV region internal ports. An SPVID pool reserves a set of VLANs for SPVID use only. The number of VLANs reserved in the SPVID pool must be equal to or greater than the number of base-VLANs times the number of nodes in the SPBV region that switch base-VLAN traffic. For example if there are two base-VLANs, twenty nodes in the region, but only 16 forward base-VLAN-tagged traffic, the minimum number of VLANs needed in the SPVID pool is 2*16 or 32 VLANs.

Use the **set spantree mstmap** command to specify a range of VLANs to reserve for the SPVID pool. The keyword **spvid** specifies SID **4095** for this Spanning Tree instance.

Note: The SPVID pool configuration must be the same for all nodes in a given SPBV region.

The following example sets the SPVID pool range to VLANs 2000 through 3000 for this device.

```
S Chassis(rw)->set spantree mstmap 2000-3000 sid spvid
S Chassis(rw)->
```

Assigning a Base-VLAN to Use SPB

Base-VLANs within the Spanning Tree domain are assigned to use SPB by changing the SID to 4093. The keyword **spbv** can be used in place of 4093.

Use the **set spantree mstmap** command to specify one or more Base-VLANs to use SPB.

The following example configures Base-VLANs 100, 200, 300, and 400 to use SPB.

```
S Chassis(rw)->set spantree mstmap 100,200,300,400 sid spbv
S Chassis(rw)->
```

Base-VLAN Configuration

There are two aspects to base-VLAN configuration:

- ECT algorithm assignment
- Base-VLAN to SPVID mapping

The base-VLAN belongs to the boundary port where packets ingress and egress the SPBV region. The base-VLAN manages operations in the SPT and provides access to the SPB network. A given base-VLAN in an SPBV region must be assigned the same ECT algorithm. The ECT determines the next switch in the path through the SPB. There are 16 ECT algorithms defined in IEEE 802.1Qaq.

Use the **set spb basevid ect-alg** command in any command mode to assign an ECT algorithm to a base-VLAN.

The following example assigns ECT algorithm **2** to base-VLAN **100**:

```
S Chassis(rw)->set spb basevid 100 ect-alg ieee 2
S Chassis(rw)->
```

By default, base-VLANs are dynamically mapped to a unique SPVID from the SPVID pool for each node in the SPBV region. You can administratively map the base-VLAN to an SPVID for each node in the SPBV region. To administratively configure the base-VLAN to SPVID mapping:

- Set the SPB VLAN mode to manual using the **set spb system mode-vlan** command in any command mode
- Map the SPVID to the base-VLAN using the **spvid** option of the **set spb basevid ect-alg** command

The following example maps SPVID **2000** to base-VLAN **100**:

```
S Chassis(rw)->set spb system mode-vlan manual
S Chassis(rw)->set spb basevid 100 ect-alg ieee 2 spvid 2000
S Chassis(rw)->
```

SPB Ports

SPB must be enabled on all ports in the SPBV region. SPB defaults to disabled on ports. Use the **set spb port status enable** command in any command mode to enable SPB on all ports accessing the SPBV region.

The following example enables SPB on ports **ge.1.1** through **ge.1.5**:

```
S Chassis(rw)->set spb port ge.1.1-5 status enable
S Chassis(rw)->
```

Configuring Shortest Path Bridging VLAN

This section provides a table of Shortest Path Bridging default values and a procedure for configuring a Shortest Path Bridging system.

Table 22-1 lists Shortest Path Bridging default values.

Table 22-1 Default Shortest Path Bridging Parameters

Parameter	Description	Default Value
SPB Status	Shortest Path Bridging device global state.	enabled
SPB Port Status	Shortest Path Bridging port state.	disabled
SPB System VLAN Mode	Configures the mode that determines whether the base-VID to SPVID mapping is manual or dynamic (auto).	auto
SPB Digest-Convention	Configures the SPB agreement digest convention.	loopfreeboth
Spanning Tree Version	Configures the Spanning Tree version; SPB requires SPT version 4	MSTP (version 3)

Procedure 22-1 describes Shortest Path Bridging configuration on the Extreme Networks S-Series devices. All commands used to configure Shortest Path Bridging can be entered in any command mode.

Procedure 22-1 Configuring Shortest Path Bridging

Step	Task	Command(s)
1.	Configure Spanning Tree on all SPB region devices.	See the *Extreme Networks S-Series Configuration Guide* Spanning Tree Configuration information.
2.	Enable SPB globally on the device. SPB is globally enabled by default.	**set spb status** {**enable** | **disable**}
3.	Set the Spanning Tree version to SPT on all devices in the SPB region.	**set spantree version spt**
4.	Configure the same Spanning Tree MST configuration name on all devices in the SPB region.	**set spantree mstcfgid** {[**cfgname** *name*] [**rev** *level*]}
5.	Configure an SPVID pool for this SPB region.	**set spantree mstmap** spvid-list **sid** {**4095** | **spvid**}
6.	Configure base-VLANs to use SPB.	**set spantree mstmap** baseVid-list **sid** {**4093** | **spbv**}
7.	Assign the ECT algorithm for each base-VLAN on this device.	**set spb basevid** *baseVid* **ect-alg ieee** *ect-alg*
8.	Optionally, set SPB system parameters: IS-IS area, digest convention, VLAN mode, or system ID.	**set spb system** [**area-address** *isis-area*] [**digest-convention** {**off** | **loopfreeboth**}] [**mode-vlan** {**auto** | **manual**}]
9.	Optionally, administratively assign the base-VLAN to SPVID mapping for the base-VLAN on each device in the SPB region.	**set spb basevid** *baseVid* **spvid** *spVid*
10.	Enable SPB on all ports that will take part in the SPB region.	**set spb port** *port-string* **status enable**

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

Terms and Definitions

Table 22-2 lists terms and definitions used in this security mode configuration discussion.

Table 22-2 Shortest Path Bridging Configuration Terms and Definitions

Term	Definition
Shortest Path Bridging (SPB)	Shortest Path Bridging (SPB), defined in IEEE 802.1aq, is a protocol that provides data traffic a shortest cost path between any pair of switches in the SPB network, and features dynamic route calculation in a loop-free Layer-2 network, and fast convergence time using IS-IS.
Base-VLAN	The VLAN located on the SPBV boundary port that manages operations in the SPT and provides access to the SPBV network.
Shortest Path VID (SPVID)	The VLAN located on the port internal to the SPB region used to identify a base-VLAN and SPT within the SPB network.
SPB Boundary Port	The SPB port packets ingress and egress the SPB region on located on the base-VLAN.
SPB Internal Port	A port that faces into the SPB region and is a member of the SPVID.
Equal Cost Tree (ECT) algorithm	One of 16 algorithms defined in the IEEE 802.1Qaq standard used to select a path among available equal cost paths.

23

Routing as a Service (RaaS) Configuration

This chapter provides information about configuring and Monitoring Routing as a Service (RaaS) on S-Series devices.

Using Routing as a Service (RaaS) in Your Network

Routing as a Service (RaaS), also known as Virtual Fabric Routing, provides for scalable and efficient virtualized routing over any L2 SPB network infrastructure by scaling the fabric from a single chassis to a collection of devices that use L2 protocols to form its topology. The L2 topology protocol can form a single path service like Spanning Tree or a multipath service like Shortest Path Bridging (SPB). RaaS presumes the L2 service can proliferate all VLANs to all devices.

RaaS offers a routing solution that efficiently utilizes the L2 infrastructure by leveraging its topology protocols in place of L3 protocols. RaaS supports the establishment of a network-wide, distributed virtual routing system where all of the devices in the SPB network work as a single and collective layer 3 forwarding mechanism, allowing routing to become an integrated service of the layer-2 domain.

RaaS functions within a SPB network. SPB is an emerging layer 2 technology defined by IEEE that augments Spanning Tree to utilize multiple paths and defines Shortes Path Bridging VLANs (SPBV). In a traditional SPBV network, routers attach at the edge to forward traffic between customer VLANS. Although a viable solution, routing at the edge of the SPB network typically does not provide the most direct path through the network. Routed packets first egress the layer-2 network on one VLAN to a connecting router which forwards them onto another VLAN within the same layer-2 network, thereby traversing the layer-2 network twice.

RaaS provides an integrated routing service that leverages layer-2 features such as VLAN propagation, multipath topology, fast convergence, and MAC reachability to provide a simpler and efficient routing service that eliminates all routing protocols within the SPB network. By eliminating routing protocols, this feature can scale to support routing across the VLAN interfaces that may be present in an SPB domain.

There are two types of routers within an RaaS network:

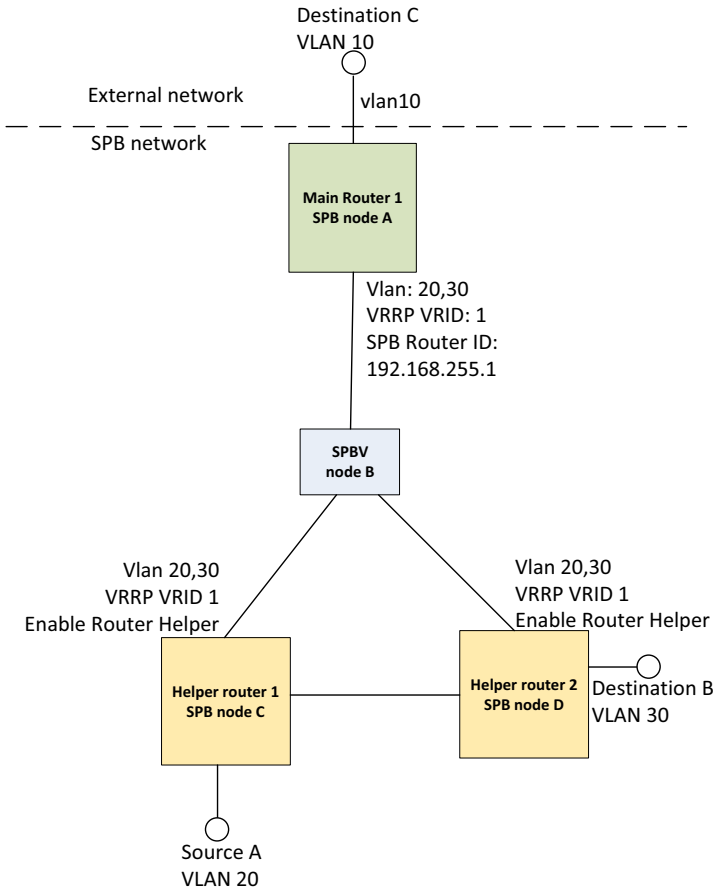
- Helper Router – An SPB switch node that ingresses customer VLANs, is configured for VRRP, contains route table entries confined to connected VLAN interfaces, and is able to forward all packets with destinations external to the connected customer VLANs to the Main Router.
- Main Router – A standard L3 router within the SPB network that is configured for VRRP and is used to forward all packets external to the directly connected Helper Router VLANs.

RaaS counts on hosts within a layer-2 domain being no more than one routed hop away. Assuming all VLAN interfaces are on every edge device, RaaS Helper Routers route directly to their destinations using layer-2 services to perform the multipath and MAC reachability. Packets are forwarded to a Main Router only when an RaaS client cannot route the packet. RaaS distributes routing throughout the SPB network leaving more complex IP forwarding to a few selected Main Routers.

RaaS helper routing devices utilize virtual IP addressing concepts described by VRRP allowing for simple and shared routing configurations to be deployed on all participating devices.

Figure 23-1 displays a RaaS overview.

Figure 23-1 Routing as a Service Overview



Helper Routers 1 and 2 are each configured for VLANs 20 and 30, with VRRP configured for VRID 1, and the Helper router feature enabled on the VRRP. A packet on Source A VLAN 20 destined for Destination B on VLAN 30 will be routed by Helper Router 1 then forwarded to Destination B over the shortest path.

SPB node B is configured as a transit node for VLANs 20 and 30. Node B does not have direct access to customer destinations so it is not configured as a Helper router.

Main Router 1 is configured for VLANs 20, 30, and 10, with VRRP configured for VRID 1 on VLANs 20 and 30, and SPB router ID 192.168.255.1. All Helper routers on VRID 1 are aware of Main Router 1 through the SPB router ID configuration.

A packet sourced on Source A VLAN 20 with a destination of Destination C on VLAN 10 will be forwarded to the Main Router 1 on the shortest path then routed to VLAN 10, destination C.

Implementing Routing as a Service

To implement RaaS:

1. Configure SPBV on all nodes within the SPB network as detailed in Chapter 22, **Shortest Path Bridging (SPB) Configuration**
2. Identify all customer facing VLANs within the SPB network and assure that all nodes within the SPB network have access to those VLANs
3. Identify all nodes in the SPB network that are directly connected to customers that will be configured as SPB Helper routers
4. Identify L3 routers with VLANs exterior to the SPB network that will be configured as Main routers
5. Configure VRRP on all customer VLAN interfaces interior to the SPB network on each Main and Helper router, enabling the helper-router function on each Helper router VLAN interface
6. Configure a unique SPB router ID on each Main router

Routing as a Service Configuration Overview

Helper Router Configuration

The Helper router is enabled under VRRP per VRID per VLAN interface on access SPB switches that ingress to customer VLANs. Helper router route tables confine routes to connected VLAN interfaces where an interface represents a customer VLAN. The Helper router learns the identity of Main routers by the propagation of type 250 TLV through the SPB network by IS-IS. Helper routers redirect unresolved destination networks to the Main routers.

The Main router responds to ARP requests for any virtual IP address and sends VRRP advertisements to ensure the virtual MAC remains in bridge FDBs within the SPB domain. Helper routers install the VRRP virtual MAC address into the local filter database for packet processing by the forwarding plane.

Use the **vrrp fabric-route-mode** command in interface configuration mode, specifying the VRID and the **helper-router** option, to enable this SPB node as a Helper router.

The following example configures VLAN 20 for Helper routing on the Helper router node by:

- Entering configuration mode for VLAN **20**
- Configuring the interface IP address as **10.1.20.2/24**

- Creating VRRP as VRID **1** for version **v2-ipv4**
- Configuring the VRRP IP address as **10.1.20.3**
- Enabling the Helper router feature
- Enabling VRRP for VRID **1**

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 20
S Chassis(rw)-config-intf-vlan.0.20)->ip address 10.1.20.2/24
S Chassis(rw)-config-intf-vlan.0.20)->vrrp create 1 v2-ipv4
S Chassis(rw)-config-intf-vlan.0.20)->vrrp address 1 10.1.20.3
S Chassis(rw)-config-intf-vlan.0.20)->vrrp fabric-route-mode 1 helper-router
S Chassis(rw)-config-intf-vlan.0.20)->vrrp enable 1
S Chassis(rw)-config-intf-vlan.0.20)->no shutdown
S Chassis(rw)-config-intf-vlan.0.20)->exit
S Chassis(rw-config)->
```

Main Router Configuration

The Main router forwards any packets destined for interfaces exterior to the SPB network, or for whatever reason are not resolved by the Helper routers within the RaaS. VRRP is configured on interfaces within the SPB network.

Main routers identify themselves to the Helper routers by way of an Main router ID unique to each Main router in the format of an IPv4 address.

Use the **raas** command, in global router configuration mode, to specify the Main router ID.

The following example configures the Main router ID and VLAN 20 for Main routing on the Main router node by:

- Configuring the Main router ID as **192.168.255.1**
- Entering configuration mode for VLAN **20**
- Configuring the interface IP address as **10.1.20.1/24**
- Enable fabric route mode on the main router to allow backup VRRP routers to forward
- Creating VRRP as VRID **1** for version **v2-ipv4**
- Configuring the VRRP IP address as **10.1.20.3**
- Enabling VRRP for VRID **1**

```
S Chassis(rw)->
S Chassis(rw)->configure
S Chassis(rw-config)->raas 192.168.255.1
S Chassis(rw-config)->interface vlan 20
S Chassis(rw)-config-intf-vlan.0.20)->ip address 10.1.20.1/24
S Chassis(rw)-config-intf-vlan.0.20)->vrrp create 1 v2-ipv4
S Chassis(rw)-config-intf-vlan.0.20)->vrrp address 1 10.1.20.3
S Chassis(rw)-config-intf-vlan.0.20)->vrrp fabric-route-mode 1
S Chassis(rw)-config-intf-vlan.0.20)->vrrp enable 1
S Chassis(rw)-config-intf-vlan.0.20)->no shutdown
S Chassis(rw)-config-intf-vlan.0.20)->exit
```

```
S Chassis(rw-config)->
```

Configuring Routing as a Service

Procedure 23-1 describes RaaS configuration on the Extreme Networks S-Series devices.

Procedure 23-1 Configuring RaaS

Step	Task	Command(s)
1.	Configure SPBV on all nodes within the SPB network.	See Chapter 22, **Shortest Path Bridging (SPB) Configuration** for SPBV configuration details.
2.	Configure VRRP on all VLAN interfaces interior to the SPB network on each Main and Helper router.	See Chapter 48, **Virtual Router Redundancy Protocol (VRRP) Configuration** for VRRP configuration details.
3.	Enable VRRP Helper routing mode on all Helper routers in the SPB network.	**vrrp fabric-route-mode** *vrid* **helper-router**
4.	Configure the Main router RaaS router ID on each Main router.	**raas** *router-id*

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

RaaS Configuration Example

The following example configuration is for a basic RaaS SPB network as displayed in Figure 23-1 on page 23-2. See Chapter 22, **Shortest Path Bridging (SPB) Configuration** for details on SPB network configuration for each device. SPBV node B has no attached customers and therefore is not part of the RaaS configuration. RaaS configuration is required for:

- Main Router 1 SPB node A
- Helper Router 1 SPB node C
- Helper Router 2 SPB node D

Main Router 1 SPB Node A

Main Router 1 is configured with a Loopback **0**, VLAN **1** management interface, VLAN **10** for access external to the SPB, and VLAN **20** and VLAN **30** for interfaces internal to the SPB network. This basic configuration contains a single VRRP VRID. A non-basic RaaS configuration may contain multiple VRIDs as needed. The Main router ID is set to **192.168.255.1**.

```
S Chassis(rw)->
S Chassis(rw)->configure
S Chassis(rw-config)->raas 192.168.255.1
S Chassis(rw-config)->interface loopback.0.0
S Chassis(rw-config-intf-loop.0.0)->ip address 192.168.255.1/32
S Chassis(rw-config-intf-loop.0.0)->no shutdown
S Chassis(rw-config-intf-loop.0.0)->exit
S Chassis(rw-config)->interface vlan 1
S Chassis(rw)-config-intf-vlan.0.1)->description “management”
```

```
S Chassis(rw)-config-intf-vlan.0.1)->ip address 192.168.1.10/24 255.255.255.0
primary
S Chassis(rw)-config-intf-vlan.0.1)->no ip proxy-arp
S Chassis(rw)-config-intf-vlan.0.1)->no ip forwarding
S Chassis(rw)-config-intf-vlan.0.1)->no shutdown
S Chassis(rw)-config-intf-vlan.0.1)->exit
S Chassis(rw-config)->interface vlan 10
S Chassis(rw)-config-intf-vlan.0.10)->description "externalnetwork"
S Chassis(rw)-config-intf-vlan.0.10)->ip address 192.168.1.1/24
S Chassis(rw)-config-intf-vlan.0.10)->no shutdown
S Chassis(rw)-config-intf-vlan.0.10)->exit
S Chassis(rw-config)->interface vlan 20
S Chassis(rw)-config-intf-vlan.0.20)->description "spbnetwork20"
S Chassis(rw)-config-intf-vlan.0.20)->ip address 10.1.20.1/24
S Chassis(rw)-config-intf-vlan.0.20)->vrrp create 1 v2-ipv4
S Chassis(rw)-config-intf-vlan.0.20)->vrrp address 1 10.1.20.3
S Chassis(rw)-config-intf-vlan.0.20)->vrrp enable 1
S Chassis(rw)-config-intf-vlan.0.20)->no shutdown
S Chassis(rw)-config-intf-vlan.0.20)->exit
S Chassis(rw-config)->interface vlan 30
S Chassis(rw)-config-intf-vlan.0.30)->description "spbnetwork30"
S Chassis(rw)-config-intf-vlan.0.30)->ip address 10.1.30.1/24
S Chassis(rw)-config-intf-vlan.0.30)->vrrp create 1 v2-ipv4
S Chassis(rw)-config-intf-vlan.0.30)->vrrp address 1 10.1.30.3
S Chassis(rw)-config-intf-vlan.0.30)->vrrp enable 1
S Chassis(rw)-config-intf-vlan.0.30)->no shutdown
S Chassis(rw)-config-intf-vlan.0.30)->exit
S Chassis(rw-config)->router ospf 1
S Chassis(rw-config-ospf-1)->router-id 192.168.255.1
S Chassis(rw-config-ospf-1)->network 192.168.10.0 0.0.0.255 area 0.0.0.0
S Chassis(rw-config-ospf-1)->exit
S Chassis(rw-config)->
```

Helper Router 1 SPB Node C

The Helper router 1 is configured with a VLAN 1 management interface, and VLAN **20** and VLAN **30** for interfaces internal to the SPB network. On both internal VLANs, VRRP is enabled with Helper router mode.

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface loopback.0.0
S Chassis(rw-config-intf-loop.0.0)->ip address 192.168.255.10/32
S Chassis(rw-config-intf-loop.0.0)->no shutdown
S Chassis(rw-config-intf-loop.0.0)->exit
S Chassis(rw-config)->interface vlan 1
S Chassis(rw)-config-intf-vlan.0.1)->description "management"
```

```
S Chassis(rw)-config-intf-vlan.0.1)->ip address 192.168.1.2/24 255.255.255.0
primary
S Chassis(rw)-config-intf-vlan.0.1)->no ip proxy-arp
S Chassis(rw)-config-intf-vlan.0.1)->no ip forwarding
S Chassis(rw)-config-intf-vlan.0.1)->no shutdown
S Chassis(rw)-config-intf-vlan.0.1)->exit
S Chassis(rw-config)->interface vlan 20
S Chassis(rw)-config-intf-vlan.0.20)->description “spbnetwork20”
S Chassis(rw)-config-intf-vlan.0.20)->ip address 10.1.20.2/24
S Chassis(rw)-config-intf-vlan.0.20)->vrrp create 1 v2-ipv4
S Chassis(rw)-config-intf-vlan.0.20)->vrrp address 1 10.1.20.20
S Chassis(rw)-config-intf-vlan.0.20)->vrrp fabric-route-mode 1 helper-router
S Chassis(rw)-config-intf-vlan.0.20)->vrrp enable 1
S Chassis(rw)-config-intf-vlan.0.20)->no shutdown
S Chassis(rw)-config-intf-vlan.0.20)->exit
S Chassis(rw-config)->interface vlan 30
S Chassis(rw)-config-intf-vlan.0.30)->description “spbnetwork30”
S Chassis(rw)-config-intf-vlan.0.30)->ip address 10.1.30.3/24
S Chassis(rw)-config-intf-vlan.0.30)->vrrp create 1 v2-ipv4
S Chassis(rw)-config-intf-vlan.0.30)->vrrp address 1 10.1.30.30
S Chassis(rw)-config-intf-vlan.0.30)->vrrp fabric-route-mode 1 helper-router
S Chassis(rw)-config-intf-vlan.0.30)->vrrp enable 1
S Chassis(rw)-config-intf-vlan.0.30)->no shutdown
S Chassis(rw)-config-intf-vlan.0.30)->exit
S Chassis(rw-config)->
```

Helper Router 2 SPB Node D

The Helper router 2is configured with a VLAN 1 management interface, and VLAN **20** and VLAN **30** for interfaces internal to the SPB network. On both internal VLANs, VRRP is enabled with Helper router mode.

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface loopback.0.0
S Chassis(rw-config-intf-loop.0.0)->ip address 192.168.255.20/32
S Chassis(rw-config-intf-loop.0.0)->no shutdown
S Chassis(rw-config-intf-loop.0.0)->exit
S Chassis(rw-config)->interface vlan 1
S Chassis(rw)-config-intf-vlan.0.1)->description “management”
S Chassis(rw)-config-intf-vlan.0.1)->ip address 192.168.1.3/24 255.255.255.0
primary
S Chassis(rw)-config-intf-vlan.0.1)->no ip proxy-arp
S Chassis(rw)-config-intf-vlan.0.1)->no ip forwarding
S Chassis(rw)-config-intf-vlan.0.1)->no shutdown
S Chassis(rw)-config-intf-vlan.0.1)->exit
S Chassis(rw-config)->interface vlan 20
S Chassis(rw)-config-intf-vlan.0.20)->description “spbnetwork20”
```

```
S Chassis(rw)-config-intf-vlan.0.20)->ip address 10.1.20.2/24
S Chassis(rw)-config-intf-vlan.0.20)->vrrp create 1 v2-ipv4
S Chassis(rw)-config-intf-vlan.0.20)->vrrp address 1 10.1.20.20
S Chassis(rw)-config-intf-vlan.0.20)->vrrp fabric-route-mode 1 helper-router
S Chassis(rw)-config-intf-vlan.0.20)->vrrp enable 1
S Chassis(rw)-config-intf-vlan.0.20)->no shutdown
S Chassis(rw)-config-intf-vlan.0.20)->exit
S Chassis(rw-config)->interface vlan 30
S Chassis(rw)-config-intf-vlan.0.30)->description "spbnetwork30"
S Chassis(rw)-config-intf-vlan.0.30)->ip address 10.1.30.3/24
S Chassis(rw)-config-intf-vlan.0.30)->vrrp create 1 v2-ipv4
S Chassis(rw)-config-intf-vlan.0.30)->vrrp address 1 10.1.30.30
S Chassis(rw)-config-intf-vlan.0.30)->vrrp fabric-route-mode 1 helper-router
S Chassis(rw)-config-intf-vlan.0.30)->vrrp enable 1
S Chassis(rw)-config-intf-vlan.0.30)->no shutdown
S Chassis(rw)-config-intf-vlan.0.30)->exit
S Chassis(rw-config)->
```

Terms and Definitions

Table 23-1 lists terms and definitions used in this security mode configuration discussion.

Table 23-1 RaaS Configuration Terms and Definitions

Term	Definition
Shortest Path Bridging (SPB)	Shortest Path Bridging (SPB), defined in IEEE 802.1aq, is a protocol that provides data traffic a shortest cost path between any pair of switches in the SPB network, and features dynamic route calculation in a loop-free Layer-2 network, and fast convergence time using IS-IS.
Helper router	An SPB switch node that ingresses customer VLANs, is configured for VRRP, contains route table entries confined to connected VLAN interfaces, and is able to forward all packets with destinations external to the connected customer VLANs to the Main Router.
Main Router	A standard L3 router within the SPB network that is configured for VRRP and used to forward all packets external to the directly connected Helper Router VLANs.
Routing as a Service (RaaS)	A L2 feature using SPBV that provides for scalable and efficient virtualized routing over any L2 network infrastructure, by scaling the fabric from a single chassis to a collection of devices that use L2 protocols to form its topology.
Main router ID	A router ID in an IPv4 format that identifies the router as a Main router in the RaaS network.

24

VLAN Configuration

This chapter provides the following information about configuring and monitoring 802.1Q VLANs on Extreme Networks S-Series devices.

Note: This document describes the configuration and operation of VLANs as defined by the IEEE 802.1Q standard and assumes that all devices being configured support that standard. No other types of VLANs will be covered.

Using VLANs in Your Network

A VLAN is a Virtual Local Area Network — a grouping of network devices that is logically segmented by functions, project teams, or applications without regard to the physical location of users. For example, several end stations might be grouped as a department, such as Engineering or Finance, having the same attributes as a LAN, even though they are not all on the same physical LAN segment.

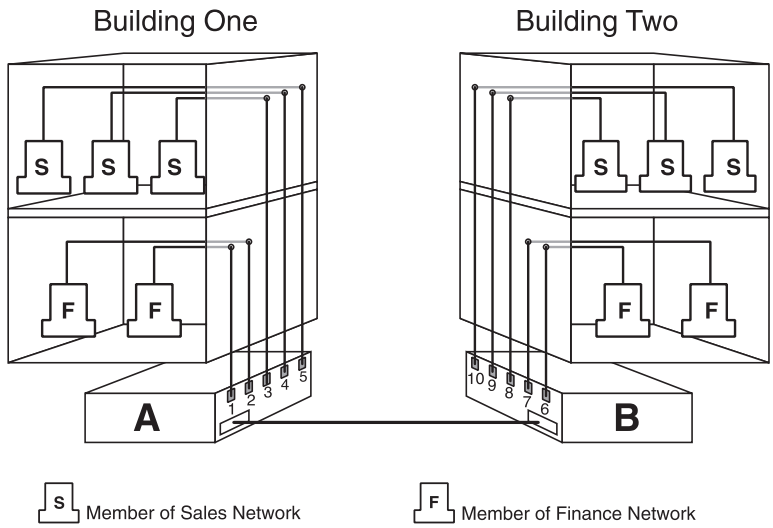
To accomplish this logical grouping, the network administrator uses 802.1Q VLAN-capable switching devices and assigns each switch port in a particular group to a VLAN. Ports in a VLAN share broadcast traffic and belong to the same broadcast domain. Broadcast traffic in one VLAN is not transmitted outside that VLAN.

Virtual LANs allow you to partition network traffic into logical groups and control the flow of that traffic through the network. Once the traffic and, in effect, the users creating the traffic, are assigned to a VLAN, then broadcast and multicast traffic is contained within the VLAN and users can be allowed or denied access to any of the network's resources. Also, you have the option of configuring some or all of the ports on a device to allow frames received with a particular VLAN ID and protocol to be transmitted on a limited number of ports. This keeps the traffic associated with a particular VLAN and protocol isolated from the other parts of the network.

The primary benefit of 802.1Q VLAN technology is that it allows you to localize and segregate traffic, improving your administrative efficiency, and enhancing your network security and performance.

Figure 24-1 shows a simple example of using port-based VLANs to achieve these benefits. In this example, two buildings house the Sales and Finance departments of a single company, and each building has its own internal network. The end stations in each building connect to a switch on the bottom floor. The two switches are connected to one another with a high speed link.

Figure 24-1 VLAN Business Scenario



Without any VLANs configured, the entire network in the example in Figure 24-1 would be a broadcast domain, and the switches would follow the IEEE 802.1D bridging specification to send data between stations. A broadcast or multicast transmission from a Sales workstation in Building One would propagate to all the switch ports on Switch A, cross the high speed link to Switch B, and then be propagated out all switch ports on Switch B. The switches treat each port as being equivalent to any other port, and have no understanding of the departmental memberships of each workstation.

Once Sales and Finance are placed on two separate VLANs, each switch understands that certain individual ports or frames are members of separate workgroups. In this environment, a broadcast or multicast data transmission from one of the Sales stations in Building One would reach Switch A, be sent to the ports connected to other local members of the Sales VLAN, cross the high speed link to Switch B, and then be sent to any other ports and workstations on Switch B that are members of the Sales VLAN. Separate VLANs also provides unicast separation between Sales and Finance. Finance can not ping Sales unless there is a routed VLAN configured for both Finance and Sales.

Another benefit to VLAN use in the preceding example would be your ability to leverage existing investments in time and equipment during company reorganization. If, for instance, the Finance users change location but remain in the same VLAN connected to the same switch port, their network addresses do not change, and switch and router configuration is left intact.

Implementing VLANs

By default, all Extreme Networks switches run in 802.1Q VLAN operational mode. All ports on all Extreme Networks switches are assigned to a default VLAN (VLAN ID 1), which is enabled to operate and assigns all ports an egress status of untagged. This means that all ports will be allowed to transmit frames from the switch without a VLAN tag in their header. Also, there are no forbidden ports (prevented from transmitting frames) configured.

You can use the CLI commands described in this document to create additional VLANs, to customize VLANs to support your organizational requirements, and to monitor VLAN configuration.

Preparing for VLAN Configuration

A little forethought and planning is essential to a successful VLAN implementation. Before attempting to configure a single device for VLAN operation, consider the following:

- What is the purpose of my VLAN design? (For example: security or traffic broadcast containment).
- How many VLANs will be required?
- What stations (end users, servers, etc.) will belong to them?
- What ports on the switch are connected to those stations?
- What ports will be configured as dynamic VLAN GARP VLAN Registration Protocol (GVRP) or Multiple VLAN Registration Protocol (MVRP) aware ports?
- What VLANs will not perform dynamic VLAN GVRP or MVRP processing?

Determining how you want information to flow and how your network resources can be best used to accomplish this will help you customize the tasks described in this document to suit your needs and infrastructure.

Once your planning is complete, you would proceed through the steps described in "Configuring VLANs" on page 24-9.

Understanding How VLANs Operate

802.1Q VLAN operation differs slightly from how a switched networking system operates. These differences are due to the importance of keeping track of each frame and its VLAN association as it passes from switch to switch, or from port to port within a switch.

VLAN-enabled switches act on how frames are classified into a particular VLAN. Sometimes, VLAN classification is based on tags in the headers of data frames. These VLAN tags are added to data frames by the switch as the frames are transmitted out certain ports, and are later used to make forwarding decisions by the switch and other VLAN aware switches. In the absence of a VLAN tag header, the classification of a frame into a particular VLAN depends upon the configuration of the switch port that received the frame.

Learning Modes and Filtering Databases

Addressing information the switch learns about a VLAN is stored in the filtering database assigned to that VLAN. This database contains source addresses, their source ports, and VLAN IDs, and is referred to when a switch makes a decision as to where to forward a VLAN tagged frame. Each filtering database is assigned a Filtering Database ID (FID). The FID a VLAN belongs to can be displayed using the **show vlan** command.

A switch learns and uses VLAN addressing information by the following modes:

- **Independent Virtual Local Area Network (VLAN) Learning (IVL)**: Each VLAN uses its own filtering database. Transparent source address learning performed as a result of incoming VLAN traffic is not made available to any other VLAN for forwarding purposes. This setting is useful for handling devices (such as servers) with NICs that share a common MAC address. One FID is assigned per VLAN. The FID value is the same as the VID it is assigned to. This is the default mode on Extreme Networks switches.
- **Shared Virtual Local Area Network (VLAN) Learning (SVL)**: Two or more VLANs are grouped to share common source address information. This setting is useful for configuring more complex VLAN traffic patterns, without forcing the switch to flood the unicast traffic in each direction. This allows VLANs to share addressing information. It enables ports or switches in different VLANs to communicate with each other (when their individual ports are configured to allow this to occur). One FID is used by two or more VLANs. The FID value defaults to the lowest VID in the filtering database.

The VLAN learning mode for the switch and the assignment of multiple VLANs to a FID are configured in VLAN constraints using the **set vlan constraint** command. See "Appendix F" of the *IEEE Std 802.1Q™2011* standard for a detailed discussion of shared and independent VLAN learning modes.

VLAN Assignment and Forwarding

Receiving Frames from VLAN Ports

By default, Extreme Networks switches run in 802.1Q operational mode, which means that every frame received by the switch must belong to, or be assigned to, a VLAN. The type of frame under consideration and the filter setting of the switch determines how it forwards VLAN frames. This involves processing traffic as it enters (ingresses) and exits (egresses) the VLAN switch ports as described below.

Untagged Frames

When, for example, the switch receives a frame from Port 1 and determines the frame does not currently have a VLAN tag, but recognizes that Port 1 is a member of VLAN A, it will classify the frame to VLAN A. In this fashion, all untagged frames entering a VLAN switch assume membership in a VLAN.

Note: A VLAN ID is always assigned to a port. By default, it is the default VLAN (VLAN ID = 1).

The switch will now decide what to do with the frame, as described in "Forwarding Decisions" on page 24-5.

Tagged Frames

When, for example, the switch receives a tagged frame from Port 4 and determines the frame is tagged for VLAN C, it will classify it to that VLAN regardless of its port VLAN ID (PVID). This frame may have already been through a VLAN aware switch, or originated from a station capable of specifying a VLAN membership. If a switch receives a frame containing a tag, the switch will classify the frame in regard to its tag rather than the PVID for its port, following the ingress precedence rules listed below.

Ingress Precedence

VLAN assignment for received (ingress) frames is determined by the following precedence:

1. 802.1Q VLAN tag (tagged frames only)
2. Policy or Traffic Classification (which may overwrite the 802.1Q VLAN tag) For more information, refer to "Configuring Protocol-Based VLAN Classification" on page 24-15.
3. Port VID (PVID)

Forwarding Decisions

VLAN forwarding decisions for transmitting frames is determined by whether or not the traffic being classified is or is not in the VLAN's forwarding database as follows:

- **Unlearned traffic**: When a frame's destination MAC address is not in the VLAN's forwarding database (FDB), it will be forwarded out of every port on the VLAN's egress list with the frame format that is specified. Refer to"Broadcasts, Multicasts, and Unlearned Unicasts"below for an example.
- **Learned traffic**: When a frame's destination MAC address is in the VLAN's forwarding database, it will be forwarded out of the learned port with the frame format that is specified. Refer to "Learned Unicasts" below for an example.

Broadcasts, Multicasts, and Unlearned Unicasts

If a frame with a broadcast, multicast, or other unknown address is received by an 802.1Q VLAN aware switch, the switch checks the VLAN classification of the frame. The switch then forwards the frame out all ports that are identified in the Forwarding List for that VLAN. For example, if Port 3, shown in the example in Figure 24-2, received the frame, the frame would then be sent to all ports that had VLAN C in their Port VLAN List.

Learned Unicasts

When a VLAN switch receives a frame with a known MAC address as its destination address, the action taken by the switch to determine how the frame is transmitted depends on the VLAN, the VLAN associated FID, and if the port identified to send the frame is enabled to do so.

When a frame is received it is classified into a VLAN. The destination address is looked up in the FID associated with the VLAN. If a match is found, it is forwarded out the port identified in the lookup if, and only if, that port is allowed to transmit frames for that VLAN. If a match is not found, then the frame is flooded out all ports that are allowed to transmit frames belonging to that VLAN.

Adding a MIB-II Interface Entry to a VLAN

A VTAP interface provides the data source input of a port mirror or SMON statistics collection. VTAP creation is the mechanism for adding a MIB-II interface table entry for a VLAN. When creating a VTAP interface, the specified VLAN is assigned a MIB-II ifIndex. A VLAN will not have a MIB-II ifIndex if a VTAP interface does not exist for it. Use the set vlan interface command to create a VTAP interface.

This example shows how to create a non-volatile MIB-II interface entry mapped to VLAN 1:

```
S Chassis(rw)->set vlan interface 1 create
S Chassis(rw)->show vlan interface 1
VLAN MIB-II Interfaces

Max Interfaces     : 16
Current Interfaces : 1

VLAN       Port          Storage Type
```

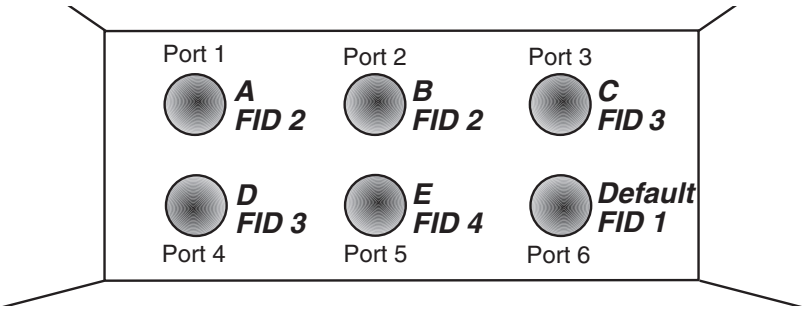
```
-------------------------------------
  1      vtap.0.1         non-volatile
S Chassis(rw)->
```

Example of a VLAN Switch in Operation

The operation of an 802.1Q VLAN switch is best understood from a point of view of the switch itself. To illustrate this concept, the examples that follow view the switch operations from *inside* the switch.

Figure 24-2 depicts the inside of a switch with six ports, numbered 1 through 6. The switch has been configured to associate VLAN A and B with FID 2, VLAN C and D with FID 3, and VLAN E with FID 4. It shows how a forwarding decision is made by comparing a frame's destination MAC to the FID to which it is classified.

Figure 24-2 Inside the Switch



Assume a unicast untagged frame is received on Port 3 in the example in Figure 24-2. The frame is classified for VLAN C (the frame's PVID is VLAN C). The switch would make its forwarding decision by comparing the destination MAC address to information previously learned and entered into its filtering database. In this case, the MAC address is looked up in the FDB for FID 3, which is associated with VLANs C and D. Let's say the switch recognizes the destination MAC of the frame as being located out Port 4.

Having made the forwarding decision based on entries in the FID, the switch now examines the port VLAN egress list of Port 4 to determine if it is allowed to transmit frames belonging to VLAN C. If so, the frame is transmitted out Port 4. If Port 4 has not been configured to transmit frames belonging to VLAN C, the frame is discarded.

If, on the other hand, a unicast untagged frame is received on Port 5, it would be classified for VLAN E. Port 5 has is own filtering database and is not aware of what addressing information has been learned by other VLANs. Port 5 looks up the destination MAC address in its FID. If it finds a match, it forwards the frame out the appropriate port, if and only if, that port is allowed to transmit frames for VLAN E. If a match is not found, the frame is flooded out all ports that are allowed to transmit VLAN E frames.

VLAN Support on Extreme Networks S-Series Switches

For information about...	Refer to page...
Dynamic VLAN Support	24-8

Maximum Active VLANs

The total number of active VLANs supported on Extreme Networks S-Series switches is up to 4094.

Configurable Range

The allowable user-configurable range for VLAN IDs (VIDs) on Extreme Networks S-Series switches is from 2 through 4094. This range is based on the following rules:

- **VID 0** is the null VLAN ID, indicating that the tag header in the frame contains priority information rather than a VLAN identifier. It cannot be configured as a port VLAN ID (PVID).
- **VID 1** is designated the default PVID value for classifying frames on ingress through a switched port. This default can be changed on a per-port basis.
- **VID 4095** is reserved by IEEE for implementation use.

Notes: Each VLAN ID in a network must be unique. If you enter a duplicate VLAN ID, the Extreme Networks switch assumes you intend to modify the existing VLAN.

VLAN Types

Extreme Networks switches support traffic classification for the following VLAN types:

Static and Dynamic VLANs

All VLANs on an Extreme Networks switch are categorized as being either static or dynamic. Static VLANs are those that are explicitly created on the switch itself, persistently remaining as part of the configuration, regardless of actual usage. Dynamic VLANs, on the other hand, are not necessarily persistent. Their presence relies on the implementation of GVRP or MVRP and its effect on egress membership as described in “Dynamic VLAN Support” on page 24-8.

Port-Based VLANs

Port-based VLANs are configured by associating switch ports to VLANs in two ways: first, by manipulating the port VLAN ID (PVID); and second, by adding the port itself to the egress list of the VLAN corresponding to the PVID. Any traffic received by a port is associated to the VLAN identified by the port's PVID. By virtue of this association, this traffic may egress the switch only on those ports listed on the VLAN's egress list. For example, given a VLAN named “Marketing,” with an ID value of 6, by changing the PVID values of ports 1 through 3 to 6, and adding those ports to the egress list of the VLAN, we effectively restrict the broadcast domain of Marketing to those three ports. If a broadcast frame is received on port 1, it will be transmitted out ports 2 and 3 only. In this sense, VLAN membership is determined by the location of traffic ingress, and from the perspective of the access layer—where users are most commonly located—egress is generally untagged.

Policy-Based VLANs

Rather than making VLAN membership decisions simply based on port configuration, each incoming frame can be examined by the classification engine which uses a match-based logic to

assign the frame to a desired VLAN. For example, you could set up a policy which designates all e-mail traffic between the management officers of a company to a specific VLAN so that this traffic is restricted to certain portions of the network. With respect to network usage, the administrative advantages of policy classification would be application provisioning, acceptable use policy, and distribution layer policy. All of these provisions may involve simultaneous utilization of inter-switch links by multiple VLANs, requiring particular attention to tagged, forbidden, and untagged egress settings.

As described above, PVID determines the VLAN to which all untagged frames received on associated ports will be classified. Policy classification to a VLAN takes precedence over PVID assignment if:

- policy classification is configured to a VLAN, and
- PVID override has been enabled for a policy profile, and assigned to port(s) associated with the PVID.

For more information, refer to the Policy Classification chapter.

Dynamic VLAN Support

Dynamic VLAN support automatically creates VLANs across a switched network by dynamically establishing and updating a device's knowledge of the set of VLANs that currently have active members. Two dynamic VLAN capable protocols are supported on the S-Series:

- GARP (Generic Attribute Registration Protocol) VLAN Registration Protocol (GVRP)
- Multiple VLAN Registration Protocol (MVRP)

MVRP is a replacement protocol for GVRP based upon the Multiple Registration Protocol (MRP) defined by the IEEE 802.1ak amendment to the IEEE 802.1Q standard. MVRP uses a single packet event propagation for all 4094 VLANs, providing a more efficient encoding of PDUs which reduces the amount of traffic generated by the protocol. MVRP also performs filter database flushing on a per-VID basis during a topology change.

In a GVRP context, Spanning Tree performs filter database flushing on a per-port basis during a topology change. MVRP's ability to localize topology changes result in a more rapid healing of network failures without interrupting services to unaffected VLANs.

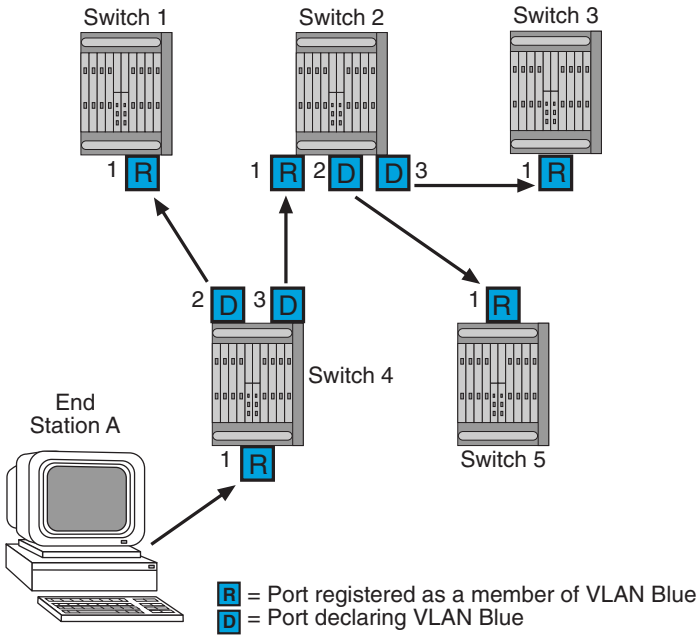
By default, both GVRP and MVRP are enabled both globally and on all VLANs, and disabled at the port level. For dynamic VLAN creation to occur, either MVRP or GVRP must be enabled globally as well as on each individual port. GVRP and MVRP can not both be enabled on the same port. Dynamic VLAN creation can be configured on both a per port and per VLAN basis.

How Dynamic VLAN Support Works

When a VLAN has egress, the information is transmitted out ports on the device in a GVRP or MVRP formatted frame, using the GVRP or MVRP multicast MAC address. A switch that receives this frame examines the frame and extracts the VLAN IDs. The dynamic VLAN protocol then dynamically registers (creates) the VLANs and adds the receiving port to its tagged member list for the extracted VLAN IDs. The information is then transmitted out the other GVRP or MVRP configured ports of the device.

Figure 24-3 shows an example of how VLAN Blue from end station A would be propagated across a switch network. In this figure, port 1 of Switch 4 is registered as being a member of VLAN Blue and Switch 4 declares this fact out all its ports (2 and 3) to Switch 1 and Switch 2. These two switches register this in the port egress lists of the ports (Switch 1, port 1 and Switch 2, port 1) that received the frames with the information. Switch 2, which is connected to Switch 3 and Switch 5 declares the same information to those two switches and the port egress list of each port is updated with the new information, accordingly.

Figure 24-3 Example of VLAN Propagation Using GVRP or MVRP



Note: If a port is set to "forbidden" for the egress list of a VLAN, then the VLAN's egress list will not be dynamically updated with that port.

Administratively configuring a VLAN on an 802.1Q switch creates a static VLAN entry that will always remain registered and will not time out. However, GVRP or MVRP dynamically created entries will time out, and their registrations will be removed from the member list if the end station is removed. This ensures that, if switches are disconnected or if end stations are removed, the registered information remains accurate.

The end result of the dynamic VLAN configuration is that each port's egress list is updated with information about VLANs that reside on that port, even if the actual station on the VLAN is several hops away.

Configuring VLANs

Once you have planned your implementation strategy as described in"Preparing for VLAN Configuration" on page 24-3, you can begin configuring VLANs as described in this section.

Default Settings

Table 24-1 lists VLAN parameters and their default values.

Table 24-1 Default VLAN Parameters

Parameter	Description	Default Value
garp or mrp timer	Configures the three GARP or MRP timers. These timers are critical and should only be modified by someone familiar with the 802.1Q standard.	• Join timer: 20 centiseconds • Leave timer: 60 centiseconds • Leaveall timer: 1000 centiseconds
gvrp	Enables or disables the GARP VLAN Registration Protocol (GVRP) on a specific set of ports or all ports for dynamic VLAN creation. GVRP must be enabled when using GVRP for dynamic VLAN creation. GVRP can not be enabled on ports MVRP is enabled on.	• Disabled at the port level • Enabled at the global level
mvrp	Enables or disables the Multiple VLAN Registration Protocol (MVRP) on a specific set of ports or all ports. MVRP must be enabled when using MVRP for dynamic VLAN creation. MVRP can not be enabled on ports GVRP is enabled on.	• Disabled at the port level • Enabled at the global level
vlan restricted	Disables dynamic VLAN creation on a VLAN basis when VLAN restricted is enabled on the VLAN.	Dynamic VLAN is enabled on all VLANs. VLAN restricted feature is disabled.
IGMP last member query interval	Configures the last member query interval. This is the maximum response time inserted into group-specific queries which are sent in response to Leave Group messages. It is also the amount of time between group-specific query messages.	1 second
IGMP VLAN max response time	Configures the maximum query response time (in tenths of a second).	100 deciseconds (10 seconds)
IGMP VLAN query interval	Configures the frequency (in seconds) of host-query frame transmissions.	125 seconds
IGMP VLAN robustness	Configures the robustness value.	2
IGMP VLAN version	Selects the IGMP version. Options are version 1 or version 2.	Version 2
port discard	Ports can be set to discard frames based on whether or not they contain a VLAN tag.	No frames are discarded
port ingress filter	When enabled on a port, the VLAN IDs of incoming frames are compared to the port's egress list. If the received VLAN ID does not match a VLAN ID on the port's egress list, the frame is dropped.	Disabled

Table 24-1 Default VLAN Parameters (continued)

Parameter	Description	Default Value
port vlan ID (PVID)	802.1Q VLAN/port association.	VLAN1/ Default VLAN
vlan constraint	Configures VLANs to use an independent or shared filtering database.	VLANs use an independent filtering database
vlan dynamicegress	Enables or disables dynamic egress processing for a given VLAN.	Disabled
vlan egress	Configures the egress ports for a VLAN and the type of egress for the ports. Egress type can be tagged, untagged, or forbidden.	Tagged
vlan name	Associates a text name to one or more VLANs.	None

Configuring Static VLANs

Procedure 24-1 describes how to create and configure a static VLAN. Unspecified parameters use their default values.

Procedure 24-1 Static VLAN Configuration

Step	Task	Command(s)
1.	Show existing VLANs.	**show vlan**
2.	Create VLAN. Valid values are **1**–**4094**. Each *vlan-id* must be unique. If an existing *vlan-id* is entered, the existing VLAN is modified.	**set vlan create** *vlan-id*
3.	Optionally, assign a name to the VLAN. Valid strings are from 1 to 32 characters.	**set vlan name** *vlan-id string*
4.	Assign switched ports to the VLAN. This sets the port VLAN ID (PVID). The PVID determines the VLAN to which all untagged frames received on the port will be classified.	**set port vlan** *port-string vlan-id*
	Note: If the VLAN specified has not already been created, the above command will create it. It will also add the VLAN to the port's egress list as untagged, and remove the default VLAN from the port's egress list. This automatically changes the existing untagged VLAN egress permission to match the new PVID value.	

Procedure 24-1 Static VLAN Configuration (continued)

Step	Task	Command(s)
5.	Configure VLAN egress, which determines which ports a frame belonging to the VLAN may be forwarded out on.	
	Static configuration: Add the port to the VLAN egress list for the device. • The default setting, **tagged**, allows the port to transmit frames for a particular VLAN. • The **untagged** setting allows the port to transmit frames without a VLAN tag. This setting is usually used to configure a port connected to an end user device. • The **forbidden** setting prevents the port from participating in the specified VLAN and ensures that any dynamic requests for the port to join the VLAN will be ignored.	**set vlan egress** *vlan-id port-string* **forbidden** | **tagged** | **untagged**
	If necessary, remove ports from the VLAN egress list. • If specified, the **forbidden** setting will be cleared from the designated ports and the ports will be reset as allowed to egress frames, if so configured by either static or dynamic means. • If **forbidden** is not specified, tagged and untagged egress settings will be cleared from the designated ports.	**clear vlan egress** *vlan-list port-string* [**forbidden**]
	Dynamic configuration: By default, dynamic egress is disabled on all VLANs. If dynamic egress is enabled for a VLAN, the device will add the port receiving a frame to the VLAN's egress list as untagged according to the VLAN ID of the received frame.	**set vlan dynamicegress** *vlan-id* {**enable** | **disable**}
6.	Optionally, set VLAN constraints to control the filtering database a VLAN will use for forwarding traffic. Filtering databases can be shared or independent. By default, filtering databases are independent.	**set vlan constraint** *vlan-id set-num* [**shared** | **independent**]
7.	Optionally, enable ingress filtering on a port to drop those incoming frames that do not have a VLAN ID that matches a VLAN ID on the port's egress list.	**set port ingress-filter** *port-string* **enable**
8.	Optionally, choose to discard tagged or untagged, (or both) frames on selected ports. Select **none** to allow all frames to pass through.	**set port discard** *port-string* {**tagged** | **untagged** | **none | both**}
9.	If the device supports routing, enter interface configuration mode and configure an IP address on the VLAN interface.	**configure** **interface vlan** *vlan-id* **ip address** *ip-address ip-mask* **no shutdown**

Procedure 24-1 Static VLAN Configuration (continued)

Step	Task	Command(s)
	Note: Each VLAN interface must be configured for routing separately using the interface command shown above. To end configuration on one interface before configuring another, type **exit** at the command prompt. Enabling interface configuration mode is required for completing interface-specific configuration tasks.	

Example Configuration

The following shows an example S-Series device configuration using the steps in Procedure 24-1. In this example, VLAN 100 is created and named VLANRED. Ports ge.1.2, 1.3 and 1.4 are assigned to VLAN 100 and added to its egress list. VLAN 100 is then configured as a routing interface with an IP address of 120.20.20.24.

```
S Chassis(rw)->set vlan create 100
S Chassis(rw)->set vlan name 100 VLANRED
S Chassis(rw)->set port vlan ge.1.2-4 100
    The PVID is used to classify untagged frames as they
    ingress into a given port.  Would you like to add the selected
    port(s) to this VLAN's untagged egress list and remove them
    from all other VLANs untagged egress list (y/n)  [n]? y
    NOTE:  Choosing 'y' will not remove the port(s) from previously
    configured tagged egress lists.
S Chassis(rw)->router
S Chassis(rw)-router)->configure terminal
S Chassis(rw)-router-config)->interface vlan 100
S Chassis(rw)-router-config-intf-Vlan-100)->ip address 120.20.20.1/24
S Chassis(rw)-router(config-intf-Vlan-100)->no shutdown
```

If you want to configure a port to drop incoming frames that do not have a VLAN ID that matches a VLAN ID on the port's egress list, use the **set port ingress-filter** command. For example:

```
S Chassis(rw)->set port ingress-filter ge.1.2-4 enable
```

If you want to configure a port to discard tagged or untagged incoming frames, use the **set port discard** command. For example, to configure the ports to drop tagged frames on ingress:

```
S Chassis(rw)->set port discard ge.1.2-4 tagged
```

Creating a Secure Management VLAN

If you are configuring an Extreme Networks device for multiple VLANs, it may be desirable to configure a management-only VLAN. This allows a station connected to the management VLAN to manage the device. It also makes management secure by preventing configuration through ports assigned to other VLANs.

Procedure 24-2 provides an example of how to create a secure management VLAN. This example, which sets the new VLAN as VLAN 2, assumes the management station is attached to ge.1.1, and wants untagged frames.The process described in this section would be repeated on every switch device that is connected in the network to ensure that each switch has a secure management VLAN.

Procedure 24-2 Secure Management VLAN Configuration

Step	Task	Command(s)
1.	Create a new VLAN.	**set vlan create 2**
2.	Set the PVID for the host port and the desired switch port to the VLAN created in Step 2.	**set port vlan host.0.1; ge.1.1 2**
3.	If not done automatically when executing the previous command, add the host port and desired switch port(s) to the new VLAN's egress list.	**set vlan egress 2 host.0.1; ge.1.1 2 untagged**
4.	Set a private community name to assign to this VLAN for which you can configure access rights and policies.	**set snmp community private**
	Note: By default, community name—which determines remote access for SNMP management—is set to **public** with read-write access. For more information, refer to your device's SNMP documentation.	

Configuring Dynamic VLANs

Procedure 24-3 describes the configuration of dynamic VLANs using the GARP (Generic Attribute Registration Protocol) VLAN Registration Protocol (GVRP). Procedure 24-4 on page 24-15 describes the configuration of dynamic VLANs using the Multiple VLAN Registration Protocol (MVRP). Both GVRP and MVRP are enabled globally and disabled at the port level. A port can only be enabled for GVRP or MVRP, but not both. Whichever dynamic VLAN protocol you are using, it must be globally enabled and also enabled on specific ports in order to generate and process the appropriate GVRP or MVRP advertisement frames. The dynamic VLAN protocol must be enabled on the VLAN for the appropriate advertisement frames to be processed. Dynamic VLAN creation can be disabled per VLAN using the VLAN restriction feature supported by both GVRP and MVRP.

Note: Refer to "Dynamic VLAN Support" on page 24-8 for conceptual information about both GVRP and MVRP.

Caution: The setting of GARP or MRP timers is critical and should only be changed by personnel familiar with 802.1Q standards.

Procedure 24-3 GVRP Configuration

Step	Task	Command(s)
1.	Show existing GVRP configuration for a port or list of ports. If no *port-string* is entered, the global GVRP configuration and all port GVRP configurations are displayed.	**show gvrp** [*port-string*]
2.	Enable GVRP on those ports assigned to a VLAN. GVRP is disabled at the port level by default.	**set gvrp enable** *port-string*
3.	Optionally, disable GVRP processing on specified VLANs.	**set gvrp vlan** *vlan-list* **restricted disable**
4.	Display GVRP status for system VLANs.	**show gvrp vlan** {*vlan-list* | **all**} **restricted**
5.	Display the existing GARP timer values.	**show garp timer** [*port-string*]

Procedure 24-3 GVRP Configuration (continued)

Step	Task	Command(s)
6.	Optionally, set the GARP join, leave, and leaveall timer values. Each timer value is in centiseconds.	**set garp timer** {[**join** *timer-value*] [**leave** *timer-value*] [**leaveall** *timer-value*]} *port-string*

Procedure 24-4 MVRP Configuration

Step	Task	Command(s)
1.	Show existing MVRP configuration for a port or list of ports. If no *port-string* is entered, the global MVRP configuration and all port MVRP configurations are displayed.	**show mvrp** [*port-string*]
2.	Enable MVRP on those ports assigned to a VLAN. MVRP is disabled at the port level by default.	**set mvrp enable** *port-string*
3.	Optionally, disable MVRP processing on specified VLANs.	**set mvrp vlan** *vlan-list* **restricted disable**
4.	Display MVRP status for system VLANs.	**show mvrp vlan** {*vlan-list* | **all**} **restricted**
5.	Display the existing MRP timer values.	**show mrp timer** [*port-string*]
6.	Optionally, set the MRP join, leave, and leaveall timer values or enable or disable the periodic timer which has a fixed interval of 1 second. Each timer value is in centiseconds.	**set mrp timer** {[**join** *timer-value*] [**leave** *timer-value*] [**leaveall** *timer-value*] [**periodic** {**enable** | **disable**} *port-string*

Configuring Protocol-Based VLAN Classification

Protocol-based VLANs can be configured using the policy classification CLI commands, as shown in this section, or NetSight Policy Manager.

Procedure 24-5 describes how to define protocol-based frame filtering policies to assign frames to particular VLANs. Refer to your Extreme Networks policy configuration and CLI documentation for more information.

Note: Depending on your Extreme Networks switching device, your options for configuring policy classification may differ from the examples provided in this section. Refer to your device's documentation for a list of CLI commands and functions supported.

Procedure 24-5 Configuring Protocol-Based VLAN Classification

Step	Task	Command(s)
1.	Create the VLANs to which frames will be assigned by the policy. Valid values are **1**–**4094**.	**set vlan create** *vlan-id*
2.	Configure VLAN egress, which determines which ports a frame belonging to the VLAN may be forwarded out on. The default setting, **tagged**, allows the port to transmit frames for a particular VLAN.	**set vlan egress** *vlan-id port-string* [**forbidden** | **tagged** | **untagged**]

Procedure 24-5 Configuring Protocol-Based VLAN Classification (continued)

Step	Task	Command(s)
3.	Disable ingress filtering on the ingress ports on which the policy will be applied. Disabled is the default ingress filtering setting.	**set port ingress-filter** *port-string* **disable**
4.	Create the policy profile that enables PVID override. This function allows a policy rule classifying a frame to a VLAN to override PVID assignment configured with the **set port vlan** command. When none of its associated classification rules match, the configuration of the policy profile itself will determine how frames are handled by default. In this case, the default VLAN is specified with the **pvid** *pvid* parameter.	**set policy profile** *profile-index* [**name** *name*] [**pvid-status** {**enable** | **disable**}] [**pvid** *pvid*]
5.	Configure the administrative rules that will assign the policy profile to all frames received on the desired ingress ports.	**set policy rule admin-profile port** *port-string* [**port-string** *port-string*] [**admin-pid** *admin-pid*]
6.	Configure the classification rules that will define the protocol to filter on and the VLAN ID to which matching frames will be assigned.	**set policy rule** *profile-index* {*protocol data* [**mask** *mask*]} [**vlan** *vlan*]

Example Configuration

The following shows an example S-Series device configuration using the steps in Procedure 24-5. This example configures a policy that ensures that IP traffic received on the specified ingress ports will be mapped to VLAN 2, while all other types of traffic will be mapped to VLAN 3.

1. Two VLANs are created: VLAN 2 and VLAN 3.
2. Ports 1 through 5 on the Gigabit Ethernet IOM in slot 4 are configured as egress ports for the VLANs while ports 8 through 10 on the Gigabit Ethernet IOM in slot 5 are configured as ingress ports that will do the policy classification.
3. Policy profile number 1 is created that enables PVID override and defines the default behavior (classify to VLAN 3) if none of the classification rules created for the profile are matched.
4. Administrative rules are created that apply policy profile number 1 to all frames received on the ingress ports ge.5.8 through 10.
5. Classification rules are created for policy profile number 1 that assign IP frames to VLAN 2. The rules identify IP frames by using the **ether** protocol parameter, which classifies on the Type field in the headers of Layer 2 Ethernet II frames, and the protocol data of 0x0800 (IP type), 0x0806 (ARP type), and 0x8035 (RARP type).

```
S Chassis(rw)->set vlan create 2,3
S Chassis(rw)->set vlan egress 2 ge.4.1-2
S Chassis(rw)->set vlan egress 3 ge.4.3-5
S Chassis(rw)->set port ingress-filter ge.5.8-10 disable
S Chassis(rw)->set policy profile 1 name protocol_based_vlan pvid-status enable
   pvid 3
S Chassis(rw)->set policy rule admin-profile port ge.5.8 port-string ge.5.8
   admin-pid 1
S Chassis(rw)->set policy rule admin-profile port ge.5.9 port-string ge.5.9
   admin-pid 1
```

```
S Chassis(rw)->set policy rule admin-profile port ge.5.10 port-string ge.5.10
   admin-pid 1
S Chassis(rw)->set policy rule 1 ether 0x0800 mask 16 vlan 2
S Chassis(rw)->set policy rule 1 ether 0x0806 mask 16 vlan 2
S Chassis(rw)->set policy rule 1 ether 0x8035 mask 16 vlan 2
```

Configuring IGMP VLAN Snooping

IGMP Layer 2 snooping allows the Extreme Networks switch for a specific VLAN to actively participate in IGMP traffic forwarding. IGMP snooping depends on the presence of an upstream IGMP querier. Whenever it receives an IGMP query, the switch forwards the query out the appropriate VLAN ports. IGMP snooping allows per-port traffic patterns in VLANs with multiple ports. It is disabled by default.

For more information, refer to the *Extreme Networks S-Series CLI Reference*.

Procedure 24-6 describes how to configure IGMP snooping for a VLAN.

Procedure 24-6 IGMP Snooping for a VLAN Configuration

Step	Task	Command(s)
1.	Enable IGMP snooping for a VLAN or a range of VLANs.	**set igmp enable** *vlan-id*
2.	Enable querying on this VLAN, and specify the IGMP querier source address.	**set igmp query-enable** *vlan-id* **address** *ip-address*
3.	Set the version of IGMP to use. Enter **1** for IGMPv1, **2** for IGMPv2, or **3** for IGMPv3.	**set igmp config** *vlan-id* **igmp-version 1|2|3**
4.	Set the Last Member interval value, which can be 1–255.	**set igmp config** *vlan-id* **last-member-interval** *value*
5.	Set the Max Response Time which can be 1–255 seconds.	**set igmp config** *vlan-id* **max-response-time** *seconds*
6.	Set the Query Interval, which can be 1–65535 seconds.	**set igmp config** *vlan-id* **query-interval** *seconds*
7.	Set the Robustness value, which can be 2–255.	**set igmp config** *vlan-id* **robustness** *value*
8.	Optionally, create a static IGMP entry, or add ports to an existing entry. The entry can be in the form of an IP multicast address or IP group address.	***set*** **igmp** *add-static {IP-multicast-address | IP-group-address vlan-id}* **[modify]** *port-string*

Monitoring VLANs

Table 24-2 describes the **show** commands that display information about VLAN configurations. Refer to *Extreme Networks S-Series CLI Reference* for a description of the output of each **show** command.

Table 24-2 Displaying VLAN Information

Task	Command
Display all existing VLANs.	**show vlan**
Display VLAN information for a port or range of ports.	**show vlan portinfo** [**port** *port-string*] [**vlan** *vlan*]

Table 24-2 Displaying VLAN Information (continued)

Task	Command
Display the VLAN constraint setting.	**show vlan constraint** [*vlan id*]
Display the VLAN dynamic egress setting.	**show vlan dynamicegress** [*vlan id*]
Display all static VLANs.	**show vlan static**
Display ports assigned to VLANs.	**show port vlan** [*port-string*]
Display existing GVRP settings.	**show gvrp** [*port-string*]
Display existing MVRP settings.	**show mvrp** [*port-string*]
Display existing GVRP VLAN restricted settings.	**show gvrp vlan** {*vlan-list* | **all**} **restricted**
Display existing MVRP VLAN restricted settings.	**show mvrp vlan** {*vlan-list* | **all**} **restricted**
Display GARP timer vlaues for one or more ports.	**show garp timer** [*port-string*]
Display MRP timer vlaues for one or more ports.	**show mrp timer** [*port-string*]
Display IGMP VLAN configuration.	**show igmp config** [*vlan id*]
Display IGMP enable state of VLAN.	**show igmp enable** [*vlan id*]
Display all groups on a given VLAN.	**show igmp groups** [*vlan id*]
Display IGMP VLAN query state.	**show igmp query** [*vlan id*]
Display static ports on the given vid, group.	**show igmp static** [*vlan id*]

Terms and Definitions

Table 24-3 lists terms and definitions used in VLAN configuration.

Table 24-3 VLAN Terms and Definitions

Term	Definition
Default VLAN	The VLAN to which all ports are assigned upon initialization. The default VLAN has a VLAN ID of 1 and cannot be deleted or renamed.
Filtering Database	A database structure within the switch that keeps track of the associations between MAC addresses, VLANs, and interface (port) numbers. The Filtering Database is referred to when a switch makes a forwarding decision on a frame.
Filtering Database Identifier (FID)	Addressing information that the device learns about a VLAN is stored in the filtering database assigned to that VLAN. Several VLANs can be assigned to the same FID to allow those VLANs to share addressing information. This enables the devices in the different VLANs to communicate with each other when the individual ports have been configured to allow communication to occur. The configuration is accomplished using the Local Management VLAN Forwarding Configuration screen. By default a VLAN is assigned to the FID that matches its VLAN ID.
Forwarding List	A list of the ports on a particular device that are eligible to transmit frames for a selected VLAN.
GARP Multicast Registration Protocol (GMRP)	A GARP application that functions in a similar fashion as GVRP, except that GMRP registers multicast addresses on ports to control the flooding of multicast frames.

Table 24-3 VLAN Terms and Definitions (continued)

Term	Definition
GARP VLAN Registration Protocol (GVRP)	A GARP application used to dynamically create VLANs across a switched network.
Generic Attribute Registration Protocol (GARP)	GARP is a protocol used to propagate state information throughout a switched network.
Multiple VLAN Registration Protocol (MVRP)	An MRP application used to dynamically create VLANs across a switched network.
Multiple Registration Protocol (MRP)	MRP is a protocol used to propagate state information throughout a switched network.
Port VLAN List	A per port list of all eligible VLANs whose frames can be forwarded out one specific port and the frame format (tagged or untagged) of transmissions for that port. The Port VLAN List specifies what VLANs are associated with a single port for frame transmission purposes.
Tag Header (VLAN Tag)	Four bytes of data inserted in a frame that identifies the VLAN/frame classification. The Tag Header is inserted into the frame directly after the Source MAC address field. Twelve bits of the Tag Header represent the VLAN ID. The remaining bits are other control information.
Tagged Frame	A data frame that contains a Tag Header. A VLAN aware device can add the Tag Header to any frame it transmits.
Untagged Frame	A data frame that does not have a Tag Header.
VLAN ID	A unique number (between 1 and 4094) that identifies a particular VLAN.
VLAN Name	A 32-character alphanumeric name associated with a VLAN ID. The VLAN Name is intended to make user-defined VLANs easier to identify and remember.

VLAN Provider Bridges

By extending VLAN tagging technology via the IEEE 802.1Q-2011 standard, an organization managing a service provider network can provide external user groups (departments, customers) to each have their own separate logical network (a LAN consisting of multiple VLANs) through the provider network to predefined egress ports. The provider can assign a single VLAN through their network for all traffic egressing through a port, instead of having to create and manage a separate VLAN for each customer VLAN. This feature, commonly called Provider Bridges (also VLAN stacking, or Q-in-Q), performs Layer 2 tunneling from one customer network location through the provider network to another customer network location.

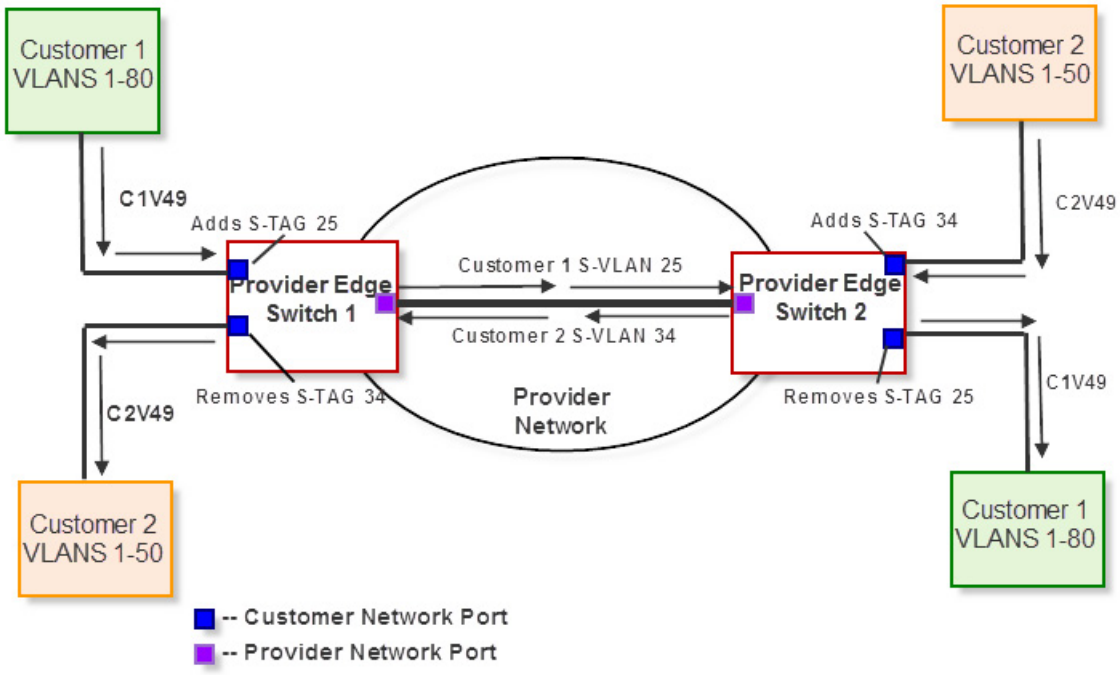
Provider bridges allow a service provider to assign customer traffic to a service instance called an S-VLAN (Service VLAN). When a packet from a customer VLAN arrives on an S-Series switch at the edge of the provider network, it is already identified by a tag called a C-TAG (customer tag). The edge switch encapsulates the packet with another tag called an S-TAG (service provider tag) for relay through the service provider network. The packet egresses from the provider network to the destination customer site through an egress port at which the S-TAG is stripped off. The packet continues on to the remote customer network guided by its original C-VLAN tagging, unaltered.

Table 24-4 Provider Bridge VLAN Tags

TAG Name	Description
C-TAG	Customer VLAN tag. 32 bit tag where the first 16 bits represent the ether type for the customer (0x8100) and the lower 16 bits represent the Priority Code Point (PCP), Canonical Frame Indicator (CFI), and customer VLAN ID (C-VID) associated with the packet.
S-TAG	Service provider VLAN tag. 32 bit tag where the first 16 bits represent the ether type for the service provider (0x88a8) and the lower 16 bits represent the PCP, DE (Drop Eligible), and service provider VLAN ID (S-VID) associated with the packet.

Figure 24-4 illustrates the Provider Bridges function in a provider network with two customers, each with a campus on either side of the provider network and the need to connect their VLANs inexpensively, transparently, and securely. VLAN 49 from Customer 1 ingresses the provider network through the Customer Network Port (CNP) on Provider Edge Switch 1, which adds S-TAG 25 to packets with the C-VID for Customer 1 VLAN 49 (C1V49). After traversing the provider network as S-VLAN 25, the packets egress the provider network at the designated CNP on Provider Edge Switch 2, where the S-TAG 25 is removed and the packets are forwarded to their destination. Even though Customer 2 has a VLAN 49 (C2V49), this traffic is tagged with S-TAG 34 and traverses the provider network as S-VLAN 34.

Figure 24-4 Provider Bridges in Provider Network



If other traffic from Customer 1 (for example, VLAN 22) must traverse the provider network to get to the remote Customer 1 campus, the provider edge switches would add the S-TAG for S-VLAN 25 on C1V22 packets as well. In this way all Customer 1 traffic (80 VLANs) can pass over the provider network using a single S-VLAN, and arrive at their destinations with C-VLAN intact.

Table 24-5 defines the types of ports used in the Provider Bridge architecture. In hardware terms they are identical, but they differ in their roles in the bridging feature architecture.

Table 24-5 Provider Bridge-related Port Types

Port Type	Description
Customer Bridge Port	Customer bridge ports are the existing, default type of ports carrying C-VLAN traffic on provider edge switches. They interpret the C-TAG and relay packets using the C-VID.
Customer Network Port (CNPs)	CNPs are ports resident on provider edge switches that connect the customer network to the provider network. Traffic received on a CNP is assigned to an S-VID based on the ingress settings of the CNP and encapsulated with an S-TAG. CNPs remove the S-TAG from the packets egressing the provider network. In cases where two provider networks are connected via CNPs, packets egress the device with the S-TAG. In these cases the ingress S-VID is translated to a relay S-VID which maps the S-VLANs of one provider network to the S-VLANs of the second provider network. Customer Network Ports are configured as untagged egress ports via the **set vlan egress** command.
Provider Network Port (PNPs)	PNPs interconnect switches within the provider network and operate using S-TAGs and S-VIDs for relay operation. Provider Network Ports are configured as tagged egress ports via the **set vlan egress** command.

Configuring Provider Bridges

There are two possible bridge modes for bridging customer traffic:

- Customer Bridge Mode

  The default mode of routing external C-VLANs via C-TAGs.

- Provider Bridge Mode

  Provider Bridge mode enables tunneling in provider network by adding S-TAGs to C-VLANs and transporting them as dedicated service VLANs.

Customer Bridge Mode

The default mode for bridging and routing of external (customer) VLANs through an Extreme Networks S-Series switch is called Customer Bridging. In a customer bridge, all packets are bridged and routed based on the VLAN identified in the C-TAG. Routing through and egress from the network must be configured on a per-VLAN basis, so there must be a VLAN configured for each VLAN passing through the switch.

Provider Bridge Mode

When the bridge mode is set to provider-bridge, switches in a service provider network relay packets based on service VLANs (using S-TAGs) rather than customer VLANs (using C-TAGs). The VLAN configuration now operates over the S-VLAN for relay operations instead of the C-VLAN. Customer-provided STP and MVRP addresses are preserved in the packets, but are ignored in the provider network. The provider network can be configured to use its own Spanning Tree and MVRP addresses to switch packets around the S-VLAN.

Any customer traffic received on a provider network ingress port will be placed in the S-VLAN defined for that port, and will be delivered to the customer network connected to the provider

network egress port. For bidirectional VLAN traffic between the two remote customer networks, both of these provider ports must be configured on the same service VLAN.

Procedure 24-7 shows an example configuration for transporting any number of received customer VLANs across a provider network in provider bridge mode. This configuration would apply to one of the switches in the provider network. A similar configuration must be executed on each provider switch in the path to the destination customer network. There must be an S-VLAN configured for each egress port to a customer network.

Procedure 24-7 Configuring a Provider Bridge

Step	Task	Command(s)
1.	Set the bridge mode to provider bridge.	**set bridge mode provider-bridge**
2.	Create an S-VLAN in the provider network. In this example, the new provider S-VLAN is 400.	**set vlan create 400**
3.	Add port ge.1.1 and ge.1.2 (both on this switch) to the S-VLAN 400 egress port list. This action sets the ports' VLAN ID to 400.	**set port vlan ge.1.1-2 400 modify-egress**
4.	Specify that port ge.1.1 can transmit frames out to the customer network without (VLAN 400) provider tagging.	**set vlan egress 400 ge.1.1 untagged**
5.	At port ge.1.2, the frames for VLAN 400 are egressed to another switch in the provider network **tagged** with a VLAN 400 S-TAG.	**set vlan egress 400 ge.1.2 tagged**

Figure 24-5 illustrates the following example configuration:

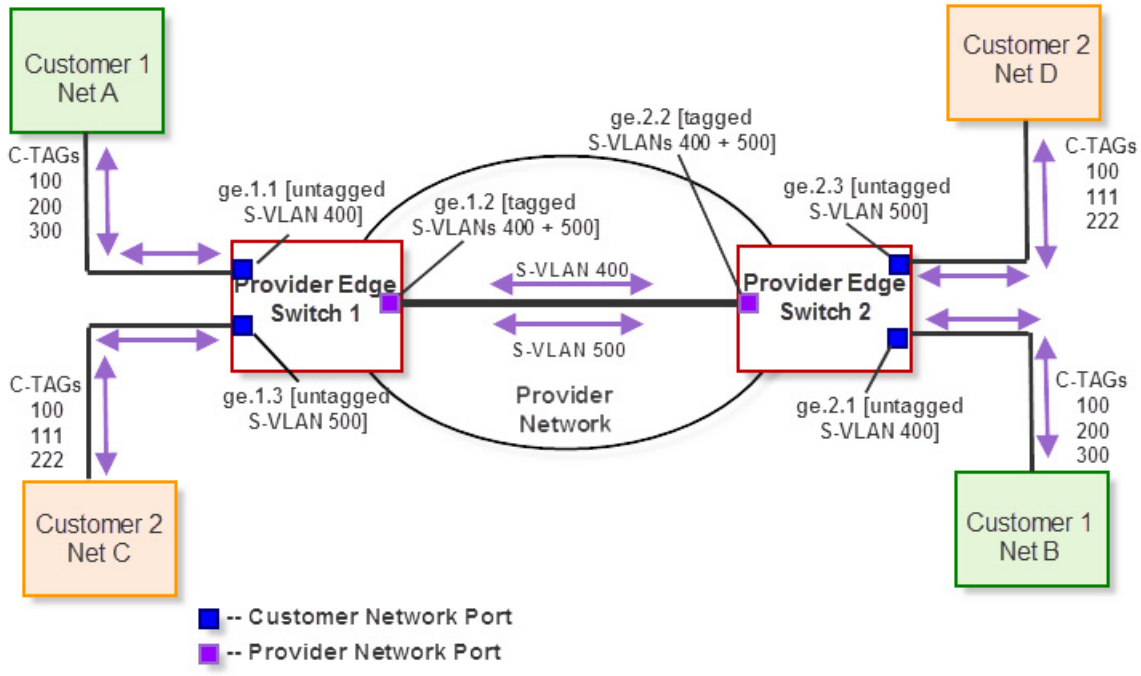
Provider Edge Switch 1:

```
set bridge mode provider-bridge
set vlan create 400
set port vlan ge.1.1-2 400 modify-egress
set vlan egress 400 ge.1.1 untagged
set vlan egress 400 ge.1.2 tagged
set vlan create 500
set port vlan ge.1.2-3 500 modify-egress
set vlan egress 500 ge.1.3 untagged
set vlan egress 500 ge.1.2 tagged
```

Provider Edge Switch 2:

```
set bridge mode provider-bridge
set vlan create 400
set port vlan ge.2.2-3 400 modify-egress
set vlan egress 400 ge.2.1 untagged
set vlan egress 400 ge.2.2 tagged
set vlan create 500
set port vlan ge.2.2-3 500 modify-egress
set vlan egress 500 ge.2.3 untagged
set vlan egress 500 ge.2.2 tagged
```

Figure 24-5 Provider Bridge Configuration Example



To clear provider bridge mode, revert your provider network by setting each bridge back to customer bridge mode. Use the **show bridge mode** command to view the current bridge mode.

25

Link Aggregation Control Protocol (LACP) Configuration

This document describes the link aggregation feature and its configuration on Extreme Networks S-Series devices.

Using Link Aggregation in Your Network

IEEE 802.3ad link aggregation provides a standardized means of grouping multiple parallel Ethernet interfaces into a single logical Layer 2 link. The formed group of Ethernet interfaces is referred to as a Link Aggregation Group (LAG). Dynamic LAG formation and activation is provided by the Link Aggregation Control Protocol (LACP).

Each pair of LAG physical ports is made up of a local port on the device responsible for LACP negotiation, referred to as the actor, and its directly linked remote port on the device participating in the LACP negotiation, referred to as the partner. LAGs form automatically based upon a set of criteria (see "How a LAG Forms" on page 25-3).

Only LAG members in the attached state carry user traffic. Once the LAG is formed, the system ID, made up of a system priority and the device MAC address, determines which device will be in charge of choosing the LAG port members that will be moved to the attached state. While port speed is not a criteria for joining a LAG, the port speed must match for all ports that are placed in the LACP attached state. Aggregatable ports not selected to carry traffic for this LAG are available to the next LAG as long as LAG resources are not depleted. Should LAG resources become depleted, aggregatable ports are placed in LACP standby state.

802.3ad LACP aggregations can be run between combinations of switches, routers, and edge devices, such as a server, that support LACP.

Note: Earlier (proprietary) implementations of port aggregation referred to groups of aggregated ports as "trunks".

The concept of grouping multiple ports into a single link is not a new idea. Cabletron's SmartTrunk, Cisco's Inter Switch Link trunking, and Adaptec's Duralink are previous examples. The problem with these older methods, from the network administrators point of view, is that they are proprietary. Administrators who wanted to implement faster logical links faced major problems if they also wanted, or needed, to use a different brand of networking hardware. Link aggregation is standards based allowing for interoperability between multiple vendors in the network.

Older implementations required manual configuration. With LACP, if a set of links can aggregate, they will aggregate. LACP's ability to automatically aggregate links represents a timesaver for the network administrator who will not be required to manually configure the aggregates. However, manual overrides are provided for when the administrator needs to customize. Link aggregation also provides for rapid configuration and reconfiguration when there are changes in the physical connections. Link aggregation will automatically and quickly converge the new configuration. This convergence typically occurs in one second or less.

Link aggregation is a cost effective way to implement increased bandwidth. A major benefit of link aggregation is the ability to incrementally add bandwidth in a linear fashion. Without link aggregation, if there is a need to increase the bandwidth for a 100Mbps pipe, the only choice is an exponential upgrade to a 1000Mbps pipe. If there is a need for a 300Mbps pipe, aggregating three 100Mbps ports is both less expensive, because a forklift hardware upgrade is avoided, and makes for more efficient use of the system ports that are already available.

The physical links within the aggregate can serve as redundant backups to one another. Since only a single MAC address representing the entire aggregate is presented to the MAC client, the failure of any link within the aggregate is transparent. Failover is handled within the link aggregation sublayer.

Implementing Link Aggregation

To implement link aggregation:

- Enable LACP on the network device
- Optionally set a non-default system priority for the device
- Optionally change the administratively assigned key for each port on the device
- Optionally enable single port LAGs on the device
- Enable LACP port active state on all ports that will take part in link aggregation
- Optionally change LAG parameters on each port
- Optionally change how flows will behave when changes take place to the LAG
- Optionally change the load balancing behavior for flows over the LAG
- Optionally assign static ports to a LAG when the partner device only supports a non-LACP method of aggregation

Link Aggregation Overview

This section provides an overview of link aggregation configuration.

LACP Operation

In order to allow LACP to determine whether a set of links connect to the same device, and to determine whether those links are compatible from the point of view of aggregation, it is necessary to be able to establish:

- A globally unique identifier for each device that participates in link aggregation.
- A means of identifying the set of capabilities associated with each port and with each aggregator, as understood by a given device.
- A means of identifying a LAG and its associated aggregator.

For each aggregatable port in the device, LACP:

- Maintains configuration information (reflecting the inherent properties of the individual links as well as those established by network administration) to control aggregation.
- Exchanges configuration information with other devices to allocate the link to a LAG.

Note: A given link is allocated to, at most, one LAG at a time. The allocation mechanism attempts to maximize aggregation, subject to management controls.

- Attaches the port to the aggregator used by the LAG, and detaches the port from the aggregator when it is no longer used by the LAG.
- Uses information from the partner device's link aggregation control entity to decide whether to aggregate ports.

The operation of LACP involves the following activities:

- Checking that candidate links can actually be aggregated.
- Controlling the addition of a link to a LAG and the creation of the group if necessary.
- Monitoring the status of aggregated links to ensure that the aggregation is still valid.
- Removing a link from a LAG if its membership is no longer valid, and removing the group if it no longer has any member links.

How a LAG Forms

LAGs form automatically with LACP enabled on the device. There are four criteria for forming a LAG. Both actor and partner ports must:

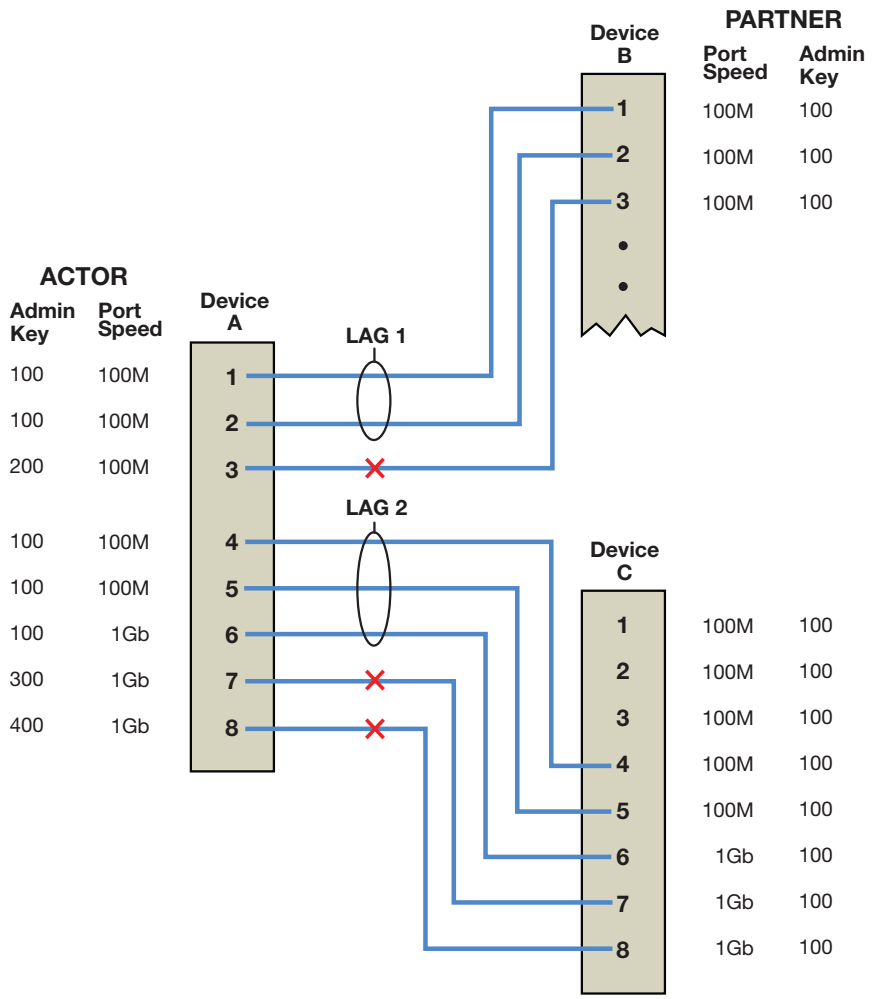
1. Operate in full duplex mode.
2. Have matching local LAG and physical port admin keys for the device controlling LAG formation.
3. Operate in parallel in that a LAG can have only two devices associated with it.
4. Consist of two or more physical actor to partner port pairings unless the single port LAG feature is enabled.

Figure 25-1 displays a LAG formation example containing three devices with five 100Mbps ports and three 1Gb ports configured. For this example, all ports are operating in full-duplex mode, and

the admin key for all LAG ports has been set to 100. Device A is the actor and therefore determines which ports will join a LAG. Devices B and C are the partners.

In our example two LAGs have formed because the actor ports are shared between two partner devices. Attempting to form a single LAG using all the actor ports would have broken the rule that actor and partner ports must operate in parallel.

Figure 25-1 LAG Formation



Actor ports 1 - 3 on device A directly connect to partner ports 1 - 3 on device B:

- We have already stated that all ports are operating in full-duplex mode, so rule 1 is satisfied for all three ports.
- Investigating the port admin keys, we see that ports 1 and 2 on device A are set to 100 (the same setting as all LAG ports on the device), while port 3 on device A is set to 200. Because the port admin keys are the same for both the LAG port and these physical ports, ports 1 and 2 satisfy rule 2. Because the admin key for physical port 3 is different from any possible LAG for this device, port 3 can not be part of any LAG.
- Because ports 1 and 2 for both the actor and partner operate in parallel with each other, rule 3 is satisfied for these ports.
- Rule 4 is satisfied, regardless of whether single port LAGs are enabled, because there are two aggregatable port pairings between devices A and B.

For these reasons, LAG 1 (lag.0.1) is formed using actor and partner ports 1 and 2.

Actor ports 4 - 8 on device A directly connect to partner ports 4 - 8 on device C:

- Because all ports are operating in full-duplex mode, rule one is satisfied for all five ports.
- Investigating port admin keys, we see that ports 4 - 6 on device A are set to 100 (the same setting as all LAG ports on the device), while ports 7 and 8 on device A are set to 300 and 400, respectively. Because port admin keys for all LAGs and the physical ports 4 - 6 are the same, physical ports 4 - 6 satisfy rule 2. Because the admin key settings for physical ports 7 and 8 do not agree with any LAG admin key setting on the device, ports 7 and 8 can not be part of any LAG.
- Because ports 4 - 6 for both the actor and partner operate in parallel with each other, rule 3 is satisfied for these ports.
- Rule 4 is satisfied, regardless of whether single port LAG is enabled, because there are three aggregatable port pairings between devices A and C.

For these reasons, LAG 2 is formed using actor and partner ports 4 - 6.

Note: Port speed is not a consideration in the forming phase for LAGs. LAG 2 contains 100Mbps and 1Gb port members.

Attached Ports

Once a LAG is formed, two steps must take place before traffic can pass over the LAG:

- The device that will choose which ports to move to the attached state must be identified
- The process of moving the chosen ports to the LACP attached state must take place

A system ID, made up of the device MAC address and the system priority, is associated with each device. The device with the lower system priority is in charge of selecting the LAG members to move to the attached state. If a system priority tie occurs, the system with the lower MAC address value breaks the tie.

Only LAG members with the same port speed can be moved to the attached state. In a case where multiple speeds are present in a LAG, the LAG member with the lowest port priority on the device in charge, as well as all other members with the same port speed as the member with the lowest port priority, are selected and moved to the attached state. Using LAG2 in Figure 25-1 on page 25-4 as an example, if the LAG2 member port priorities are set as shown in Table 25-1 on page 25-5, ports 4 and 5 are moved to the attached state.

Table 25-1 LAG2 Port Priority Assignments

Port Number	Port Speed	Port Priority
4	100Mbps	200
5	100Mbps	300
6	1Gb	300

This is true because port 4 has the lowest priority of the three ports currently in the LAG, and port 5 has the same speed as the port with the lowest priority in the LAG, regardless of its priority. Because port 6 has both a different speed and a higher priority than the port with the lowest priority in the LAG, it is not moved to the attached state.

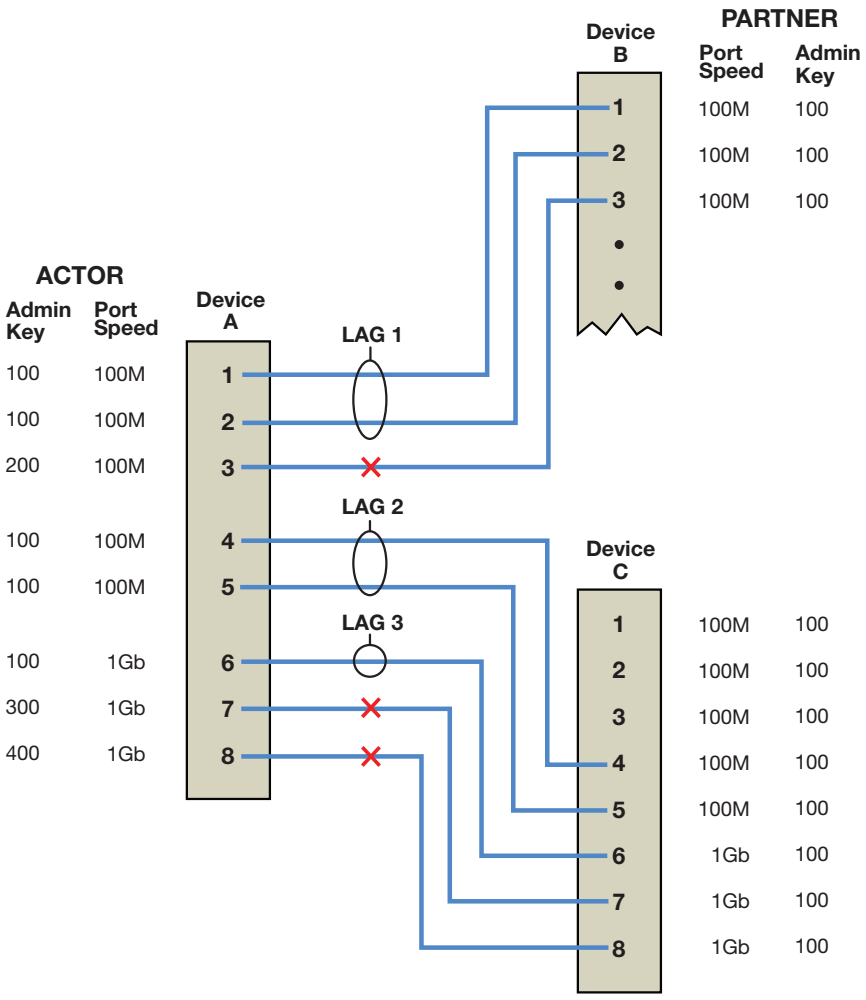
If LAG members with different port speeds should tie for the lowest port priority, the LAG member with the lowest port number breaks the tie. In our example, should all three ports have

the same port priority, ports 4 and 5 would still be the ports moved to the attached state because port 4 has the lowest port number and port 5 has the same port speed as port 4.

If in our example you wanted the reverse outcome of port 6 moved to the attached state instead of ports 4 and 5, setting port 6 to a lower priority than ports 4 and 5, as well as enabling the single port LAG feature on this device, would accomplish that goal.

Aggregatable ports not moved to the attached state are made available to form another LAG providing a LAG resource is available for this system. Port 6 in Figure 25-1 on page 25-4, was not moved to the attached state. The only criteria port 6 does not meet to form its own LAG is rule 4: being a single aggregatable port. The single port LAG feature must be enabled for port 6 to form a LAG. If single port LAG is enabled on this system, port 6 would form and attach to LAG 3. Figure 25-2 illustrates the three LAGs described in this example.

Figure 25-2 LAGs Moved to Attached State



Should an aggregatable port be available with all LAG resources depleted for this system, the port is placed in LACP standby state. Ports in standby state do not forward traffic. If all ports initially moved to the attach state for a given LAG become unavailable, a LAG resource will then be available. LACP will initiate a new selection process using the ports in standby state, using the same rules as the initial process of forming LAGs and moving ports to the attached state.

Single Port Attached State Rules

By default, a LAG must contain two or more actor and partner port pairs for the LAG to be initiated by this device. A feature exists to allow the creation of a single port LAG that is disabled by default. If single port LAG is enabled, a single port LAG can be created on this device. If single port LAG is disabled, a single port LAG will not be initiated by this device. If a peer device is able to form a single port LAG and advertises its willingness to do so, a single port LAG can form.

There are three conditions under which a single port LAG can exist and the LAG member can be moved to the attached state:

- The single port LAG feature is enabled.

  or,

- The single port LAG feature is disabled, but the peer device is able and willing to form a single port LAG.

  or,

- An already existing LAG configuration persists through a device or module reset. If upon reset there is only a single port active for an already existing LAG, that single port will move to the attached state regardless of the single port LAG setting.

LAG Port Parameters

LAG port parameters can be changed per port.

Table 25-2 specifies the LACP port parameters that can be changed.

Table 25-2 LAG Port Parameters

Term	Definition
Port Admin Key	The port admin key can be set for both the actor and partner side of the link. The admin key only affects the local device. LACP uses this value to determine which underlying physical ports are capable of aggregating. Aggregator ports allow only underlying ports with physical port and LAG admin keys that match to join a LAG. Setting the physical port admin key to a different value than any LAG resource on the device will ensure that this link does not join a LAG. Valid values are **1** - **65535**. Default value is **32768.**
Port Priority	Port priority can be set for both the actor and partner side of the link. The port priority plays a role in determining which set of ports will move to the attached state and pass traffic. The lower port priority, for the port on the system in charge of selecting ports to move to the attached state, determines which ports will actually move to the attached state. If a LAG is made up of ports with different speeds, setting a lower port priority to ports with the desired speed for the LAG will ensure that those ports move to the attached state. Port priority is also used to determine which ports join a LAG if the number of ports available exceeds the number of ports supported for that device. Valid values are **0** - **65535**, with lower values designating higher priority. Default value is **32768**.

Table 25-2 LAG Port Parameters (continued)

Term	Definition
Administrative State	A number of port level administrative states can be set for both the actor and partner ports. The following port administrative states are set by default: • **lacpactive -** Transmitting LACP PDUs is enabled. • **lacptimeout -** Transmitting LACP PDUs every 30 seconds. If this state is disabled, LACP PDUs are transmitted every 1 second. Note that the actor and partner LACP timeout values must agree. • **lacpagg -** Aggregation on this port is enabled. • **lacpsync -** Transition to synchronization state is allowed. • **lacpcollect -** Transition to collection state is allowed. • **lacpdist -** Transition to distribution state is allowed. • **lacpdef** - Transition to defaulted state is allowed. • **lacpexpire -** Transition to expired state is allowed. **Notes:** It is recommended that you do not change these default states unless you know what you are doing. Contact Extreme Networks customer support should you need assistance modifying port level administrative states.
Partner Default System ID	A default partner system ID can be set. This is a default MAC address for the system partner.
LACP PDU processing	(Optional) LACP PDU processing can be enabled or disabled for this port.

Flow Regeneration

Flow regeneration determines how flows will behave when a new port joins a link aggregation. When enabled, LACP will redistribute all existing flows over the LAG, taking into account the new port(s) that joined the LAG. It will also attempt to load balance existing flows to take advantage of the new port that has joined the LAG. When flow regeneration is disabled and a new port joins the LAG, the distribution of current flows remains unchanged and does not take advantage of the new port. All new flows will take into account the new port on the LAG. Flow regeneration is disabled by default.

The Out-Port Algorithm

The out-port algorithm determines the criteria to be used for data forwarding port selection. There are three algorithm criteria to choose from:

- Destination IP address and Source IP address (dip-sip). This is the most finely tuned criteria in that a port will be assigned based upon a specific IP address combination for the flow. All flows for this IP address combination transit the assigned physical port.
- Destination MAC address and Source MAC address (da-sa). This criteria is less finely tuned in that a port will be assigned based upon the MAC address combination for the flow. All flows for this MAC address combination transit the assigned port.
- Simple round robin (round-robin). This is the least finely tuned criteria in that a port is assigned based upon the next port in a round robin sequence with no consideration to the source or destination of the flow.

Note: The round robin out-port algorithm should not be assigned if fragmented frames exist in the network. Use of round robin can result in the fragments being sent out different ports, causing out of order packets.

Static Port Assignment

Static port assignment allows you to assign ports to a LAG when the partner device does not support LACP, but does support another proprietary form of link aggregation. To assign a static port, specify the LAG port ID, the admin key value for this LAG, and the ports to be assigned. If you do not specify an admin key value, a key will be assigned according to the specified aggregator. For example, a key of 4 would be assigned to lag.0.4.

Platform LAG and Physical Port Support

The number of LAGs and the number of ports per LAG supported are platform specific. The number of LAGs supported is on a system basis. See Table 25-3 for a listing of the number of LAGs and the number of ports per LAG supported for your platform.

Table 25-3 Extreme Networks Platform LAG Support

Extreme Networks Platform	Number of LAGs Supported	Number of Ports in a LAG
S-Series modules	127	64
S-Series SSA modules	62	64

Configuring Link Aggregation

This section provides details for the configuration of link aggregation on the S-Series products.

Table 25-4 lists link aggregation parameters and their default values.

Table 25-4 Default Link Aggregation Parameters

Parameter	Description	Default Value
LACP State	Current state of LACP on the device.	Enabled
System Priority	LACP system priority for this device.	32768
Port Key	The Port Administrative Key (also referred to as operational key).	32768
Port Priority	Determines which ports move to the attached state when ports of different speeds form a LAG. Also determines which ports join a LAG if the ports available exceed the number of ports supported by the device.	32768
Single Port State	Allows or disallows a LAG to be created with a single port.	Disabled (disallows creation of a single port LAG)
LACP Port Active State	Port state providing for transmission of LACP PDUs.	Disabled
LACP Port Timeout State	Port state determining the frequency of LACP PDU transmission and period before declaring the partner LACP port down if no response is received.	30 second: frequency of LACP PDU transmission 90 seconds: period before declaring the partner port down

Procedure 25-1 describes how to configure link aggregation.

Procedure 25-1 Configuring Link Aggregation

Step	Task	Command(s)
1.	In switch command mode, enable LACP on the device.	**set lacp** {**disable** | **enable**}
2.	Optionally, change the system priority for the device.	**set lacp asyspri** *value*
3.	Optionally, change the administratively assigned key for each aggregation on the device.	**set lacp aadminkey** *port-string value*
4.	Optionally, enable single port LAGs on the device.	**set lacp singleportlag** {**enable** | **disable**}

5.	Optionally, enable port active state for all LAG participating ports and modify the LAG port parameters. See Table 25-2 on page 25-7 for a description of port parameters.	**set port lacp port** *port-string*
		{
		[**aadminkey** *aadminkey*] [**aportpri** *aportpri*] [**padminsyspri** *padminsyspri*] [**padminsysid** *padminsysid*] [**padminkey** *padminkey*] [**padminportpri** *padminportpri*] [**padminport** *padminport*]
		[**aadminstate** {**lacpactive** | **lacptimeout** | **lacpagg** | **lacpsync** | **lacpcollect** | **lacpdist** | **lacpdef** | **lacpexpire**}]
		[**padminstate** {**lacpactive** | **lacptimeout** | **lacpagg** | **lacpsync** | **lacpcollect** | **lacpdist** | **lacpdef** | **lacpexpire**}]
		[**enable** | [**disable**]
		}

6.	Optionally, change how flows behave when a port joins or is removed from a LAG.	**set lacp flowRegeneration** {**enable** | **disable**}
7.	Optionally, change the out-port behavior for flows over the LAG.	**set lacp outportAlgorithm** {**dip-sip** | **da-sa** | **round-robin**}
8.	Optionally, assign static ports to a LAG when the partner device only supports a non-LACP method of aggregation.	**set lacp static** *lagportstring* [*key*] *port-string*

Table 25-5 describes how to manage link aggregation.

Table 25-5 Managing Link Aggregation

Task	Command
Reset LACP to the default state of enabled.	**clear lacp state**
Reset LACP system priority or admin key settings to the default values.	**clear lacp** {[**asyspri**] [**aadminkey** *port-string*]}
Remove specific static ports from an aggregation.	**clear lacp static** *lagportstring port-string*
Reset the single port LAG feature to the default value of disabled.	**clear lacp singleportlag**

Table 25-5 Managing Link Aggregation (continued)

Task	Command

Reset a link aggregation port setting to the default value for one or more ports. See Table 25-2 on page 25-7 for a description of port parameters.	**clear port lacp port** *port-string*
	{
	[**aadminkey**] [**aportpri**] [**padminsyspri**] [**padminsysid**] [**padminkey**] [**padminportpri**] [**padminport**]
	[**aadminstate** {**lacpactive** | **lacptimeout** | **lacpagg** | **lacpsync** | **lacpcollect** | **lacpdist** | **lacpdef** | **lacpexpire** | **all**}]
	[**padminstate** {**lacpactive** | **lacptimeout** | **lacpagg** | **lacpsync** | **lacpcollect** | **lacpdist** | **lacpdef** | **lacpexpire** | **all**}]
	}

Reset the LACP flow regeneration setting to its default value of disabled.	**clear lacp flowRegeneration**
Reset the LACP out-put algorithm setting to its default value of DIS-SIP.	**clear lacp outportAlgorithm**

Table 25-6 describes how to display link aggregation information and statistics.

Table 25-6 Displaying Link Aggregation Information and Statistics

Task	Command
Display the global LACP enable state, or display information about one or more aggregator ports.	**show lacp** [**state** | *port-string*]
Display the status of the single port LAG function.	**show lacp singleportlag**
Display link aggregation information for one or more underlying physical ports.	**show port lacp port** *port-string* {[**status** {**detail** | **summary**}] | [**counters**]} [**sort** {**port** | **lag**}]
Display LACP flow regeneration state.	**show lacp flowRegeneration**
Display the current configured out-port algorithm.	**show lacp outportAlgorithm**

Link Aggregation Configuration Examples

This section presents two configuration examples:

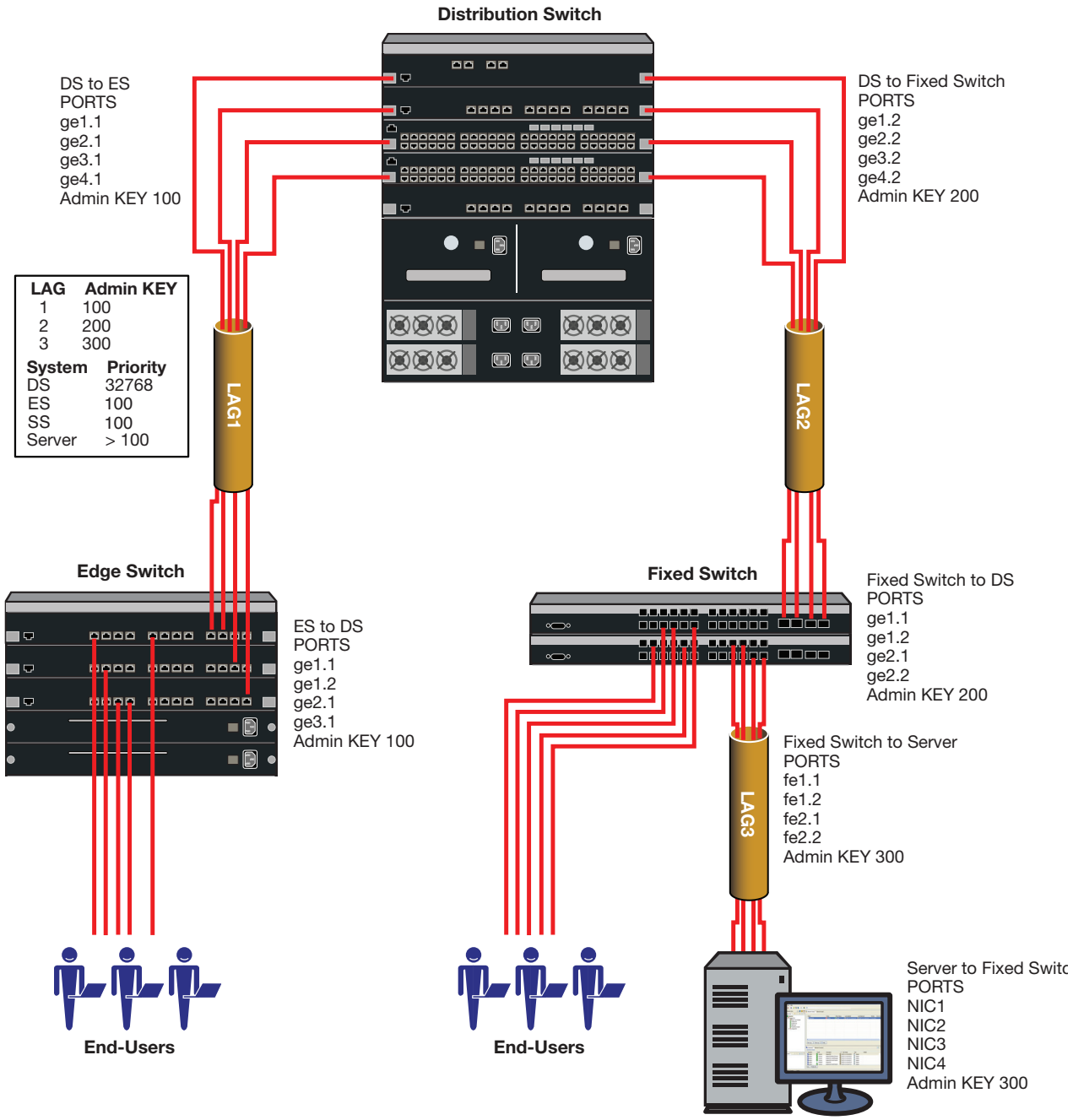
- An example of link aggregations between multiple devices
- An example of link aggregation when a LAG contains physical ports with different speeds

Link Aggregation Configuration Example 1

This example provides a link aggregation configuration example that includes an edge switch, a distribution switch, and two Fixed Switches that will aggregate both end-users at the edge and the data from a local server.

See Figure 25-3 on page 12 for an illustration of this example, including port, key, and system priority assignments.

Figure 25-3 Example 1 Multiple Device Configuration



Three LAGs are created for the example:

- LAG 1 provides an uplink aggregate of four 1Gb ports for the edge switch devices to the distribution switch.
- LAG2 provides an uplink aggregate of four 1Gb ports for the Fixed Switches to the distribution switch for both the end-user and server data flows.
- LAG3 provides an aggregate of four 100Mbps ports between the Fixed Switches and the server.

Each LAG consists of four ports. The primary goal of the aggregates in this example is to provide link and slot redundancy for the affected data streams. With that in mind, LAG members are

spread between available system slots. Four out of the five distribution switch available slots are used providing complete redundancy at the distribution switch. All three slots are used in the edge switch. The four ports from the server to the Fixed Switches and the Fixed Switches to the distribution switch are evenly split between the two Fixed Switches.

For this example we will manually configure the LAGs that will form and prevent any other LAGs from forming. Because we have specific port to LAG goals in mind, the first thing we want to do on each device is to ensure that LAGs form only where we configure them. Since the admin key for the LAG and its associated ports must agree for the LAG to form, an easy way to ensure that LAGs do not automatically form is to set the admin key for all LAGS on all devices to a non-default value. The physical ports will initially retain admin key defaults. In our example, the admin keys for all LAGs are set to the highest configurable value of 65535.

Both physical port and LAG admin keys will be set as shown in Table 25-7 to ensure that the LAGs form only for the desired ports.

Table 25-7 LAG and Physical Port Admin Key Assignments

Device	LAG	LAG Admin Key	Physical Port	Physical Port Admin Key
Distribution Switch	**1**	100	ge.1.1	100
			ge.2.1	100
			ge.3.1	100
			ge.4.1	100
	2	200	ge.1.2	200
			ge.2.2	200
			ge.3.2	200
			ge.4.2	200
Edge Switch	**1**	100	ge.1.1	100
			ge.1.2	100
			ge.2.1	100
			ge.3.1	100
Fixed Switch	**2**	200	ge.1.1	200
			ge.1.2	200
			ge.2.1	200
			ge.2.2	200
	3	300	ge.1.1	300
			ge.1.2	300
			ge.2.1	300
			ge.2.2	300
Server	**3**	300	NIC1 ETH	300
			NIC2 ETH	300
			NIC3 ETH	300
			NIC4 ETH	300

Which device determines port selection for the LAG is an optional consideration. If system priorities remain at the default value, the lowest MAC address device determines port selection for the LAG. For purposes of this example, we will set the system priority of the edge switch to 100 to ensure it will control port selection for LAG1, instead of the distribution switch. The Fixed Switch system priority will be set to 100 to ensure it will control port selection for LAG2, instead of the distribution switch. For the Fixed Switch to control port selection for LAG3 requires that you ensure that the server has a system priority higher than 100.

Each LAG in our example is made up of physical ports of the same speed, so there is no need to set the port priority to a non-default value. The only port value to be changed is the admin key for each physical port and each LAG. These modifications are detailed in Table 25-7 on page 25-13.

Given that the intent of the example is to have three LAGs of 4 ports each, there is no need to enable the single port LAG feature. Once the LAGs initiate, they will persist across resets. Should only a single port be active after a reset, the LAG will form regardless of the single port LAG feature setting.

Flow regeneration is enabled for the distribution switch and edge switch in our example. This setting will ensure that should a LAG port become disabled and then become active again, LACP will redistribute existing flows over all the ports in the new LAG. Both the 7100-Series and Fixed Switch platforms do not support flow regeneration.

The output algorithm defaults to selecting the output port based upon the destination and source IP address. This setting will not be changed in our example. In any case, note that the Fixed Switch does not support the output algorithm feature.

Configuring the Distribution Switch

The first thing we want to do is set the admin key for all LAGs to the non-default value of 65535 so that no LAGs will automatically form:

```
S Chassis(rw)->set lacp aadminkey lag.0.* 65535
```

LAGs 1 and 2 will form on the distribution switch so we need to set the admin keys for these LAGs:

```
S Chassis(rw)->set lacp aadminkey lag.0.1 100
S Chassis(rw)->set lacp aadminkey lag.0.2 200
```

We next want to enable the port active state and set the admin keys for the distribution switch physical ports:

```
S Chassis(rw)->set port lacp port ge.1.1 aadminkey 100 enable
S Chassis(rw)->set port lacp port ge.2.1 aadminkey 100 enable
S Chassis(rw)->set port lacp port ge.3.1 aadminkey 100 enable
S Chassis(rw)->set port lacp port ge.4.1 aadminkey 100 enable
S Chassis(rw)->set port lacp port ge.1.2 aadminkey 200 enable
S Chassis(rw)->set port lacp port ge.2.2 aadminkey 200 enable
S Chassis(rw)->set port lacp port ge.3.2 aadminkey 200 enable
S Chassis(rw)->set port lacp port ge.4.2 aadminkey 200 enable
```

Because we want the edge switch and the Fixed Switch to be in charge of port selection, the system priority for the distribution switch will be left at the default value of 32768. We next enable flow regeneration on the distribution switch:

```
S Chassis(rw)->set lacp flowRegeneration enable
```

Configuring the Edge Switch

The first thing we want to do is set the admin key for all LAGs to the non-default value of 65535 so that no LAGs will automatically form:

```
S Chassis(rw)->set lacp aadminkey lag.0.* 65535
```

LAG 1 will form on the edge switch so we need to set the admin key for this LAG:

```
S Chassis(rw)->set lacp aadminkey lag.0.1 100
```

We next want to enable the port active state and set the admin keys for the edge switch physical ports:

```
S Chassis(rw)->set port lacp port ge.1.1 aadminkey 100 enable
S Chassis(rw)->set port lacp port ge.1.2 aadminkey 100 enable
S Chassis(rw)->set port lacp port ge.2.1 aadminkey 100 enable
S Chassis(rw)->set port lacp port ge.3.1 aadminkey 100 enable
```

Next we want to change the system priority for the edge switch so that it will be in charge of port selection on LAG1:

```
S Chassis(rw)->set lacp asyspri 100
```

We next enable flow regeneration on the edge switch:

```
S Chassis(rw)->set lacp flowRegeneration enable
```

Configuring the Fixed Switch

The first thing we want to do is set the admin key for all LAGs to the non-default value of 65535 so that no LAGs will automatically form:

```
FixedSwitch(rw)->set lacp aadminkey lag.0.* 65535
```

LAGs 2 and 3 will form on the Fixed Switch so we need to set the admin key for this LAG:

```
FixedSwitch(rw)->set lacp aadminkey lag.0.2 200
FixedSwitch(rw)->set lacp aadminkey lag.0.3 300
```

We next want to enable the port active state and set the admin keys for the Fixed Switch physical ports:

```
FixedSwitch(rw)->set port lacp port ge.1.1 aadminkey 200 enable
FixedSwitch(rw)->set port lacp port ge.1.2 aadminkey 200 enable
FixedSwitch(rw)->set port lacp port ge.2.1 aadminkey 200 enable
FixedSwitch(rw)->set port lacp port ge.2.2 aadminkey 200 enable
FixedSwitch(rw)->set port lacp port ge.1.1 aadminkey 300 enable
FixedSwitch(rw)->set port lacp port ge.1.2 aadminkey 300 enable
FixedSwitch(rw)->set port lacp port ge.2.1 aadminkey 300 enable
FixedSwitch(rw)->set port lacp port ge.2.2 aadminkey 300 enable
```

Next we want to change the system priority for the Fixed Switch so that it will be in charge of port selection on LAGs 2 and 3:

```
FixedSwitch(rw)->set lacp asyspri 100
```

Configuring the Server

Configuring link aggregation on the server is dependent upon the installed LACP application. There are three aspects to link aggregation on the server you must ensure for this example:

- The admin key for LAG3 must be set to 300
- The admin keys for each NIC port must be set to 300
- The system priority for the server must be set greater than 100 to ensure that the Fixed Switch will control port selection

This completes the example 1 configuration.

Link Aggregation Configuration Example 2

It is unlikely that you will run out of LAG resources for most link aggregation configurations, but it is possible. See Table 25-3 on page 25-9 for a listing of LAG support for your system. Should you run out of LAG resources, excess aggregatable ports are placed in standby mode.

Making use of the port priority parameter, this example shows how you can ensure the order in which aggregatable ports form a LAG and are moved to the attached state. In configuration example 2, two uplink LAGs will be manually configured between two edge switch chassis. The first LAG consists of two 1 Gb ports. The second LAG consists of eight 100 Mbps ports. In this example we will ensure that the two 1Gb port LAG forms before the eight 100 Mbps port LAG.

See Figure 25-4 on page 25-17 for an illustration of this example, including port, key and port priority assignments.

The LAG configuration will ensure that the two 1Gb ports attach to the first available LAG (LAG1). The eight 100Mbps ports will then attach to the second available LAG (LAG2)

Which device determines port selection for the LAG is an optional consideration. For this example, system priorities are not modified, the lowest MAC address device will determine port selection for the LAG.

There are two physical port speeds in our example, 100Mbps and 1Gb. A LAG only moves ports of the same speed to the attached state. Selecting the ports to move to attached state is based upon the lowest port priority. If port priorities are the same, the lowest port number breaks the tie. For our example, we want to ensure that the 1Gb ports are moved to the attached state for LAG1. Port priority for 1Gb ports is set to 100. Port priority for 100Mbps ports is left at the default value of 32768.

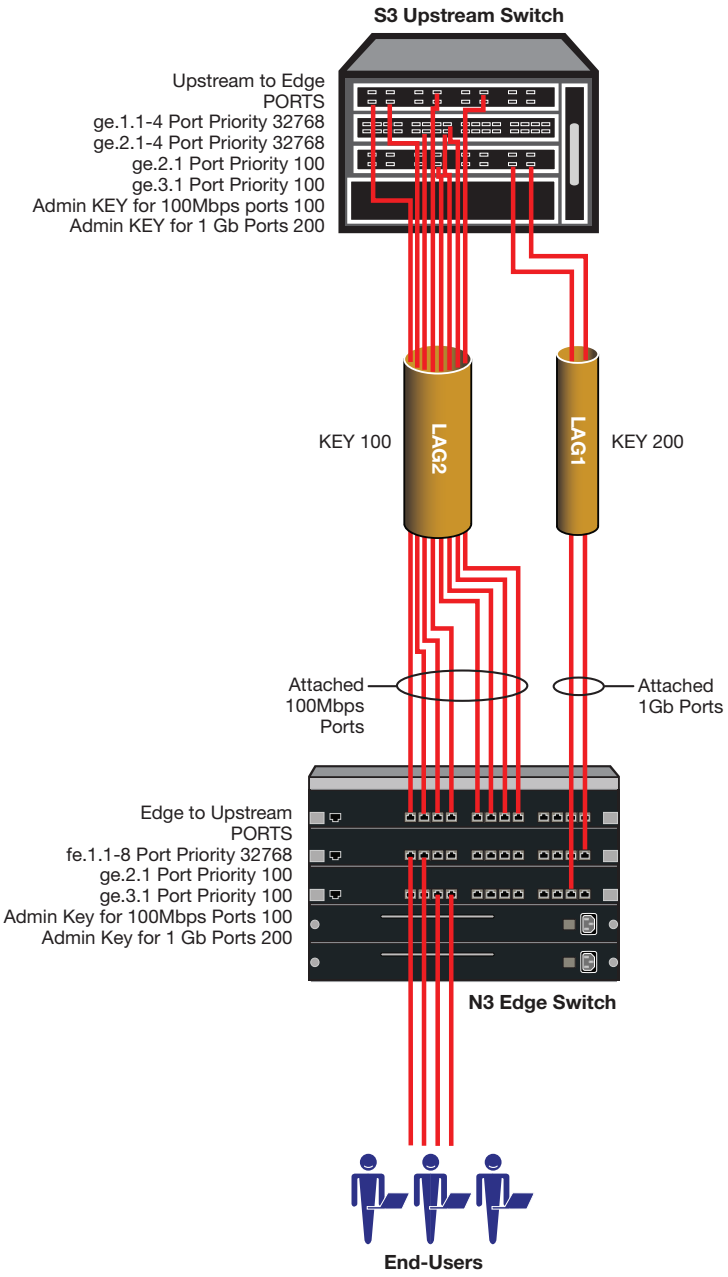
The admin key for each 100 Mbps to 1Gb physical port link and LAG in the example is set to 100, and for each 1Gb to 1Gb physical port link and LAG is set to 200. This ensures that LAGs will form for each set of ports.

For this example we will allow single port LAGs to form. The single port LAG feature will be set to enabled for both devices.

Flow regeneration is enabled for both devices in our example. This setting will ensure that should a LAG port drop out and then become active again, LACP will redistribute existing flows over all the ports in the new LAG.

The output algorithm defaults to selecting the output port based upon the destination and source IP address. This setting will not be changed in our example.

Figure 25-4 Example 2 Configuration



Configuring the Edge Switch

For this example, we want LAGs to form wherever they can so we will not change the default admin key setting for all LAGs as we did in the multiple device example. Because we want LAG1 and LAG2, as described for this example, to form for specific ports, we set separate admin keys for these LAGs (LAG1 to 200 and LAG2 to 100):

```
S Chassis(rw)->set lacp aadminkey lag.0.1 200
S Chassis(rw)->set lacp aadminkey lag.0.2 100
```

We next want to enable the port active state and set the admin keys for the edge switch physical ports associated with LAG1 and LAG2:

```
S Chassis(rw)->set port lacp port ge.2.1 aadminkey 200 enable
S Chassis(rw)->set port lacp port ge.3.1 aadminkey 200 enable
S Chassis(rw)->set port lacp port fe.1.1 aadminkey 100 enable
S Chassis(rw)->set port lacp port fe.1.2 aadminkey 100 enable
S Chassis(rw)->set port lacp port fe.1.3 aadminkey 100 enable
S Chassis(rw)->set port lacp port fe.1.4 aadminkey 100 enable
S Chassis(rw)->set port lacp port fe.1.5 aadminkey 100 enable
S Chassis(rw)->set port lacp port fe.1.6 aadminkey 100 enable
S Chassis(rw)->set port lacp port fe.1.7 aadminkey 100 enable
S Chassis(rw)->set port lacp port fe.1.8 aadminkey 100 enable
```

System priority determines which device will be in charge of port selection. This is an optional consideration. For this example we will leave system priority at the default value and allow the device with the lowest MAC address to determine port selection.

Port priority determines which aggregatable ports available for a LAG are moved to the attached state when different speed physical ports form a LAG. For this example we want to ensure that the 1Gb ports move to the attached state for LAG1. We will set the port priority to 100 for the 1Gb actor ports should this device be in charge of selecting ports to move to the attached state:

```
S Chassis(rw)->set port lacp port ge.2.1 aportpri 100
S Chassis(rw)->set port lacp port ge.3.1 aportpri 100
```

We next enable single port LAGs on this device:

```
S Chassis(rw)->set lacp singleportlag enable
```

We next enable flow regeneration on the edge switch:

```
S Chassis(rw)->set lacp flowRegeneration enable
```

Configuring the Upstream Switch

For this example, we want LAGs to form wherever they can so we will not change the default admin key setting for all LAGs as we did in the multiple device example. Because we want LAG1 and LAG2, as described for this example, to form for specific ports, we set the admin key of LAG 1 to 200 and the admin key of LAG 2 to 100:

```
S Chassis(rw)->set lacp aadminkey lag.0.1 200
S Chassis(rw)->set lacp aadminkey lag.0.2 100
```

We next want to enable the port active state and set the admin keys for the upstream switch physical ports associated with LAG 1 and LAG 2:

```
S Chassis(rw)->set port lacp port ge.2.1 aadminkey 200 enable
S Chassis(rw)->set port lacp port ge.3.1 aadminkey 200 enable
S Chassis(rw)->set port lacp port ge.1.1 aadminkey 100 enable
S Chassis(rw)->set port lacp port ge.1.2 aadminkey 100 enable
S Chassis(rw)->set port lacp port ge.1.3 aadminkey 100 enable
S Chassis(rw)->set port lacp port ge.1.4 aadminkey 100 enable
S Chassis(rw)->set port lacp port ge.2.1 aadminkey 100 enable
S Chassis(rw)->set port lacp port ge.2.2 aadminkey 100 enable
S Chassis(rw)->set port lacp port ge.2.3 aadminkey 100 enable
S Chassis(rw)->set port lacp port ge.2.4 aadminkey 100 enable
```

System priority determines which device will be in charge of port selection. This is an optional consideration. For this example we will leave system priority at the default value and allow the device with the lowest MAC address to determine port selection.

Port priority determines which aggregatable ports available for a LAG are moved to the attached state when different speed physical ports form a LAG. For this example we want to ensure that the 1Gb ports move to the attached state for LAG1. We will set the port priority to 100 for the 1Gb actor ports should this device be in charge of selecting ports to move to the attached state:

```
S Chassis(rw)->set port lacp port ge.2.1 aportpri 100
S Chassis(rw)->set port lacp port ge.3.1 aportpri 100
```

We next enable single port LAGs on this device:

```
S Chassis(rw)->set lacp singleportlag enable
```

We next enable flow regeneration on the upstream switch:

```
S Chassis(rw)->set lacp flowRegeneration enable
```

This completes the example 2 configuration.

Terms and Definitions

Table 25-8 lists terms and definitions used in this link aggregation configuration discussion.

Table 25-8 Link Aggregation Configuration Terms and Definitions

Term	Definition
Aggregator	Virtual port that controls link aggregation for underlying physical ports. Each device provides aggregator ports, which are designated in the CLI as **lag.0.1** through **lag.0.x** (depending upon the device, see Table 25-3 on page 25-9 for LAG resources available on your device).
LAG	Link Aggregation Group. Once underlying physical ports (i.e.; **ge.x.x**) are associated with an aggregator port, the resulting aggregation will be represented as one LAG with a **lag.x.x** port designation.
LACPDU	Link Aggregation Control Protocol Data Unit. The protocol exchanges aggregation state/mode information by way of a port's actor and partner operational states. LACPDUs sent by the first party (the actor) convey to the second party (the actor's protocol partner) what the actor knows, both about its own state and that of its partner.
Actor and Partner	An actor is the local device sending LACPDUs. Its protocol partner is the device on the other end of the link aggregation. Each maintains current status of the other via LACPDUs containing information about their ports' LACP status and operational state.
Admin Key	Value assigned to aggregator ports and physical ports that are candidates for joining a LAG. The LACP implementation uses this value to determine which underlying physical ports are capable of aggregating by comparing keys. Aggregator ports allow only underlying ports with admin keys that match the aggregator to join their LAG.
Port Priority	Port priority determines which physical ports are moved to the attached state when physical ports of differing speeds form a LAG. Port priority also determines which ports will join a LAG when the number of supported ports for a LAG is exceeded.
System Priority	Value used to build a LAG ID, which determines aggregation precedence. If there are two partner devices competing for the same aggregator, LACP compares the LAG IDs for each grouping of ports. The LAG with the lower LAG ID is given precedence and will be allowed to use the aggregator.

26

Policy Configuration

This document describes the Extreme Networks policy feature and its configuration on Extreme Networks S-Series devices.

Using Policy in Your Network

Policy is a component of Secure Networks that provides for the configuration of role-based profiles for securing and provisioning network resources based upon the role the user or device plays within the enterprise. By first defining the user or device role, network resources can be granularly tailored to a specific user, system, service, or port-based context by configuring and assigning rules to the policy role. A policy role can be configured for any combination of Class of Service, VLAN assignment, classification rule precedence, logging, accounting, or default behavior based upon L2, L3, and L4 packet fields. Hybrid authentication allows either policy or dynamic VLAN assignment, or both, to be applied through RADIUS authorization.

The three primary benefits of using Extreme Networks Secure Networks policy in your network are provisioning and control of network resources, security, and centralized operational efficiency using the Extreme Networks NetSight Policy Manager.

Policy provides for the provisioning and control of network resources by creating policy roles that allow you to determine network provisioning and control at the appropriate network layer, for a given user or device. With a role defined, rules can be created based upon up to 29 traffic classification types for traffic drop or forwarding. A Class of Service (CoS) can be associated with each role for purposes of setting priority, forwarding queue, rate limiting, and rate shaping.

Security can be enhanced by allowing only intended users and devices access to network protocols and capabilities. Some examples are:

- Using HTTP redirection to force a client's web browser to be redirected to a particular administrative web page.
- Ensuring that only approved stations can use SNMP, preventing unauthorized stations from viewing, reading, and writing network management information

- Preventing edge clients from attaching network services that are appropriately restricted to data centers and managed by the enterprise IT organization such as DHCP and DNS services
- Identifying and restricting routing to legitimate routing IP addresses to prevent DoS, spoofing, data integrity and other routing related security issues
- Ensuring that FTP/TFTP file transfers and firmware upgrades only originate from authorized file and configuration management servers
- Preventing clients from using legacy protocols such as IPX, AppleTalk, and DECnet that should no longer be running on your network

Extreme Networks NetSight Policy Manager provides a centralized point and click configuration, and one click pushing of defined policy out to all network elements. Use the Extreme Networks NetSight Policy Manager for ease of initial configuration and response to security and provisioning issues that may come up during real-time network operation.

Implementing Policy

To implement policy:

- Identify the roles of users and devices in your organization that access the network
- Create a policy role for each identified user role
- Associate classification rules and administrative profiles with each policy role
- Optionally, configure a class of service and associate it directly with the policy role or through a classification rule
- Optionally, enable hybrid authentication, which allows RADIUS filter-ID and tunnel attributes to be used to dynamically assign policy roles and VLANs to authenticating users
- Optionally, set device response to invalid policy
- Optionally configure Captive Portal Redirection which uses HTTP redirection to force a client's web browser to be redirected to a particular administrative web page.

Policy Overview

Introduction

This section provides an overview of policy configuration. Policy is implemented on an Extreme Networks platform by associating users and devices in the network with defined enterprise roles (such as sales, engineering, or administration) that are configured in a policy role. The policy role is associated with rules that define how network resources will be provisioned and controlled for role members, as well as how security will be applied to the role member. An administrative profile associates a specific role member traffic classification with a policy role.

Note: In a CLI configuration context, the policy role is configured within a policy profile using the **set policy profile** command. throughout this discussion, policy role and policy profile mean the same thing.

The Extreme Networks NetSight Policy Manager

Extreme Networks NetSight Policy Manager is a management GUI that automates the definition and enforcement of network-wide policy rules. It eliminates the need to configure policies on a device-by-device basis using complex CLI commands. The Policy Manager's GUI provides ease of classification rule and policy role creation, because you only define policies once using an easy to

understand point and click GUI— and regardless of the number of moves, adds or changes to the policy role, Policy Manager automatically enforces roles on Extreme Networks security-enabled infrastructure devices.

This document presents policy configuration from the perspective of the CLI. Though it is possible to configure policy from the CLI, CLI policy configuration in even a small network can be prohibitively complex from an operational point of view. It is highly recommended that policy configuration be performed using the NetSight Policy Manager. The NetSight Policy Manager provides:

- Ease of rule and policy role creation
- The ability to store and and retrieve roles and policies
- The ability, with a single click, to enforce policy across multiple devices

The official Policy Manager documentation is accessed using online help from within the application. This online documentation completely covers the configuration of policy in a Policy Manager context. For access to the Policy Manager data sheet or to setup a demo of the product, see http://www.extremenetworks.com/product/netsight/.

Understanding Roles in a Secure Network

The capacity to define roles is directly derived from the ability of the Extreme Networks S-Series, stackable, and standalone devices to isolate packet flows by inspecting Layer 2, Layer 3, and Layer 4 packet fields while maintaining line rate. This capability allows for the granular application of a policy to a:

- Specific user (MAC, IP address or interface)
- Group of users (masked MAC or IP address)
- System (IP address)
- Service (such as TCP or UDP)
- Port (physical or application)

Because users, devices, and applications are all identifiable within a flow, a network administrator has the capacity to define and control network access and usage by the actual role the user or device plays in the network. The nature of the security challenge, application access, or amount of network resource required by a given attached user or device, is very much dependent upon the "role" that user or device plays in the enterprise. Defining and applying each role assures that network access and resource usage align with the security requirements, network capabilities, and legitimate user needs as defined by the network administrator.

The Policy Role

A role, such as sales, admin, or engineering, is first identified and defined in the abstract as the basis for configuring a policy role. Once a role is defined, a policy role is configured and applied to the appropriate context using a set of rules that can control and prioritize various types of network traffic. The rules that make up a policy role contain both classification definitions and actions to be enforced when a classification is matched. Classifications include Layer 2, Layer 3, and Layer 4 packet fields. Policy actions that can be enforced include VLAN assignment, filtering, inbound rate limiting, outbound rate shaping, priority class mapping and logging.

Policy Roles

Defining a Policy Role

The policy role is a container that holds all aspects of policy configuration for a specific role. Policy roles are identified by a numeric profile-index value between **1** and the maximum number of roles supported on the platform. Please see your device's firmware release notes for the maximum number of roles supported. Policy roles are configured using the **set policy profile** command. Policy configuration is either directly specified with the **set policy profile** command or is associated with the role by specifying the profile-index value within the command syntax where the given policy option is configured. For example, when configuring a policy maptable entry using the **set policy maptable** command (see "VLAN-to-Policy Mapping" on page 26-5), the command syntax requires that you identify the policy role the maptable entry will be associated with, by specifying the profile-index value.

When modifying an existing policy role the default behavior is to replace the existing role with the new policy role configuration. Use the **append** option to limit the change to the existing policy role to the options specified in the entered command.

A policy role can also be identified by a text name of between 1 and 64 characters. This name value is used by the RADIUS filter-ID attribute to identify the policy role to be applied by the switch with a successful authentication.

The following example creates a policy profile with a profile-index value of **1** and a profile name, **student**, to be used by the RADIUS filter-ID functionality:

```
S Chassis(rw)->set policy profile 1 student
```

Setting a Default VLAN for this Role

A default VLAN can be configured for a policy role. A default VLAN will only be used when either a VLAN is not specifically assigned by a classification rule or all policy role classification rules are missed. To configure a default VLAN, enable **pvid-status** and specify the port VLAN to be used. **pvid-status** is disabled by default.

Note: Extreme Networks supports the assignment of port VLAN-IDs 1 - 4094. VLAN-IDs 0 and 4095 can not be assigned as port VLAN-IDs, but do have special meanings within a policy context and can be assigned to the **pvid** parameter (See "VLAN Support on Extreme Networks S-Series Switches" on page 24-6 for further information on these two VLAN-IDs). Within a policy context:

- **0** - Specifies an explicit deny all
- **4095** - Specifies an explicit permit all

The following example creates a policy profile with a profile-index value of **1** and associates with it a default VLAN with an ID of **2**.

```
S Chassis(rw)->set policy profile 1 pvid-status enable pvid 2
```

Assigning a Class of Service to this Role

How a packet is treated as it transits the link can be configured in the Class of Service (CoS). It is through a CoS that Quality of Service (QoS) is implemented. A CoS can be configured for the following values:

- 802.1p priority
- IP Type of Service (ToS) rewrite value
- Priority Transmit Queue (TxQ) along with a forwarding behavior
- Inbound and outbound rate limiter per transmit queue

- Outbound rate shaper per transmit queue

CoS configurations are identified by a numeric value between 0 - 255. 0 - 7 are fixed 802.1p CoS configurations. CoS configurations 8 - 255 are user configurable. Policy uses the **cos** option followed by the CoS configuration ID value to associate a CoS with a policy role.

See *Chapter 53*, **Quality of Service (QoS) Configuration** for a complete discussion of QoS configuration.

The following example creates a policy profile with a profile-index value of **1** and associates with the profile a user configured CoS **8**:

```
S Chassis(rw)->set policy profile 1 cos-status enable cos 8
```

Adding Tagged, Untagged, and Forbidden Ports to the VLAN Egress Lists

The VLAN Egress list contains a list of ports that a frame for this VLAN can exit. Specified ports are automatically assigned to the VLAN egress list for this policy role as tagged, untagged, or forbidden. Ports are added to the VLAN egress list using the **egress-vlans**, **forbidden-vlans**, and **untagged-vlans** options of the **set policy profile** command.

Applying a Destination Mirror to a Role

Destination mirrors can be created for one or more IP addresses or VLANs. See the Chapter 8, **Port Mirroring Configuration** for destination mirror information. Use the **mirror-destination** role option to specify a destination mirror index value to apply to this role.

Clearing already configured destination mirrors and prohibiting mirroring can also be set per role. Use the **clear-mirror** role option to clear mirroring on this role. Use the **prohibit-mirror** role option to prohibit mirroring on this role.

Overwriting VLAN Tags Priority and Classification Settings

TCI overwrite supports the application of rules to a policy role that overwrite the current user priority and other classification information in the VLAN tag's TCI field. TCI overwrite can be used at either the port level or associated with a policy role, but not both. When enabled at the port level, port level configuration takes precedence and the TCI overwrite configuration at the policy role level is ignored.

Use the **set policy profile tci-overwrite** command to enable TCI overwrite on a policy role.

Use the **set port tcioverwrite** command to enable TCI overwrite on the specified port.

VLAN-to-Policy Mapping

VLAN-to-Policy mapping provides for the manual configuration of a VLAN-to-Policy association that creates a policy maptable entry between the specified VLAN and the specified policy role. A policy maptable holds the VLAN-to-Policy mappings. When an incoming tagged VLAN packet is seen by the switch, a lookup of the policy maptable determines whether a VLAN-to-policy mapping exists. If the mapping exists, the associated policy is applied to this packet.

This feature can be used at the distribution layer in environments where non-policy capable edge switches are deployed and there is no possibility of applying Extreme Networks policy at the edge. Tagged frames received at the distribution layer interface for a VLAN with an entry in the policy maptable will have the associated policy applied to the frame.

Use the **set policy maptable** command specifying a single VLAN ID or range of IDs and the policy profile-index to create a policy maptable entry.

The following example creates a policy maptable entry associating VLAN **100** and policy profile **10**:

```
S Chassis(rw)->set policy maptable 100 10
```

Applying Policy Using the RADIUS Response Attributes

If an authentication method that requires communication with an authentication server is configured for a user, the RADIUS filter-ID attribute can be used to dynamically assign a policy role to the authenticating user. Supported RADIUS attributes are sent to the switch in the RADIUS access-accept message. The RADIUS filter-ID can also be applied in hybrid authentication mode. Hybrid authentication mode determines how the RADIUS filter-ID and the three RFC 3580 VLAN tunnel attributes (VLAN Authorization), when either or all are included in the RADIUS access-accept message, will be handled by the switch. The three VLAN tunnel attributes define the base VLAN-ID to be applied to the user. In either case, conflict resolution between RADIUS attributes is provided by the maptable response feature.

Note: VLAN-to-policy mapping to maptable response configuration behavior is as follows:

- If the RADIUS response is set to **policy**, any VLAN-to-policy maptable configuration is ignored for all platforms.
- If the RADIUS response is set to **tunnel**, VLAN-to-policy mapping can occur on an S-Series platform.
- If the RADIUS response is set to **both** and both the filter-ID and tunnel attributes are present, VLAN-to-policy mapping configuration is ignored.

See the "Policy Maptable Response" on page 56-13 for a detailed RADIUS response discussion.

Please see *"Configuring RADIUS"* on page 56-29 for a discussion of RADIUS configuration, the RADIUS filter-ID, and VLAN authorization.

Use the **policy** option of the **set policy maptable response** command to configure the switch to dynamically assign a policy using the RADIUS filter-ID in the RADIUS response message.

The following example specifies that the RADIUS filter-ID, if it is present in the RADIUS response message when a user authenticates, should be used to apply the specified policy to the user:

```
S Chassis(rw)->set policy maptable response policy
```

Applying Policy Using Hybrid Authentication Mode

Hybrid authentication is an authentication capability that allows the switch to use both the filter-ID and tunnel attributes in the RADIUS response message to determine how to treat the authenticating user.

Hybrid authentication is configured by specifying the **both** option in the **set policy maptable response** command. The **both** option:

- Applies the VLAN tunnel attributes if they exist and the filter-ID attribute does not
- Applies the filter-ID attribute if it exists and the VLAN tunnel attributes do not
- Applies both the filter-ID and the VLAN tunnel attributes if all attributes exist

If all attributes exist, the following rules apply:

- The policy role will be enforced, with the exception that any port PVID specified in the role will be replaced with the VLAN tunnel attributes
- The policy map is ignored because the policy role is explicitly assigned
- VLAN classification rules are assigned as defined by the policy role

vlanauthorization must be enabled or the VLAN tunnel attributes are ignored and the default VLAN is used. Please see "Configuring VLAN Authorization" on page 56-28 for a complete VLAN Authorization discussion.

Hybrid Mode support eliminates the dependency of VLAN assignment based on roles. As a result, VLANs can be assigned via the tunnel-private-group-ID, as defined per RFC3580, while assigning roles via the filter-ID. This separation gives administrators more flexibility to segment their networks for efficiency beyond the role limits associated with the B3, C3, and G3 platforms.

The following example specifies that either or both the vlan-tunnel and filter-ID attributes can be included in the RADIUS response message:

```
S Chassis(rw)->set policy maptable response both
```

Device Response to Invalid Policy

The action that the device should take when asked to apply an invalid or unknown policy can be specified. The available actions are:

- Ignore the result and search for the next policy assignment rule. If all rules are missed, the default policy is applied.
- Block traffic
- Forward traffic as if no policy has been assigned using 802.1D/Q rules

Use the **set policy invalid action** command to specify a default action to take when asked to apply an invalid or unknown policy.

The following example specifies that an attempt to apply an invalid or unknown policy should be ignored:

```
S Chassis(rw)->set policy invalid action default-policy
```

Disabling an Ingress Port on First Profile Rule Use

A policy profile can be set to disable an ingress port on the first use of any profile rule assigned to the policy profile. The disable-port feature is disabled by default. Use the **set policy profile disable-port** command to enable or disable the disable-port feature for the specified policy profile. This command disables the port if any rule for this profile is used. To limit disabling of ports to the first use of a specific policy rule, see "Disabling an Ingress Port Per Policy Rule" on page 26-11.

Use the **clear policy disabled-ports** to clear ports from the disabled state due to the first use of a policy rule on those ports.

Use the **show policy disabled-ports** command to display ports that have been disabled by a profile rule enabled for disabled ports.

Use the **show policy rule port-hit** command to display rule hits that have occurred, displayed on a per port basis.

Use the **show policy rule usage-list** command to display usage for all rules whether a rule hit has occurred or not. The usage field of this command displays whether a hit has occurred for a listed rule.

Use the **clear policy usage-list** command to clear statistics displayed in the **show policy rule usage-list** command. This command only clears displayed statistics.

Clearing Policy Rule Usage Statistics

Statistics are gathered for policy rule usage on a port basis for the first time a rule hit occurs and on a usage list basis for all rules assigned to a policy. Use the **set policy autoclear** command to clear these statistics when operational status "up" is detected on the port.

Classification Rules

Classification rules associate specific traffic classifications or policy behaviors with the policy role. There are two aspects of classification rule configuration:

- The association of a traffic classification with a policy role by assigning the traffic classification to an administrative profile.
- The assignment of policy rules that define desired policy behaviors for the specified traffic classification type.

Both the administrative profile and policy rules are associated with the policy role by specifying the **admin-pid** option, in the case of an administrative profile, or a **profile-index** value, in the case of the policy rule. Administrative profiles and policy rules are configured using the **set policy rule** command.

The administrative profile assigns a traffic classification to a policy role by using the **admin-profile** option of the **set policy rule** command.

Policy rules are based on traffic classifications. Table 26-1 on page 26-8 provides the supported policy rule traffic classification command options and definitions.

A detailed discussion of supported traffic classifications is available in the "Traffic Classification Rules" section of the NetSight Policy Manager online help.

Table 26-1 Administrative Policy and Policy Rule Traffic Classifications

Traffic Classification	Description	Attribute ID
macsource	Classifies based on MAC source address.	**1**
macdest	Classifies based on MAC destination address.	**2**
ipxsource	Classifies based on source IPX address.	**3**
ipxdest	Classifies based on destination IPX address.	**4**
ipxsourcesocket	Classifies based on source IPX socket.	**5**
ipxdestsocket	Classifies based on destination IPX socket.	**6**
ipxclass	Classifies based on transmission control in IPX.	**7**
ipxtype	Classifies based on IPX packet type.	**8**
ip6source	Classifies based on IPv6 source address.	**9**
ip6dest	Classifies based on IPv6 destination address.	**10**
ip6flowlabel	Classifies based on IPv6 flow label.	**11**
ipsourcesocket	Classifies based on source IP address.	**12**
ipdestsocket	Classifies based on destination IP address.	**13**
ip frag	Classifies based on IP fragmentation value.	**14**
udpsourceportip	Classifies based on UDP source port.	**15**
udpdestportip	Classifies based on UDP destination port.	**16**
tcpsourceportip	Classifies based on TCP source port.	**17**
tcpdestportip	Classifies based on TCP destination port.	**18**
icmptype	Classifies based on ICMP packet type.	**19**
ipttl	Classifies based on Time-To-Live (TTL).	**20**

Table 26-1 Administrative Policy and Policy Rule Traffic Classifications (continued)

Traffic Classification	Description	Attribute ID
iptos	Classifies based on Type of Service field in IP packet.	**21**
ipproto	Classifies based on protocol field in IP packet.	**22**
icmp6type	Classifies based on ICMPv6 packet type.	**23**
ether	Classifies based on type field in Ethernet II packet.	**25**
llcDsapSsap	Classifies based on DSAP/SSAP pair in 802.3 type packet.	**26**
vlantag	Classifies based on VLAN tag.	**27**
tci	Classifies based on Tag Control Information.	**28**
application	Classifies based upon applications (llmnr, ssdp, or mdns-sd)	**29**
port	Classifies based on port-string.	**31**

A data value is associated with most traffic classifications to identify the specific network element for that classification. For data value and associated mask details, see the "Valid Values for Policy Classification Rules" table in the **set policy rule** command discussion of the command reference guide for your platform.

The following example enables TCI overwrite for policy profile **1**, followed by an example that enables TCI overwrite on port **ge.1.1**:

```
S Chassis(rw)->set policy profile 1 tci-overwrite enable
S Chassis(rw)->set port tcioverwrite ge.1.1 enable
```

Configuring Policy Role Traffic Classification Precedence

Each policy role has a precedence list associated with it that determines the order in which classification rules are applied to a packet. The higher the placement of the classification rule attribute in the list, the higher the precedence value of that attribute when applying classification rules.

All classification rule attributes supported by the platform have a static numeric ID value and are members of a precedence list. See Table 26-1 on page 26-8 for a listing of classification rule attributes and their associated attribute ID values in the default order of precedence.

Use the **show policy profile** command to display the current precedence list associated with a policy role.

By default, the precedence list is made up of attribute values 1-31 (with the exception that attribute value 29 is out of order), with unsupported ID values not specified. The precedence list associated with a given role can be modified using the **precedence** option in the **set policy profile** command. The following S-Series example sets the port (**31**) attribute to the highest precedence and leaves the remaining attributes in the default ordering:

```
S Chassis(rw)->set policy profile 200 precedence
31,1-2,29,3-8,12-19,21-22,25-28,9-11,23,20
S Chassis(rw)->show policy profile 200
Profile Index           :200
Profile Name            :
.
.
.
Rule Precedence         :31,1-2,29,3-8,12-19,21-22,25-28,9-11,23,20
```

```
                                :Port (31), MACSource (1), MACDest (2),
                                :Application (29), IPXSource (3), IPXDest (4),
                                :IPXSrcSocket (5), IPXDstSocket (6), IPXClass (7),
                                :IPXType (8), IPSource (12), IPDest (13), IPFrag (14),
                                :UDPSrcPort (15), UDPDestPort (16), TCPSrcPort (17),
                                :TCPDestPort (18), ICMPType (19), IPTOS (21),
                                :IPProto (22), Ether (25), LLCDSAPSSAP (26),
                                :VLANTag (27), TCI (28), IPv6Source (9),
                                :IPv6Dest (10), IPv6Flow (11), ICMP6Type (23),
                                :TTL (20)
.
.
.
S Chassis(rw)->
```

Policy Applications

A rule entry can classify the query or response from applications:

Link Local Mulitcast Name Resolution (**LLMNR**) – This protocol is based on the Domain Name System (DNS) packet format. It allows hosts to perform name resolution for hosts on the same local link. LLMNR is defined in RFC 4795.

Simple Service Discovery Protocol (**SSDP**) – SSDP is a Universal Plug-and-Play (UPnP) based protocol. SSDP uses the NOTIFY and MSEARCH HTTP methods to discover and advertise services on the network.

Multicast Domain Name System - Service Discovery (**MDNS-SD**) – DNS-SD is a service discovery protocol that utilizes the Domain Name System. Multicast DNS is a protocol that is mostly compatible with normal DNS, but uses link local multicast addressing, allowing for zeroconf functionality. MDNS-SD is primarily used with the Bonjour protocol.

Use the application classification rule attribute to create a policy application rule entry specifying the application and whether the classification is based upon query or announce for the data.

Specifying Storage Type

Specifying the storage type for a rule entry is supported. Storage types are **volatile** and **non-volatile**. Volatile storage does not persist after a reset of the device. Non-volatile storage does persist after a reset of the device. Use the **storage-type** option to specify the desired storage type for this policy rule entry.

Forward and Drop

Packets for this entry can be either forwarded or dropped for this traffic classification using the **forward** and **drop** policy rule options.

Allowed Traffic Rule-Type on a Port

Allowed traffic rule-type on a port provides for the setting, for each port, of the traffic classification rule-types that will be allowed or ignored in an admin-profile. By default, all traffic rule-types are allowed.

Use the **set policy allowed-type** command to configure a subset of traffic rule-types that will be allowed on the specified ports. All unspecified traffic rule-types will be disallowed. The **append** option provides for the addition of specified rule-types for the current subset of allowed rule-types. The **clear** option provides for setting the specified rule-types to disallowed.

Use the **show policy allowed-type** command to display a table of the current allowed and disallowed traffic rule-types for the specified port(s).

See Table 26-1 on page 26-8 for a listing of supported traffic classification rule-types. Use the attribute ID value, specified in Table 26-1, in the rule list for the **set policy allowed-type** command to identify the traffic classification to be added to or deleted from the allowed-type list for the specified ports.

The following example specifies that only traffic rule-type 1 (Source MAC Address) will be allowed for the admin-profile associated with port ge.1.5. All other rule-types will be ignored:

```
S Chassis(rw)->set policy allowed-type ge.1.5 traffic-rule 1
```

Policy Accounting

Policy accounting controls the collection of classification rule hits. If a hit occurs on a policy rule, policy accounting flags that the hit has occurred and will remain flagged until cleared. Policy accounting is enabled by default.

Policy accounting can be enabled or disabled using the **set policy accounting** command.

Policy Syslog Rule Usage

Policy syslog rule usage provides for the setting of rule usage message formatting to machine- or human-readable and sets the control for extended syslog message format.

Enabling the machine-readable option formats the rule usage messages in a raw data format that can then be parsed by a user-written scripting backend. This provides the enterprise with the ability to format the data in a manner that is most useful to the enterprise. Disabling the machine-readable option formats the same rule usage data in a human readable format.

Setting syslog rule usage to extended-format includes additional information in the rule usage syslog message. The data included in the extended format is as follows: VLAN, COS assigned, and the following fields found in the packet: DEST MAC, SRC MAC, TAG(8100:tci), Ether Type, SIP(ip), DIP(ip), Protocol, TOS/DSCP, Fragmentation indication, Destination PORT, and Source Port.

Use the **set policy syslog** command to set syslog rule usage configuration.

Quality of Service in a Policy Rules Context

Quality of Service (QoS) can be specified directly in a policy role as stated in "Assigning a Class of Service to this Role" on page 26-4. A CoS can also be applied to a policy rule. The CoS specified at the policy role level is the default and is only used if no rule is triggered. Therefore, if a CoS is applied to both the policy role and a policy rule, the CoS specified in the policy rule takes precedence over the CoS in the policy role for the traffic classification context specified in the policy rule. As stated in the policy role discussion, CoS configuration details are beyond the scope of this document. See Chapter 53, **Quality of Service (QoS) Configuration** for a complete discussion of QoS configuration.

The following example applies CoS 8 to profile-index 1 for port ge.1.1:

```
S Chassis(rw)->set policy rule 1 port ge.1.1 port-string ge.1.1 cos 8
```

Disabling an Ingress Port Per Policy Rule

A policy rule can be set to disable an ingress port, if a hit occurs for that rule, using the **disable-port** option of the **set policy rule** command. This per policy rule disable-port feature can be set to:

- **enabled** - The ingress port is disabled with this rule use

- **disabled** - The ingress port is not disabled with this rule use
- **prohibit** - Prohibits lower precedence rules from disabling the ingress port with this rule use

To disable a port for the first use of any policy profile rule, see "Disabling an Ingress Port on First Profile Rule Use" on page 26-7.

Use the **clear policy disabled-ports** to clear ports from the disabled state due to a policy rule hit on those ports.

Use the **show policy disabled-ports** command to display ports that have been disabled due to first profile rule use.

Blocking Non-Edge Protocols at the Edge Network Layer

Edge clients should be prevented from acting as servers for a number of IP services. If non-edge IP services accidently or maliciously attach to the edge of the network, they are capable of disrupting network operation. IP services should only be allowed where and when your network design requires. This section identifies ten IP Services you should consider blocking at the edge unless allowing them is part of your network architecture. See "Assigning Traffic Classification Rules" on page 26-25 for an example of how to configure a subset of these recommended IP services to drop traffic at the edge.

Table 26-2 Non-Edge Protocols

Protocol	Policy Effect
DHCP Server Protocol	Every network needs DHCP. Automatically mitigate the accidental or malicious connection of a DHCP server to the edge of your network to prevent DoS or data integrity issues, by blocking DHCP on the source port for this device.
DNS Server Protocol	DNS is critical to network operations. Automatically protect your name servers from malicious attack or unauthorized spoofing and redirection, by blocking DNS on the source port for this device.
Routing Topology Protocols	RIP, OSPF, and BGP topology protocols should only originate from authorized router connection points to ensure reliable network operations.
Router Source MAC and Router Source IP Address	Routers and default gateways should not be moving around your network without approved change processes being authorized. Prevent DoS, spoofing, data integrity and other router security issues by blocking router source MAC and router source IP addresses at the edge.
SMTP/POP Server Protocols	Prevent data theft and worm propagation by blocking SMTP at the edge.
SNMP Protocol	Only approved management stations or management data collection points need to be speaking SNMP. Prevent unauthorized users from using SNMP to view, read, or write management information.
FTP and TFTP Server Protocols	Ensure file transfers and firmware upgrades are only originating from authorized file and configuration management servers.
Web Server Protocol	Stop malicious proxies and application-layer attacks by ensuring only the right Web servers can connect from the right location at the right time, by blocking HTTP on the source port for this device.

Table 26-2 Non-Edge Protocols (continued)

Protocol	Policy Effect
Legacy Protocols	If IPX, AppleTalk, DECnet or other protocols should no longer be running on your network, prevent clients from using them. Some organizations even take the approach that unless a protocol is specifically allowed, all others are denied.

Policy Capabilities

Table 26-3 provides a listing of policy capabilities.

Table 26-3 Traffic Classification Based Policy Capabilities

Traffic Classification	Description
Dynamic PID Assign Rule	The ability to dynamically assign a policy based upon a traffic classification.
Admin PID Assign Rule	The ability to administratively assign a policy based upon a traffic classification.
VLAN Forwarding	The ability to assign a forwarding VLAN rule.
Deny	The ability to assign a drop traffic rule.
Permit	The ability to assign a forward traffic rule.
CoS Assign Rule	The ability to assign a CoS rule.
Priority	The ability to assign traffic priority using a CoS assignment.
Destination Mirror	The ability to apply a destination mirror to this rule.
Clear Mirror	The ability to clear mirroring on this rule.
Prohibit Mirror	The ability to prohibit mirroring on this rule.
Longest Prefix Rules	The ability to always look at the highest bit mask for an exact traffic classification match.
VLAN Assign Rule	The ability to assign rules based upon the ingress VLAN. (TCI overwrite must be enabled).
TCI Overwrite	The ability to overwrite user priority and other VLAN tag TCI field classification information.
Rule-Use Accounting	The ability to enable policy accounting.
Rule-Use Notification	The ability to enable syslog and traps for rule hit notification.
Invalid Policy Action	The ability to set a drop, forward, or default-policy behavior based upon an invalid action.
Port Disable Action	The ability to disable a port upon first rule hit.
Precedence Reordering	The ability to reorder traffic classification precedence for a policy role.

Captive Portal Redirection

Captive Portal Redirection uses HTTP redirection to force a client's web browser to be redirected to a particular administrative web page. A network administrator can use this feature for such web based contexts as redirecting to a page for purposes of:

- Authentication – Request a user login and password
- Payment – For example, in the context of an airport hotspot
- Use-Policy enforcement – Require installation of additional software or agree to a Terms or Service (ToS)

Captive Portal Redirection is an extension of the Policy feature. Policy roles can be configured to force redirection of HTTP traffic by specifying a web redirection class index which associates with it up to two redirection servers. The HTTP traffic to potentially be redirected is identified based upon a destination captive portal server absolute URL address containing an IPv4 address, TCP port, and path. For traffic that is placed into one of these policy roles (through authentication or policy admin-profile rules) actions will be taken based upon the contents of the policy profile. If the incoming traffic is on the configured L4 port and is not destined for the configured captive portal server IP, the switch will cause an HTTP redirect message (code 307) to be sent back to the client. If the incoming traffic is destined for the configured captive portal server IP, or it is not on one of the configured listening L4 ports, it will be handled according to the rest of the policy role configuration.

When configuring Captive Portal Redirection, the administrator can globally configure up to three ports on which policy on this device listens for client traffic that may be subject to HTTP redirection. Captive Portal Redirection provides for the configuration of up to ten web-redirect groups of captive portal servers. These web-redirect groups are identified by associating a web redirection class index with the server ID. A URL that explicitly identifies the server by an IPv4 address, TCP port, and path is configured along with the ports on which the Captive Portal feature listens for client traffic.

The Captive Portal Redirection policy roles are configured for a web-redirect group. By default, the policy profile web-redirect index is unset and has a numeric value of 0. To enable Captive Portal Redirection there must be a policy role (profile) defined that has a valid captive portal web redirection class index value of 1 – 10. In addition to the captive portal configuration, this policy role should also have rules to handle the traffic that would not be handled by Captive Portal Redirection.

The URL for the captive portal server must:

- Begin with "**http://**"
- Specify an IPv4 server IP address and TCP port delineated by a colon (**:**)
- Specify a path (/)

In the following example:

- The S-Series is configured to listen for web traffic on ports **80** and **8080**
- The web-redirect index **1** enables server **1** with a URL of **http://10.52.3.101:80/static/index.jsp**
- The web-redirect index **1** enables server **2** with a URL of **http://10.52.3.102:80/static/index.jsp**
- The web-redirect index 1 is associated with policy profile **1**

```
S Chassis(rw)->set policy captive-portal listening 80,8080

S Chassis(rw)->set policy captive-portal web-redirect 1 server 1 url
http://10.52.3.101:80/static/index.jsp status enable

S Chassis(rw)->set policy captive-portal web-redirect 1 server 2 url
http://10.52.3.102:80/static/index.jsp status enable

S Chassis(rw)->set policy profile 1 web-redirect 1

S Chassis(rw)->show policy captive-portal listening

Captive Portal Listening Ports: 80   8080

S Chassis(rw)->show policy captive-portal web-redirect 1

Web-redirect Index: 1
```

```
Server Index: 1
  Server Status:  Enabled
  Server URL:     http://10.52.3.101:80/static/index.jsp

Server Index: 2
  Server Status:  Enabled
  Server URL:     http://10.52.3.102:80/static/index.jsp
```

Configuring Policy

This section presents configuration procedures and tables including command description and syntax in the following policy areas: profile, classification, and display.

Procedure 26-1 describes how to configure policy roles and related functionality.

Procedure 26-1 Configuring Policy Roles

Step	Task	Command(s)
1.	In any command mode, create a policy role. • **name** – (Optional) Specifies a name for this policy profile; used by the filter-ID attribute. This is a string from 1 to 64 characters. • **pvid-status** – (Optional) Enables or disables PVID override for this policy profile. If all the classification rules associated with this profile are missed, then this parameter, if specified, determines the default VLAN for this profile. • **pvid** – (Optional) Specifies the PVID to assign to packets, if PVID override is enabled and invoked as the default behavior. • **cos-status** – (Optional) Enables or disables Class of Service override for this policy profile. If all the classification rules associated with this profile are missed, then this parameter, if specified, determines the default CoS assignment.	**set policy profile** *profile-index* [**name** *name*] [**pvid-status** {**enable** | **disable**}] [**pvid** *pvid*] [**cos-status** {**enable** | **disable**}] [**cos** *cos*] [**egress-vlans** *egress-vlans*] [**forbidden-vlans** *forbidden-vlans*] [**untagged-vlans** *untagged-vlans*] [**append**] [**clear**] [**tci-overwrite** {**enable** | **disable**}] [**precedence** *precedence-list*] [**mirror-destination** <mirror-index>] | [**clear-mirror**] | [**prohibit-mirror**][**syslog** {**enable** | **disable**}] [**trap** {**enable** | **disable**}] [**disable-port** {**enable** | **disable**}] [**fst** *class-index*] [**web-redirect** *redirect-index*]

Procedure 26-1 Configuring Policy Roles (continued)

Step	Task	Command(s)
	• **cos** – (Optional) Specifies a CoS value to assign to packets, if CoS override is enabled and invoked as the default behavior. Valid values are 0 to 255. • **egress-vlans** – (Optional) Specifies the port to which this policy profile is applied should be added to the egress list of the VLANs defined by egress-vlans. Packets will be formatted as tagged. • **forbidden-vlans** – (Optional) Specifies the port to which this policy profile is applied should be added as forbidden to the egress list of the VLANs defined by forbidden-vlans. Packets from this port will not be allowed to participate in the listed VLANs. • **untagged-vlans** – (Optional) Specifies the port to which this policy profile is applied should be added to the egress list of the VLANs defined by untagged-vlans. Packets will be formatted as untagged. • **append** – (Optional) Appends any egress, forbidden, or untagged specified VLANs to the existing list. If append is not specified, all previous settings for this VLAN list are replaced • **clear** – (Optional) Clears any egress, forbidden or untagged VLANs specified from the existing list. • **tci-overwrite** – (Optional) Enables or disables TCI (Tag Control Information) overwrite for this profile. When enabled, rules configured for this profile are allowed to overwrite user priority and other classification information in the VLAN tag's TCI field. If this parameter is used in a profile, TCI overwrite must be enabled on ports. See Step 3 below. • **precedence** – (Optional) Assigns a rule precedence to this profile. Lower values will be given higher precedence. • **mirror-destination** – (Optional) Applies the specified mirror destination index to this profile. • **clear-mirror** – (Optional) Clears mirroring on this profile. • **prohibit-mirror** – (Optional) Prohibits mirroring on this profile.	

Procedure 26-1 Configuring Policy Roles (continued)

Step	Task	Command(s)
	• **syslog** – (Optional) Enables or disables syslog on this profile. • **trap** – (Optional) Enables or disables traps on this profile. • **disable-port** – (Optional) Enable or disables the disabling of ingress ports on profile use. • **fst** – (Optional) Specifies a flow limit class to apply to this profile. • **web-redirect** – (Optional) Specifies a web-redirect class index associated with this profile	
2.	(Optional) Assign the action the device will apply to an invalid or unknown policy. • **default-policy** – Instructs the device to ignore this result and search for the next policy assignment rule. • **drop** – Instructs the device to block traffic. • **forward** – Instructs the device to forward traffic.	**set policy invalid action** {**default-policy** | **drop** | **forward**}
3.	(Optional) Enable or disable the TCI overwrite function on one or more ports.	**set port tcioverwrite** *port-string* {**enable** | **disable**}
4.	(Optional) Enable or disable policy accounting, which flags classification rule hits.	**set policy accounting** {**enable** | **disable**}
5.	(Optional) Set the rule usage and extended format syslog policy settings. • **machine-readable** - (Optional) Sets the formatting of rule usage messages to raw data that a user script can format according to the needs of the enterprise, otherwise message is set to human readable. • **extended-format** - (Optional) Sets the control to include additional information in the rule usage syslog messages, otherwise the original rule usage syslog message format is used.	**set policy syslog** [**machine-readable** {**enable** | **disable**}] [**extended-format** {**enable** | **disable**}]
6.	(Optional) Set a policy maptable entry that associates a VLAN with a policy profile. This option is also supported by the B3, C3, and G3 for releases 6.3 and greater.	**set policy maptable** {*vlan-list profile-index* }
7.	Optionally, set a policy maptable response. • **tunnel** - Applies the VLAN tunnel attribute. • **policy** - Applies the policy specified in the filter-ID. • **both** - Applies either or all the filter-ID and VLAN tunnel attributes or the policy depending upon whether one or both are present.	**set policy maptable response** {**tunnel** | **policy** | **both**}

Procedure 26-1 Configuring Policy Roles (continued)

Step	Task	Command(s)
8.	Optionally, set up to three Captive Portal Redirection listening ports.	**set policy captive-portal listening** *port-list*
9.	Optionally, enable a web-redirect class index specifying the server index and an absolute URL to the server including the TCP port.	**set policy captive-portal web-redirect** *web-red-index* **server** *sever-index* **url http://***server-ip-address***:***tcp-port***/***path* **status** {**enable** | **disable**}

Procedure 26-2 describes how to configure classification rules as an administrative profile or to assign policy rules to a policy role.

Procedure 26-2 Configuring Classification Rules

Step	Task	Command(s)
1.	In any command mode, optionally set an administrative profile to assign traffic classifications to a policy role. See Table 26-1 on page 26-8 for traffic classification-type descriptions. See the **set policy rule** command discussion in the command reference guide that comes with your device for traffic classification data and mask information. • **port-string** - Applies this administratively-assigned rule to a specific ingress port. S-Series devices with firmware versions 3.00.xx and higher also support the **set policy port** command as an alternative to administratively assign a profile rule to a port. • **storage-type** - (Optional) Adds or removes this entry from non-volatile storage. • **admin-pid** - Associates this administrative profile with a policy profile index ID. Valid values are 1 - 1023. • **syslog** - (Optional) Enables or disables sending of syslog messages on first rule use. • **trap** - (Optional) Enables or disables sending SNMP trap messages on first rule use. • **disable-port** - (Optional) Enables or disables the ability to disable the ingress port on first rule use. • **mirror-destination** - (Optional) Applies the specified mirror destination index to this profile. • **clear-mirror** - (Optional) Clears mirroring on this profile. • **prohibit-mirror** - (Optional) Prohibits mirroring on this profile.	**set policy rule admin-profile** *classification-type* [*data*] [**mask** *mask*] [**port-string** *port-string*] [**storage-type** {**non-volatile** | **volatile**}] [**admin-pid** *admin-pid*] [**syslog** {**enable** | **disable** | **prohibit**}] [**trap** {**enable** | **disable** | **prohibit**}] [**disable-port** {**enable** | **disable** | **prohibit**}] [**tci-overwrite** {**enable** | **disable** | **prohibit**}] [**mirror-destination** <mirror-index>] | **clear-mirror**] | [**prohibit-mirror**]

Procedure 26-2 Configuring Classification Rules (continued)

Step	Task	Command(s)
2.	In any command mode, optionally configure policy rules to associate with a policy role. See Table 26-1 on page 26-8 for traffic classification-type descriptions. See the **set policy rule** command discussion in the command reference guide that comes with your device for traffic classification data and mask information. • **port-string** - (Optional) Applies this policy rule to a specific ingress port. S-Series devices with firmware versions 3.00.xx and higher also support the **set policy port** command as an alternative way to assign a profile rule to a port. • **storage-type** - (Optional) Adds or removes this entry from non-volatile storage. • **vlan** - (Optional) Classifies this rule to a VLAN ID. • **drop** | **forward** - (Optional) Specifies that packets within this classification will be dropped or forwarded. • **cos** - (Optional) Specifies that this rule will classify to a Class-of-Service ID. Valid values are 0 - 255. A value of -1 indicates that no CoS forwarding behavior modification is desired. • **syslog** - (Optional) Enables or disables sending of syslog messages on first rule use. • **trap** - (Optional) Enables or disables sending SNMP trap messages on first rule use. • **disable-port** - (Optional) Enables or disables the ability to disable the ingress port on first rule use. • **mirror-destination** - (Optional) Applies the specified mirror destination index to this profile. • **clear-mirror** - (Optional) Clears mirroring on this profile. • **prohibit-mirror** - (Optional) Prohibits mirroring on this profile.	**set policy rule** *profile-index classification-type* [*data*] [**mask** *mask*] [**port-string** *port-string*] [**storage-type** {**non-volatile** | **volatile**}] [**vlan** *vlan*] | [**drop** | **forward**] [**admin-pid** *admin-pid*] [**cos** *cos*] [**syslog** {**enable** | **disable**}] [**trap** {**enable** | **disable**}] [**disable-port** {**enable** | **disable**}] [**mirror-destination** <mirror-index>] | [**clear-mirror**] | [**prohibit-mirror**] [**quarantine-profile** *quarantine-profile*] [**clear-quarantine-profile**] [**prohibit-quarantine-profile**] **clear-mirror**] | [**prohibit-mirror**]
3.	(Optional) Assigns a policy role to a port.	**set policy port** *port-name admin-id*
4.	(Optional) Assigns a list of allowed traffic rules that can be applied to the admin profile for one or more ports.	**set policy allowed-type** *port-string* **traffic-rule** *rule-list* [**append** | **clear**]
5.	(Optional) Enable or disable the the ability to clear rule usage information if operational status “up” is detected on any port.	**set policy autoclear** {[**enable** | **disable**] [*interval interval*] [**profile** {**enable** | **disable**}] [**ports** *port-list* [**append** | **clear**]]}
6.	(Optional) Set the status of dynamically assigned policy role options.	**set policy dynamic** [**syslog-default** {**enable** | **disable**}] [**trap-default** {**enable** | **disable**}]}

Table 26-4 describes how to display policy information and statistics.

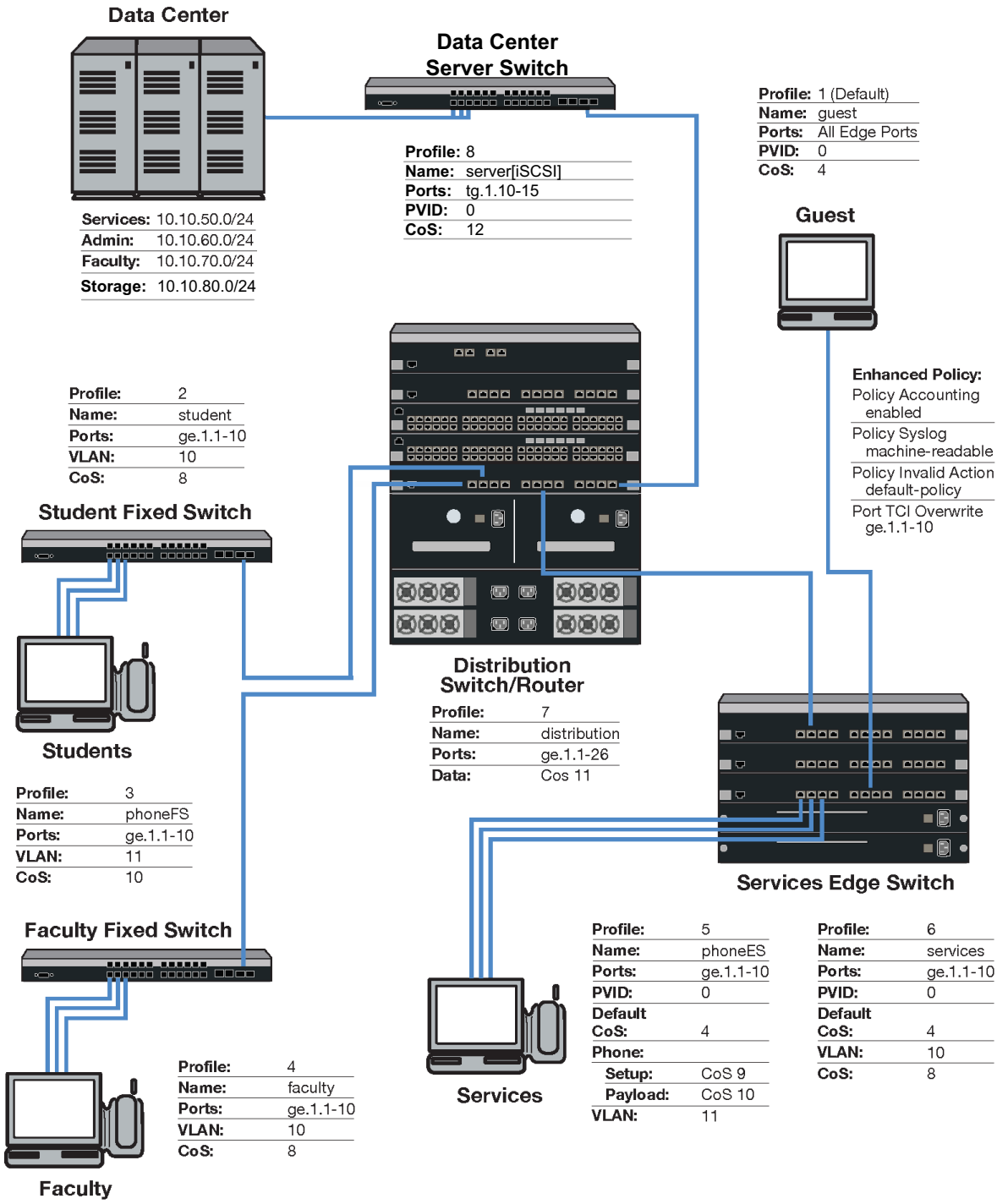
Table 26-4 Displaying Policy Configuration and Statistics

Task	Command(s)
In any command mode, display policy role information.	**show policy profile** {**all** | *profile-index* [*consecutive-pids*] [**-verbose**]}
In any command mode, display the action the device should take if asked to apply an invalid or unknown policy, or the number of times the device has detected an invalid/unknown policy, or both action and count information.	**show policy invalid** {**all** | **action** | **count**}
In any command mode, display the current control status of the collection of rule usage statistics.	**show policy accounting**
In any command mode, display syslog parameters for policy rule entries.	**show policy syslog** [**machine-readable**] [**extended-format**]
In any command mode, display VLAN-ID to policy role mappings table.	**show policy maptable** *vlan-list*
In any command mode, display TCI overwrite tag control information on one or more ports.	**show port tcioverwrite** [*port-string*]
In any command mode, display policy classification and admin rule information.	**show policy rule** *classification-type* [*data*] [**mask** *mask*] [**port-string** *port-string*] [**rule-status** {**active** | **not-in-service** | **not-ready**}] [**storage-type** {**non-volatile** | **volatile**}] [**vlan** *vlan*] | [**drop** | **forward**] [**dynamic-pid** *dynamic-pid*] [**cos** *cos*] [**admin-pid** *admin-pid*] [**syslog** {**enable** | **disable** | **prohibit**}] [-verbose] [**trap** {**enable** | **disable** | **prohibit**}] [**disable-port** {**enable** | **disable** | **prohibit**}] [**usage-list**] [**display-if-used** *port-list*] [**tci-overwrite** {**enable** | **disable** | **prohibit**}] [**mirror-destination** *mirror-index*] | [**clear-mirror**] | [**prohibit-mirror**] [**-verbose**] [**-wide**]
In any command mode, display all policy classification capabilities for this device.	**show policy capability**
In any command mode, display a list of currently supported traffic rules applied to the administrative profile for one or more ports.	**show policy allowed-type** *port-string* [**-verbose**]
In any command mode, display a count of the number of times the device has dropped syslog or trap rule usage notifications on ports.	**show policy dropped-notify**
In any command mode, display disabled ports for all rule entries.	**show policy disabled-ports**
In any command mode, display the current state of the autoclear feature.	**show policy autoclear** {**all** | **link** | **interval** | **profile** | **ports**}
In any command mode, display status of dynamically assigned roles.	**show policy dynamic** {[**syslog-default**] [**trap-default**] [**override**]}

Policy Configuration Example

This section presents a college-based policy configuration example. Figure 26-1 displays an overview of the policy configuration. This overview display is followed by a complete discussion of the configuration example.

Figure 26-1 College-Based Policy Configuration



Note: For purposes of this discussion, Edge Switch and Distribution Switch refer to S-Series platforms, and the Data Center Server Switch refers to a 7100-Series switch.

Roles

The example defines the following roles:

- **guest** – Used as the default policy for all unauthenticated ports. Connects a PC to the network providing internet only access to the network. Provides guest access to a limited number of the edge switch ports to be used specifically for internet only access. Policy is applied using the port level default configuration, or by authentication, in the case of the Services Edge Switch port internet only access PCs.
- **student** – Connects a dorm room PC to the network through a "Student" Fixed Switch port. A configured CoS rate limits the PC. Configured rules deny access to administrative and faculty servers. The PC authenticates using RADIUS. Hybrid authentication is enabled. The **student** policy role is applied using the filter-ID attribute. The base VLAN is applied using the tunnel attributes returned in the RADIUS response message. If all rules are missed, the settings configured in the **student** policy profile are applied.
- **phoneFS** – Connects a dorm room or faculty office VoIP phone to the network using a stackable fixed switch port. A configured CoS rate limits the phone and applies a high priority. The phone authenticates using RADIUS. Hybrid authentication is enabled. Policy is applied using the filter-ID returned in the RADIUS response message. The base VLAN is applied using the tunnel attributes returned in the RADIUS response message. If all rules are missed, the settings configured in the phoneFS policy profile are applied.
- **faculty** – Connects a faculty office PC to the network through a "Faculty" Fixed Switch port. A configured CoS rate limits the PC. A configured rule denies access to the administrative servers. The PC authenticates using RADIUS. Hybrid authentication is enabled. The **faculty** policy role is applied using the filter-ID attribute. The base VLAN is applied using the tunnel attributes returned in the RADIUS response message for the authenticating user. If all rules are missed, the settings configured in the **faculty** policy profile are applied.
- **phoneES** – Connects a services VoIP phone to the network using a Services Edge Switch port. A configured CoS rate limits the phone for both setup and payload, and applies a high priority. The phone authenticates using RADIUS. Tunnel authentication is enabled. The base VLAN is applied using the tunnel attributes returned in the RADIUS response message. Policy is applied using a maptable configuration. If all rules are missed, the settings configured in the **phoneES** policy profile are applied.
- **services** – Connects a services PC to the network through the Services Edge Switch port. A configured CoS rate limits the PC. Services are denied access to both the student and faculty servers. The PC authenticates using RADIUS. The base VLAN is applied using the tunnel attributes returned in the RADIUS response message for the authenticating user. The **services** policy role is applied using a policy maptable setting. The policy accounting, syslog, invalid action and TCI overwrite are enabled for this role. If all rules are missed, the settings configured in the **services** policy profile are applied.
- **distribution** – The Distribution policy role is applied at the Distribution Switch providing rate limiting.
- **server[iSCSI]** – The Server iSCSI policy role is applied to the Data Center Server Switch which provides low latency, high speed switching between the storage servers and the other servers in the data center.

Policy Domains

It is useful to break up policy implementation into logical domains for ease of understanding and configuration. For this example, it is useful to consider five domains: basic edge, standard edge on the Fixed Switch, premium edge on the Services Edge Switch, premium distribution on the Distribution Switch, and data center on the Data Center Server Switch.

Basic Edge

Protocols not appropriate to the edge should be blocked. For this example we will block DHCP, DNS, SNMP, SSH, Telnet and FTP at the edge on the data VLAN. We will forward destination port DHCP and DNS and source port for IP address request to facilitate auto configuration and IP address assignment. See "Blocking Non-Edge Protocols at the Edge Network Layer" on page 26-12 for a listing of protocols you should consider blocking at the edge.

Standard Edge

Edge Switch platforms will be rate-limited using a configured CoS that will be applied to the student and faculty, and phoneFS policy roles. Fixed Switch support for hybrid authentication depends upon the platform and firmware release. The Fixed Switch in this example supports the hybrid authentication capability. Hybrid authentication will be enabled.

Premium Edge

The Edge Switch will be rate-limited using a configured CoS that is applied to the services and phoneES policy role. This premium edge platform will be enabled for the following capabilities:

- Policy Accounting
- Syslog rule usage enabled and set to machine-readable
- Invalid policy action set to drop
- TCI overwrite enabled

Premium Distribution

The Distribution Switch Router will be rate-limited using a configured CoS. Premium distribution will be enabled for the following policy capabilities:

- Policy Accounting
- Syslog Rule Usage enabled and set to machine-readable
- Invalid policy action set to drop
- TCI overwrite enabled

Data Center

The Data Center Server Switch will provide policy for ports connecting iSCSI storage nodes with the other data center servers. This policy will allow forwarding of all TCP traffic on the iSCSI port 3260 with a CoS that provides low latency and high speed. It will also provide a bilateral set of rules that allow administrators to SSH to the switch on TCP port 22 and a destination rule to allow the node to SSH to another device.

Platform Configuration

This section will provide the CLI based policy configuration on the following platforms:

- Student Fixed Switch
- Faculty Fixed Switch
- Services Edge Switch
- Distribution Switch
- Data Center Server Switch

In CLI mode, configuration takes place on each platform. When using the NetSight Policy Manager, configuration takes place at a central location and is pushed out to the appropriate network devices.

For this configuration example, CoS related configuration will be specified as a final CoS. For details on configuring CoS, see "*Understanding QoS Configuration on the S-Series*" on page 53-10.

Note: CLI command prompts used in this configuration example have the following meaning:

- Extreme Networks(rw)-> – Input on all platforms used in this example.
- Fixed Switch(rw)-> – Input on all Fixed Switches.
- StudentFS-> – Input on the student Fixed Switch.
- FacultyFS-> – Input on the faculty Fixed Switch.
- Services(rw)-> – Input on the services S-Series device.
- Distribution(rw)-> – Input on the distribution S-Series device.
- iSCSI(wr)-> – Input on the data center 7100-Series device.

Configuring Guest Policy on Edge Platforms

All edge ports will be set with a default **guest** policy using the **set policy port** command. This guest policy provides for an internet only access to the network. Users on all ports will attempt to authenticate. If the authentication succeeds, the policy returned by authentication or, in the case of the Services Edge Switch configuration, the maptable setting, overrides the default port policy setting. If authentication fails, the guest policy is used. On the Services Edge Switch , five ports are used by PCs at locations throughout the campus, such as the library, to provide access to the internet. The PCs attached to these five ports will authenticate with the **guest** policy role. Public facing services would also be configured for guest status in a school or enterprise scenario. Public facing services are not part of this example.

Configuring the Policy Role

The guest role is configured with:

- A profile-index value of **1**
- A name of **guest**
- A PVID set to **0**
- A CoS set to **4**

Create the guest policy profile on all platforms:

```
Extreme Networks(rw)->set policy profile 1 name guest pvid-status enable pvid 0
cos-status enable cos 4
```

Assigning Traffic Classification Rules

For cases where discovery must take place to assign an IP address, DNS and DHCP traffic must be allowed. Forwarding of traffic is allowed on UDP source port 68 (IP address request) and UDP destination ports 53 (DNS) and 67 (DHCP).

```
Extreme Networks(rw)->set policy rule 1 udpsourceport 68 mask 16 forward
Extreme Networks(rw)->set policy rule 1 udpdestportIP 53 mask 16 forward
Extreme Networks(rw)->set policy rule 1 udpdestportIP 67 mask 16 forward
```

Guest policy allows internet traffic. TCP destination Ports 80, 8080, and 443 will be allowed traffic forwarding.

```
Extreme Networks(rw)->set policy rule 1 tcpdestportIP 80 mask 16 forward
Extreme Networks(rw)->set policy rule 1 tcpdestportIP 443 mask 16 forward
Extreme Networks(rw)->set policy rule 1 tcpdestport 8080 mask 16 forward
```

ARP forwarding is required on ether port 0x806.

```
Extreme Networks(rw)->set policy rule 1 ether 0x806 mask 16 forward
```

Assigning the Guest Policy Profile to All Edge Ports

Assign the guest policy profile to all Fixed Switch and Services Edge Switch ports.

```
Extreme Networks(rw)->set policy port ge.*.1-47 1
```

Configuring Policy for the Edge Student Fixed Switch

Configuring the Policy Role

The student role is configured with:

- A profile-index value of **2**
- A name of **student**
- A port VLAN of **10**
- A CoS of **8**

Create a policy role that applies a CoS 8 to data VLAN 10 and configures it to rate-limit traffic to 1M with a moderate priority of 5.

```
StudentFS(rw)->set policy profile 2 name student pvid-status enable pvid 10
cos-status enable cos 8
```

Assigning Hybrid Authentication

Configure the RADIUS server user accounts with the appropriate tunnel information using VLAN authorization and policy filter-ID for student role members and devices. Enable hybrid authentication, allowing the switch to use both the filter-ID and tunnel attributes in the RADIUS response message. Set a VLAN-to-policy mapping as backup in case the response does not include the RADIUS filter-ID attribute. This mapping is ignored if RADIUS filter-ID attribute is present in the RADIUS response message.

```
StudentFS(rw)->set policy maptable response both
StudentFS(rw)->set policy maptable 10 2
```

Assigning Traffic Classification Rules

Forward traffic on UDP source port for IP address request (68), and UDP destination ports for protocols DHCP (67) and DNS (53). Drop traffic on UDP source ports for protocols DHCP (67) and DNS (53). Drop traffic for protocols SNMP (161), SSH (22), Telnet (23) and FTP (20 and 21) on both the data and phone VLANs.

```
StudentFS(rw)->set policy rule 2 udpsourceport 68 mask 16 forward
StudentFS(rw)->set policy rule 2 udpdestport 67 mask 16 forward
StudentFS(rw)->set policy rule 2 udpdestport 53 mask 16 forward
StudentFS(rw)->set policy rule 2 udpsourceportIP 67 mask 16 drop
StudentFS(rw)->set policy rule 2 udpsourceportIP 53 mask 16 drop
StudentFS(rw)->set policy rule 2 udpdestportIP 16 mask 16 drop
StudentFS(rw)->set policy rule 2 tcpdestportIP 22 mask 16 drop
StudentFS(rw)->set policy rule 2 tcpdestportIP 23 mask 16 drop
StudentFS(rw)->set policy rule 2 tcpdestportIP 20 mask 16 drop
StudentFS(rw)->set policy rule 2 tcpdestportIP 21 mask 16 drop
```

Students should only be allowed access to the services server (subnet 10.10.50.0/24) and should be denied access to both the administrative (subnet 10.10.60.0/24) and faculty servers (subnet 10.10.70.0/24).

```
StudentFS(rw)->set policy rule 2 ipdestsocket 10.10.60.0 mask 24 drop
StudentFS(rw)->set policy rule 2 ipdestsocket 10.10.70.0 mask 24 drop
```

Configuring PhoneFS Policy for the Edge Fixed Switch

Configuring the Policy Role

The phoneFS role is configured on both the dorm room and faculty office Fixed Switches with:

- A profile-index of **3**
- A name of **phoneFS**
- A port VLAN of 11
- A CoS of 10

Because we can not apply separate rate limits to the phone setup and payload ports on the Fixed Switch using policy rules, apply CoS 10 with the higher payload appropriate rate limit of 100k bps and a high priority of 6 to the phoneFS role.

```
Fixed Switch(rw)->set policy profile 3 name phoneFS  pvid-status enable pvid 11
cos-status enable cos 10
```

Assigning Traffic Classification Rules

Drop traffic for protocols SNMP (161), SSH (22), Telnet (23) and FTP (20 and 21) on the phone VLAN. Forward traffic on UDP source port for IP address request (68) and forward traffic on UDP destination ports for protocols DHCP (67) and DNS (53) on the phone VLAN, to facilitate phone auto configuration and IP address assignment.

```
Fixed Switch(rw)->set policy rule 3 udpdestportIP 161 mask 16 drop
Fixed Switch(rw)->set policy rule 3 tcpdestportIP 22 mask 16 drop
Fixed Switch(rw)->set policy rule 3 tcpdestportIP 23 mask 16 drop
Fixed Switch(rw)->set policy rule 3 tcpdestportIP 20 mask 16 drop
Fixed Switch(rw)->set policy rule 3 tcpdestportIP 21 mask 16 drop
Fixed Switch(rw)->set policy rule 3 udpsourceport 68 mask 16 forward
Fixed Switch(rw)->set policy rule 3 udpdestportIP 67 mask 16 forward
Fixed Switch(rw)->set policy rule 3 udpdestportIP 53 mask 16 forward
```

Assigning Hybrid Authentication

Configure the RADIUS server user accounts with the appropriate tunnel information using VLAN authorization and policy filter-ID for phoneFS role members and devices. Enable hybrid authentication, allowing the switch to use both the filter-ID and tunnel attributes in the RADIUS response message. Set a VLAN-to-policy mapping as backup in case the response does not include the RADIUS filter-ID attribute. This mapping is ignored if RADIUS filter-ID attribute is present in the RADIUS response message.

```
Fixed Switch(rw)->set policy maptable response both
Fixed Switch(rw)->set policy maptable 11 3
```

Configuring Policy for the Edge Faculty Fixed Switch

Configuring the Policy Role

The faculty role is configured with:

- A profile-index value of **4**
- A name of **faculty**
- A port VLAN of **10**
- A CoS of **8**

Create a policy role that applies a CoS 8 to data VLAN 10 and configures it to rate-limit traffic to 1M with a moderate priority of 5.

```
FacultyFS(rw)->set policy profile 4 name faculty pvid-status enable pvid 10
cos-status enable cos 8
```

Assigning Hybrid Authentication

Configure the RADIUS server user accounts with the appropriate tunnel information using VLAN authorization and policy filter-ID for faculty role members and devices. Enable hybrid authentication. Set a VLAN-to-policy mapping. This mapping is ignored if the RADIUS filter-ID attribute is present in the RADIUS response message.

```
StudentFS(rw)->set policy maptable response both
StudentFS(rw)->set policy maptable 10 4
```

Assigning Traffic Classification Rules

Forward traffic on UDP source port for IP address request (68), and UDP destination ports for protocols DHCP (67) and DNS (53). Drop traffic on UDP source ports for protocols DHCP (67) and DNS (53). Drop traffic for protocols SNMP (161), SSH (22), Telnet (23) and FTP (20 and 21) on both the data and phone VLANs.

```
FacultyFS(rw)->set policy rule 4 udpsourceport 68 mask 16 forward
FacultyFS(rw)->set policy rule 4 udpdestport 67 mask 16 forward
FacultyFS(rw)->set policy rule 4 udpdestport 53 mask 16 forward
FacultyFS(rw)->set policy rule 4 udpsourceportIP 67 mask 16 drop
FacultyFS(rw)->set policy rule 4 udpsourceportIP 53 mask 16 drop
FacultyFS(rw)->set policy rule 4 udpdestportIP 16 mask 16 drop
FacultyFS(rw)->set policy rule 4 tcpdestportIP 22 mask 16 drop
FacultyFS(rw)->set policy rule 4 tcpdestportIP 23 mask 16 drop
FacultyFS(rw)->set policy rule 4 tcpdestportIP 20 mask 16 drop
FacultyFS(rw)->set policy rule 4 tcpdestportIP 21 mask 16 drop
```

Faculty should only be allowed access to the services (subnet 10.10.50.0/24) and the faculty servers (subnet 10.10.70.0/24) and should be denied access to the administrative server (subnet 10.10.60.0/24).

FacultyFS(rw)->set policy rule 4 ipdestsocket 10.10.60.0 mask 24 drop

Configuring PhoneES Policy for the Services Edge Switch

Configuring the Policy Role

The phoneES role is configured on the Services Edge Switch with:

- A profile-index of **5**

- A name of **phoneES**
- A default port VLAN of **0**
- A default CoS of **4**

Because VLANs can be applied to Services Edge Switch ports using the appropriate traffic classification, the explicit deny all PVID **0** will be applied at policy creation. Separate rate limits can be applied to the phone setup and payload ports on the Services Edge Switch using policy rules. A default CoS of 4 will be applied at policy role creation.

```
ServicesES(rw)->set policy profile 5 name phoneES pvid-status enable pvid 0
cos-status enable cos 4
```

Assigning Traffic Classification Rules

Forward traffic on UDP source port for IP address request (68) and and forward traffic on UDP destination ports for protocols DHCP (67) and DNS (53) on the phone VLAN, to facilitate phone auto configuration and IP address assignment. Drop traffic for protocols SNMP (161), SSH (22), Telnet (23) and FTP (20 and 21) on the phone VLAN.

```
ServicesES(rw)->set policy rule 5 udpsourceport 68 mask 16 forward
ServicesES(rw)->set policy rule 5 udpdestportIP 67 mask 16 forward
ServicesES(rw)->set policy rule 5 udpdestportIP 53 mask 16 forward
ServicesES(rw)->set policy rule 5 udpdestportIP 161 mask 16 drop
ServicesES(rw)->set policy rule 5 tcpdestportIP 22 mask 16 drop
ServicesES(rw)->set policy rule 5 tcpdestportIP 23 mask 16 drop
ServicesES(rw)->set policy rule 5 tcpdestportIP 20 mask 16 drop
ServicesES(rw)->set policy rule 5 tcpdestportIP 21 mask 16 drop
```

Apply a CoS 9 to phone setup data on VLAN 11, rate limiting the data to 5 pps with a high priority of 7 on port 2427.

Apply a CoS 10 to phone payload data on VLAN 11, rate limiting the data to 100k bps with a high priority of 7 for both source and destination on port 5004.

```
ServicesES(rw)->set policy rule 5 upddestIP 2427 mask 16 vlan 11 cos 9
ServicesES(rw)->set policy rule 5 updsourceIP 5004 mask 16 vlan 11 cos 10
ServicesES(rw)->set policy rule 5 upddestIP 5004 mask 16 vlan 11 cos 10
```

Assigning the VLAN-to-Policy Association

The nature of services related devices that might connect to a switch port is not as static as with the student or faculty roles. Services related network needs can run the gamut from temporary multimedia events to standard office users. There may be multiple VLAN and policy role associations that take care of services related needs, depending upon the connected user. This may include the requirement for multiple services related roles.

For services, the network administrator desires greater resource usage flexibility in assigning the policy to VLAN association. Authentication in this case will return only the tunnel attributes in the response message based upon the requirements of the authenticating user. Setting the VLAN-to-policy association will be handled by the maptable configuration, allowing for ease in changing the policy associated with a VLAN on the fly using Policy Manager. Specify that the **tunnel** attributes returned in the RADIUS response message will be used by the authenticating user. Associate VLAN 11 with policy role 5 using the **set policy maptable** command.

```
ServicesES(rw)->set policy maptable response tunnel
ServicesES(rw)->set policy maptable 11 5
```

Configuring Policy for the Services Edge Switch

Configuring the Policy Role

The services role is configured with:

- A profile-index value of **6**
- A name of **services**
- A default port VLAN of **0**
- A default CoS when no rule overrides CoS
- TCI overwrite enabled

```
ServicesES(rw)->set policy profile 6 name services pvid-status enable pvid 0
cos-status enable cos 4 tci-overwrite enable
```

Assigning the VLAN-to-Policy Association

Setting the VLAN-to-policy association will be handled by the policy maptable setting, allowing for ease in changing the policy associated with a VLAN on the fly using Policy Manager. Specify that the **tunnel** attributes returned in the RADIUS response message will be used by the authenticating user. Associate VLAN **10** with policy role **6** using the **set policy maptable** command.

```
ServicesES(rw)->set policy maptable response tunnel
ServicesES(rw)->set policy maptable 10 6
```

Assigning Traffic Classification Rules

Forward traffic on UDP source port for IP address request (68) and forward traffic on UDP destination ports for protocols DHCP (67) and DNS (53) on the data VLAN, to facilitate PC auto configuration and IP address assignment. Drop traffic for protocols SNMP (161), SSH (22), Telnet (23) and FTP (20 and 21) on the phone VLAN.

```
ServicesES(rw)->set policy rule 6 udpsourceportIP 68 mask 16 vlan 10 forward
ServicesES(rw)->set policy rule 6 udpdestportIP 67 mask 16 vlan 10 forward
ServicesES(rw)->set policy rule 6 udpdestportIP 53 mask 16 vlan 10 forward
ServicesES(rw)->set policy rule 6 udpdestportIP 67 mask 16 vlan 10 drop
ServicesES(rw)->set policy rule 6 udpdestportIP 53 mask 16 vlan 10 drop
ServicesES(rw)->set policy rule 6 udpdestportIP 161 mask 16 drop
ServicesES(rw)->set policy rule 6 tcpdestportIP 22 mask 16 drop
ServicesES(rw)->set policy rule 6 tcpdestportIP 23 mask 16 drop
ServicesES(rw)->set policy rule 6 tcpdestportIP 20 mask 16 drop
ServicesES(rw)->set policy rule 6 tcpdestportIP 21 mask 16 drop
```

Apply a CoS 8 to data VLAN 10 and configure it to rate-limit traffic to 1M and moderate priority of 5 for services IP subnet 10.10.30.0 mask 28. We will also enable traps and syslog for this subnet.

```
ServicesES(rw)->set policy rule 6 ipsourcesocket 10.10.30.0 mask 28 syslog enable
trap enable vlan 10 cos 8
```

Services should only be allowed access to the services server (subnet 10.10.50.0/24) and should be denied access to the faculty servers (subnet 10.10.70.0/24) and administrative servers (subnet 10.10.60.0/24).

```
ServicesES(rw)->set policy rule 6 ipdestsocket 10.10.60.0 mask 24 drop
ServicesES(rw)->set policy rule 6 ipdestsocket 10.10.70.0 mask 24 drop
```

Enable Enhanced Edge Switch Capabilities on the Services Edge Switch Platform

The Services Edge Switch platform supports a number of enhanced capabilities not available on the Fixed Switch platforms. The following enhanced policy capabilities are enabled: policy

accounting to flag the occurrence of a rule hit, syslog rule usage set to machine-readable for enterprise specific backend syslog statistics gathering, an invalid action set to default policy should an invalid policy occur, TCI overwrite enabled to allow for Type of Service (ToS) overwrite for the VoIP phone.

```
ServicesES(rw)->set policy accounting enable
ServicesES(rw)->set policy syslog machine-readable
ServicesES(rw)->set policy invalid action default-policy
ServicesES(rw)->set port tcioverwrite ge.1.1-10
```

Configuring the Distribution Layer Role

Configuring the Policy Role

The distribution role is configured with:

- A profile-index value of **7**
- A name of **distribution**
- A default CoS when no rule overrides CoS
- TCI overwrite enabled

```
Distribution(rw)->set policy profile 7 name distribution cos-status enable cos 4
tci-overwrite enable
```

Assigning the Traffic Classification to the Policy Role

Assign ports ge.1.1-26 to the distribution policy role, specifying the associated ports **1** - **26**, enable traps and enable syslog.

```
Distribution(rw)->set policy rule admin-profile port ge.1.1-26 admin-pid 7
port-string ge.1.1-26 trap enable syslog enable.
```

Assigning Traffic Classification Rules

Assign a CoS to distribution up and down stream link ports, rate-limiting the traffic to 25M.

```
Distribution(rw)->set policy rule 7 port ge.1.1-26 cos 11
Distribution(rw)->set policy rule 7 port ge.2.1-26 cos 11
```

Enable Enhanced Policy Capabilities on the Distribution Platform

The Distribution platform supports a number of policy capabilities not available on the Fixed Switch platforms. The following enhanced policy capabilities are enabled: policy accounting to flag the occurrence of a rule hit, syslog rule usage set to machine-readable for backend syslog statistics gathering, an invalid action set to default policy should an invalid policy occur, TCI overwrite enabled to allow for Type of Service (ToS) overwrite for the VoIP phone.

```
ServicesES(rw)->set policy accounting enable
ServicesES(rw)->set policy syslog machine-readable
ServicesES(rw)->set policy invalid action default-policy
ServicesES(rw)->set port tcioverwrite ge.1.1-26
ServicesES(rw)->set port tcioverwrite ge.2.1-26
```

Configuring Server[iSCSI] Policy on the 7100-Series Platform

Servers will access iSCSI storage by communicating with iSCSI storage nodes in the server farm through 7100-Series ports configured with the server[iSCSI] policy role. This policy will allow forwarding of all TCP traffic on the iSCSI port 3260 with a CoS that provides low latency and high speed. It will also provide a bilateral set of rules that allow administrators to SSH to the switch on TCP port 22 and a destination rule to allow the node to SSH to another device.

Configuring the Policy Role

The **server[iSCSI]** role is configured with:

- A profile-index of **12**
- A name of **server[iSCSI]**
- Ports **tg.1.10-15**
- PVID **0**
- CoS **12**

Create the server[iSCSI] role on the Data Center Server Switch with a default action of deny all (PVID **0**):

```
iSCSI(rw)->set policy profile 12 name "server[iSCSI]" pvid-status enable pvid 0
```

Assigning Traffic Classification Rules

Allow the server farm storage nodes to communicate on TCP source port 3260 with a CoS 12 that prioritizes the traffic for low latency and high speed.

```
iSCSI(rw)->set policy rule 1 tcpsourceportIP 3260 mask 16 forward cos 12
```

Allow administrator access to the device using SSH on TCP source port 22 and the node to SSH to another device on TCP destination port 22.

```
set policy rule 1 tcpsourceportIP 22 mask 16 forward
```

```
set policy rule 1 tcpsourceportIP 22 mask 16 forward
```

Apply this profile to ports **tg.1.10-15**.

```
set policy rule admin-profile port tg.1.10-15 mask 16 port-string tg.1.10-15
admin-pid 1
```

This completes the policy configuration for this school example.

Terms and Definitions

Table 26-5 lists terms and definitions used in this policy configuration discussion.

Table 26-5 Policy Configuration Terms and Definitions

Term	Definition
Administrative Profile	A logical container that assigns a traffic classification to a policy role.
Class of Service (CoS)	A logical container for packet priority, queue, and forwarding treatment that determines how the firmware treats a packet as it transits the link.
Filter-ID	A string that is formatted in the RADIUS access-accept packet sent back from the authentication server to the switch during the authentication process. In the Extreme Networks policy context, the string contains the name of the policy role to be applied to the authenticating user or device.
Hybrid Authentication	An authentication feature that allows the switch to use both the filter-ID and tunnel attributes in the RADIUS response message to determine how to treat the authenticating user.
Policy	A component of Secure Networks that provides for the configuration of a role based profile for the securing and provisioning of network resources based upon the function the user or device plays within the enterprise network.
Policy Maptable	A logical entity that can be configured to provide VLAN to policy role mappings.

Table 26-5 Policy Configuration Terms and Definitions (continued)

Term	Definition
Policy Profile/Role	A logical container for the rules that define a particular policy role.
Policy Rule	A logical container providing for the specification of policy behaviors associated with a policy role.
Role	The grouping of individual users or devices into a logical behavioral profile for the purpose of applying policy.
Rule Precedence	A numeric traffic classification value, associated with the policy role, the ordering of which on a precedence list determines the sequence in which classification rules are applied to a packet.
TCI Overwrite	A policy feature, when enabled in a policy role or specified in a policy rule, allows for the overwrite of the current user priority and other classification information in the VLAN tag's TCI field.
Traffic Classification	A network element such as MAC or IP address, packet type, port, or VLAN used as the basis for identifying the traffic to which the policy will be applied.
Untagged and Tagged VLAN	Untagged VLAN frames are classified to the VLAN associated with the port it enters. Tagged VLAN frames are classified to the VLAN specified in the VLAN tag; the PVID is ignored.
VLAN Authorization	An aspect of RFC3580 that provides for the inclusion of the VLAN tunnel attribute in the RADIUS Access-Accept packet defining the base VLAN-ID to be applied to the authenticating user or device.
VLAN Egress List	A configured list of ports that a frame for this VLAN can exit.

27

Multicast Configuration

This document describes the multicast feature and its configuration on S-Series devices.

How to Use Multicast in Your Network

Multicast is a “one source to many destinations” method of simultaneously sending information over a network using the most efficient delivery strategy over each link. Only the end stations that explicitly indicate a need to receive a given multicast stream will receive it.

Applications that take advantage of multicast include video conferencing, streaming video, corporate communications, distance learning, and distribution of software, stock quotes, and news.

Multicast technology includes the following protocols:

- Internet Group Management Protocol (IGMP) for IPv4, Multicast Listener Discovery (MLD) for IPv6
- Distance Vector Multicast Routing Protocol (DVMRP)
- Protocol Independent Multicast (PIM)

Unlike unicast and broadcast, multicast uses network infrastructure efficiently because only one copy of the source traffic is sent throughout the network, going only to interested receivers, minimizing the burden placed on the sender, network, and receiver. The routers in the network take care of replicating the packet, where necessary, to reach multiple receivers. If a router decides that there are no interested users downstream from itself, it prunes the stream back to the next router. Thus, unwanted streams are not sent to the pruned routers, saving bandwidth and preventing unwanted packets from being sent.

Implementing Multicast

You can implement the IGMP, DVMRP, and PIM multicast protocols on Extreme Networks devices using simple CLI commands as described in this document. A basic configuration process involves the following tasks:

1. Configuring the VLANs and IP interfaces on which you want to transmit multicast.
2. Enabling the multicast protocol(s) on configured interfaces.

For PIM, you must also configure a unicast routing protocol, such as OSPF. For both DVMRP and PIM for IPv4 to operate, IGMP must be enabled. For PIM for IPv6 to operate, the Multicast Listener Discovery (MLD) protocol must be enabled.

Understanding Multicast

Multicast allows a source to send a single copy of data using a single IP address from a well-defined range for an entire group of recipients (a multicast group). A source sends data to a multicast group by simply setting the destination IP address of the datagram to be the multicast group address. Sources do not need to register in any way before they can begin sending data to a group, and do not need to be members of the group themselves. Routers between the source and recipients use the group address to route the data, forwarding duplicate data packets only when the path to recipients diverges.

Hosts that wish to receive data from the multicast group join the group by sending a message to a multicast router on a local interface, using a multicast group membership discovery protocol, such as IGMP (IPv4) or MLD (IPv6). For more IGMP information, see "Internet Group Management Protocol (IGMP)" on page 27-2. For more MLD information, see Chapter 30, **Multicast Listener Discovery (MLD) Configuration**.

Multicast routers communicate among themselves using a multicast routing protocol, such as DVMRP, PIM-SM, or PIM-DM. These protocols calculate a multicast distribution tree (PIM-SM) or source-based tree (DVMRP and PIM-DM) of recipients to ensure that:

- Multicast traffic reaches all recipients that have joined the multicast group
- Multicast traffic does not reach networks that do not have any such recipients (unless the network is a transit network on the way to other recipients)
- The number of identical copies of the same data flowing over the same link is minimized

For more information, see "Distance Vector Multicast Routing Protocol (DVMRP)" on page 27-5 and "Protocol Independent Multicast (PIM)" on page 27-11.

Internet Group Management Protocol (IGMP)

Overview

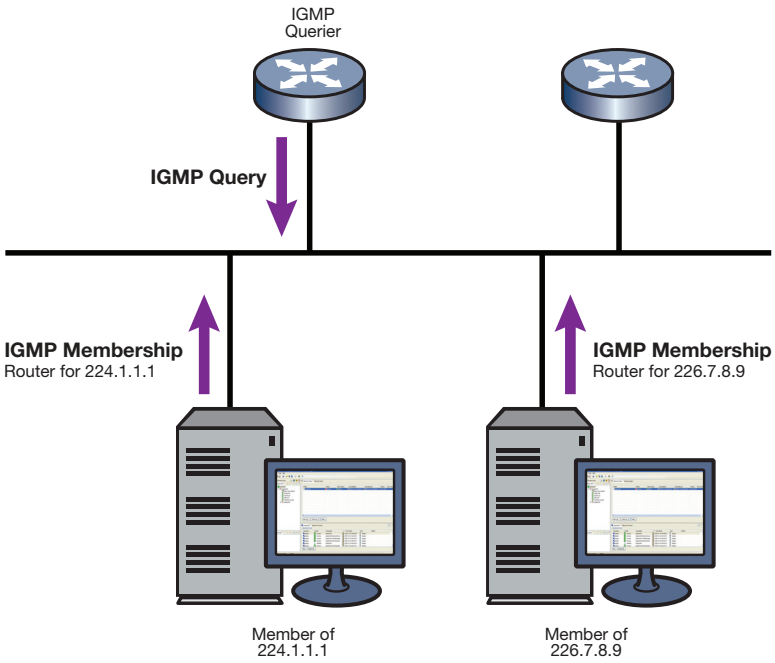
Group membership management is fundamental to the multicasting process. An arbitrary group of receivers can express interest in receiving a particular multicast stream, regardless of the physical or geographical boundaries of its members.

The purpose of IP multicast group management is to optimize a switched network's performance so multicast packets will only be forwarded to those ports containing multicast group hosts or multicast switch devices instead of flooding to all ports in the subnet (VLAN).

IGMP uses three key components to control multicast membership:

- **Source** — A server that sends an IP multicast data stream with a particular multicast destination IP and MAC address. A server may not have direct IGMP involvement, as it often does not receive a multicast stream, but only sends a multicast stream.
- **Querier** — A device that periodically sends out queries in search of multicast hosts on a directly connected network. If multiple queriers are present on the LAN, the querier with the lowest IP address assumes the role.
- **Host** — A client end station that sends one of two IGMP messages to a querier:
  - Join message — Indicates the host wants to receive transmissions associated to a particular multicast group.
  - Leave message — Indicates the host wants to stop receiving the multicast transmissions.

Figure 27-1 IGMP Querier Determining Group Membership



As shown in Figure 27-1, a multicast query-enabled device can periodically ask its hosts if they want to receive multicast traffic. If there is more than one device on the LAN performing IP multicasting, one of these devices is elected querier and assumes the responsibility of querying the LAN for group members.

Based on the group membership information learned from IGMP, a device can determine which (if any) multicast traffic needs to be forwarded to each of its ports. At Layer 3, multicast switch devices use this information, along with a multicast routing protocol, to support IP multicasting across the Internet.

IGMP provides the final step in IP multicast delivery. It is only concerned with forwarding multicast traffic from the local switch device to group members on a directly attached subnetwork or LAN segment.

IGMP neither alters nor routes any IP multicast packets. Since IGMP is not concerned with the delivery of IP multicast packets across subnetworks, an external IP multicast device is needed if IP multicast packets have to be routed across different subnetworks.

Note: On VLANs where IGMP snooping is enabled, any received multicast stream will be flooded to the VLAN until such time as the IGMP database is populated, then stream forwarding will revert to ports with group membership only.

IGMP Support on Extreme Networks Devices

Extreme Networks devices implement IGMP version 2 (RFC 2236) and IGMP version 3 (RFC 3376), which includes interoperability with version 1 hosts. IGMP version 1 is defined in RFC 1112.

Depending on your Extreme Networks device, IGMP can be configured independently at the switch level (Layer 2) and at the router level (Layer 3).

Extreme Networks devices support IGMP as follows:

- Passively snooping on the IGMP query and IGMP report packets transferred between IP multicast switches and IP multicast host groups to learn IP multicast group members. Each Layer 2 device records which ports IGMP packets are received on, depending on the kind of IGMP message, so multicast data traffic is not flooded across every port on the VLAN when it is received by the switch.

  IGMP querying is disabled by default, use **set igmp query-enable** command to enable IGMP querying as described in "Configuring IGMP" on page 27-20.

- Actively sending IGMP query messages to learn locations of multicast switches and member hosts in multicast groups within each VLAN.
- Configuration of static IGMP groups which provides for specifying the IP address (group address) and VLAN of a non-IGMP capable device, forcing the sending of multicast traffic to the device. You can configure a static IGMP group using the **set igmp static** command as described in "Configuring IGMP" on page 27-20.

Example: Sending a Multicast Stream

Figure 27-2 Sending a Multicast Stream with No Directly Attached Hosts

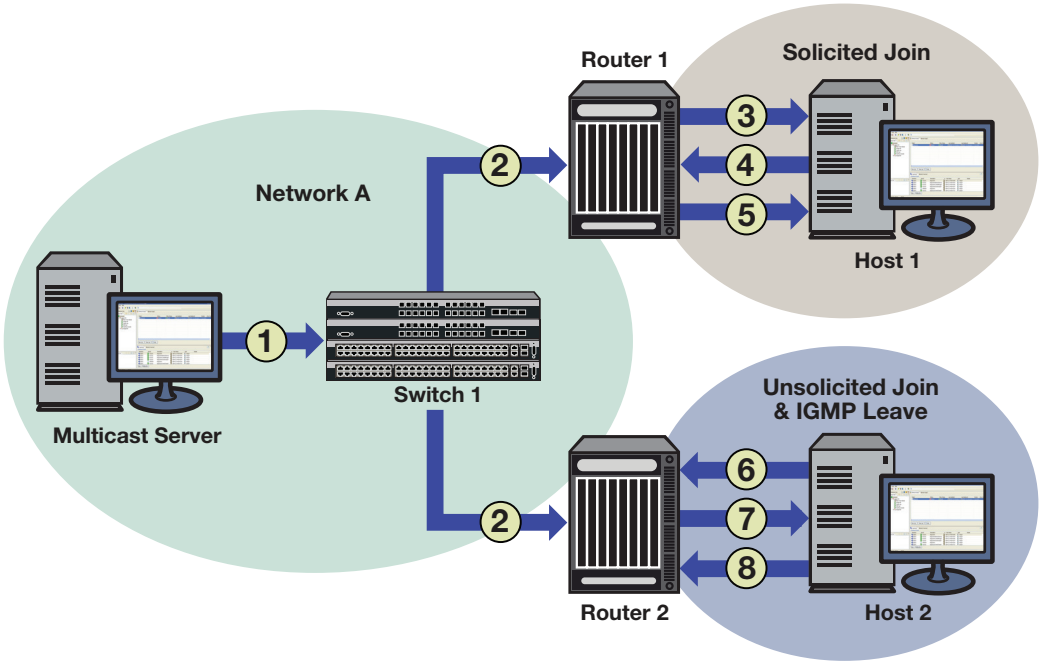


Figure 27-2 provides an example of IGMP processing on Extreme Networks devices when there are no directly attached hosts.

1. A single IP multicast server, with no directly attached hosts, sends a multicast stream into the network via Switch 1.

2. Because IGMP snooping is disabled, Switch 1 floods the multicast stream to all ports the multicast stream to all ports on the ingress VLAN.

   Each router performs an IGMP forwarding check to see if there are any hosts that want to join the multicast group on its locally attached network. Each router drops multicast packets until a host joins the group using one of the following messages:

   – **solicited join** (sent in response to an IGMP query produced by the router's interface)

     In Figure 27-2, this type of exchange occurs between Router 1 and Host 1 when:

     (3) Router 1 sends a query to potential Host 1.

     (4) Host 1 responds with a join message.

     (5) Router 1 forwards the multicast stream.

   – **unsolicited join** (sent as a request without receiving an IGMP query first)

     In Figure 27-2, this type of exchange occurs between Router 2 and Host 2 when:

     (6) Host 2 sends a join message to Router 2.

     (7) Router 2 forwards the multicast stream to Host 2.

     (8) When it no longer wants to receive the stream, Host 2 can do one of the following:

     -Send a leave message to Router 2.

     -Time out the IGMP entry by not responding to further queries from Router 2.

Distance Vector Multicast Routing Protocol (DVMRP)

Overview

DVMRP, which is used for routing multicasts within a single, autonomous system, is designed to be used as an interior gateway protocol (IGP) within a multicast domain. It is a distance-vector routing protocol that relies on IGMP functionality to provide connectionless datagram delivery to a group of hosts across a network.

Note: IGMP must be enabled for DVMRP to operate.

DVMRP routes multicast traffic using a technique known as reverse path forwarding (RPF). When a router receives IP multicast packets, it first does an RPF check to determine if the packets are received on the correct interface. If so, the router forwards the packets out to the following:

- Local IGMP receivers for that group on interfaces for which the transmitting router is the designated forwarder
- Neighbor routers that have indicated their dependence on the transmitting router for forwarding multicast packets from that source (this is determined during DVMRP Route Exchange) and from which the transmitting router has not received any prune messages.

If not, the packets are discarded by the router. The transmitting router does not forward the packets back to the source.

If a router is attached to a set of VLANs that do not want to receive from a particular multicast group, the router can send a prune message back up the distribution tree to stop subsequent packets from traveling where there are no members. DVMRP periodically re-floods in order to reach any new hosts that want to receive from a particular group.

DVMRP routers dynamically discover their neighbors by sending neighbor probe messages periodically to an IP multicast group address that is reserved for all DVMRP routers.

Key features of DVMRP are the following:

- uses the well-known multicast IP address 224.0.0.4
- uses IGMP to exchange routing datagrams
- does not require an underlying Layer 3 routing protocol to provide a path to remote multicast destinations
- combines many of the features of the Routing Information Protocol (RIP) with the Truncated Reverse Path Broadcasting (TRPB) algorithm to route multicast packets between sources and receivers

DVMRP Support on Extreme Networks Devices

DVMRP routing is implemented on Extreme Networks devices as specified in RFC 1075 and *draft-ietf-idmr-dvmrp-v3-10.txt*.

Extreme Networks devices support the following DVMRP components:

- Probe Messages for neighbor discovery
- Route Table for maintaining routes to all DVRMP networks
- Route Reports for route exchange with adjacent devices
- Mroute Table for maintaining per-source-group multicast trees
- Prune Messages for terminating multicast delivery trees
- Graft Messages for re-adding pruned multicast delivery trees

Probe Messages

Each DVMRP-enabled interface transmits multicast probe packets to inform other DVMRP routers that it is operational. Probe messages are sent every 10 seconds on every interface running DVMRP. These messages provide:

- **A mechanism for DVMRP devices to locate each other**. Probe messages contain a list of the neighbors detected for each enabled interface. If no neighbors are found, the network is considered to be a leaf network.
- **A mechanism for DVMRP devices to determine the capabilities of neighboring devices**. Probe messages contain flags about neighbors' DVMRP capabilities and version compliance.
- **A keep-alive function for quickly detecting neighbor loss**. If a probe message from an adjacent neighbor is not seen within 35 seconds, the neighbor is timed out.

Route Table

Each DVMRP-enabled device builds a DVMRP route table to maintain routes to all networks involved in DVMRP routing. As shown in the following S-Series modular switch example, the DVMRP route table contains a destination and next hop IP address, associated interface, metric value, expiration time, and up-time.

```
S Chassis(su)->show ip dvmrp route

Destination        Next Hop        Interface    Metric  Expire   Uptime
-------------------------------------------------------------------------
9.9.9.0/24         168.3.2.1       vlan.0.3200  3       00:01:52 2d, 19:34:45
21.2.2.0/24        168.3.2.1       vlan.0.3200  2       00:01:52 3d, 01:14:49
21.21.21.0/24      168.3.2.1       vlan.0.3200  2       00:01:52 3d, 01:14:49
29.2.2.0/24        168.3.2.1       vlan.0.3200  2       00:01:52 3d, 01:14:49
```

```
32.1.1.0/24         168.3.2.1       vlan.0.3200  2       00:01:52 3d, 01:14:49
32.11.11.0/24       168.3.2.1       vlan.0.3200  2       00:01:52 3d, 01:14:49
92.9.2.0/24         168.3.2.1       vlan.0.3200  2       00:01:52 3d, 01:14:49
100.3.3.0/24        Connected       vlan.0.3200  1       00:00:00 02:09:22
129.2.9.0/24        168.3.2.1       vlan.0.3200  2       00:01:52 2d, 19:02:06
139.3.9.0/28        Connected       vlan.0.390   1       00:00:00 3d, 01:14:54
160.2.2.0/24        168.3.2.1       vlan.0.3200  2       00:01:52 3d, 01:14:49
168.2.1.0/24        168.3.2.1       vlan.0.3200  2       00:01:52 3d, 01:14:49
168.3.0.0/16        Connected       vlan.0.3200  1       00:00:00 02:09:22
168.3.1.0/26        Connected       vlan.0.3100  5       00:00:00 2d, 21:54:44
168.8.1.0/24        168.3.2.1       vlan.0.3200  3       00:01:52 2d, 19:34:25
188.21.21.0/24      168.3.2.1       vlan.0.3200  2       00:01:52 3d, 01:14:49
188.23.23.0/24      168.3.2.1       vlan.0.3200  2       00:02:02 3d, 01:14:49
189.8.9.0/24        168.3.2.1       vlan.0.3200  4       00:02:02 2d, 19:34:15
191.9.1.0/24        168.3.2.1       vlan.0.3200  3       00:02:02 2d, 19:34:45
191.9.9.0/24        168.3.2.1       vlan.0.3200  3       00:02:02 2d, 19:34:45
192.9.2.0/24        168.3.2.1       vlan.0.3200  2       00:02:02 3d, 01:14:49
193.9.3.0/24        Connected       vlan.0.930   1       00:00:00 3d, 01:14:54
198.9.8.0/24        168.3.2.1       vlan.0.3200  4       00:02:02 2d, 19:34:15
198.23.23.0/24      168.3.2.1       vlan.0.3200  2       00:02:02 3d, 01:14:49
199.23.23.0/24      168.3.2.1       vlan.0.3200  2       00:02:02 3d, 01:14:49
250.9.9.0/24        168.3.2.1       vlan.0.3200  2       00:02:02 3d, 01:14:49

The number of DVMRP routes is  26
```

Route Reports

DVMRP-enabled devices send route report packets to adjacent DVMRP devices every 60 seconds. When a DVMRP device receives one, it checks to verify that the report is from a known neighbor before processing.

The first time a device sees its own address in a neighbor's probe packet, it sends a unicast copy of its entire routing table to the neighbor to reduce start-up time.

The route report packet contains data about all networks/routes of which the sending device is aware. This information is used to determine the reverse path back to a particular multicast source. Every DVMRP device keeps a separate metric associated with each route. This metric is the sum of all interface metrics between the device originating the report and the source network.

DVMRP devices accept route reports for aggregated source networks in accordance with classless inter-domain devices (CIDR). This means that, if a prune or graft is received on a downstream interface for which the source network is aggregated, then a prune or graft should be sent upstream (to the multicast source).

If a DVMRP device has a large number of DVMRP routes, it will spread route reports across the route update interval (60 seconds) to avoid bottlenecks in processing and route synchronization issues.

For the purpose of pruning, DVMRP needs to know which downstream routes depend on the device for receiving multicast streams. Using poison reverse, the upstream router maintains a table of the source network and all downstream devices that are dependent on the upstream device.

Mroute Table

DVMRP-enabled devices use the mroute table to maintain a source-specific forwarding tree.

When a DVMRP device is initialized, it assumes the role of the designated forwarder for all of its locally attached networks. Before forwarding any packets, all devices use IGMP to learn which networks would like to receive particular multicast group streams. In the case of a shared network, the device with a lower interface metric (a configurable value), or the lower IP address will become the designated forwarder.

A DVMRP device forwards multicast packets first by determining the upstream interface, and then by building the downstream interface list. If a downstream router has no hosts for a multicast stream, it sends a prune message to the upstream router. If the upstream router's outbound list is now empty, it may send a prune message to its upstream router.

If a downstream device has pruned a multicast group that a host would like to now receive, the downstream device must send a DVMRP graft message to its upstream device. The DVMRP graft will traverse the source-specific multicast delivery tree to the device that is receiving this stream.

As shown in the following example, the Mroute table displays the incoming interface IP address, the multicast group address, the uptime of the stream, incoming interface port number, and the outgoing interface port number.

```
S Chassis(su)->show ip mroute

IP Multicast Routing Table
Flags: D - Dense, S - Sparse, C - Connected, L - Local, P - Pruned
       R - RP-bit set, F - Register flag, T - SPT-bit set, J - Join SPT
Timers: Uptime/Expires

  DVMRP (191.9.1.11/32, 234.1.1.1), 00:00:36/00:00:00, flags:
        Incoming interface: vlan.0.3200
        Outgoing interface list:

  DVMRP (191.9.1.12/32, 234.1.1.1), 00:00:36/00:00:00, flags:
        Incoming interface: vlan.0.3200
        Outgoing interface list:

  DVMRP (193.9.3.30/32, 234.3.3.3), 3d, 01:13:10/00:00:00, flags:
        Incoming interface: vlan.0.930
        Outgoing interface list:
          vlan.0.3100, Forward/DVMRP, 2d, 19:32:38/00:00:00

  DVMRP (193.9.3.31/32, 234.3.3.3), 3d, 01:13:04/00:00:00, flags:
        Incoming interface: vlan.0.930
        Outgoing interface list:
          vlan.0.3100, Forward/DVMRP, 2d, 19:32:38/00:00:00

  DVMRP (193.9.3.32/32, 234.3.3.3), 3d, 01:13:11/00:00:00, flags:
        Incoming interface: vlan.0.930
        Outgoing interface list:
          vlan.0.3100, Forward/DVMRP, 2d, 19:32:38/00:00:00
```

Prune Messages

If a device receives a datagram that has no IGMP group members present, and all the downstream networks are leaf networks, the device sends a prune packet upstream to the source tree.

When sending a prune upstream, the device:

1. Decides if the upstream neighbor is capable of receiving prunes.
   - If it is not, then the sending device proceeds no further.
   - If it is, then the sending device proceeds as follows.
2. Stops any pending grafts awaiting acknowledgments.
3. Determines the prune lifetime.

   This value should be the minimum of the default prune lifetime (randomized to prevent synchronization) and the remaining prune lifetimes of the downstream neighbors.
4. Forms and transmits the packet to the upstream neighbor for the source.

To ensure the prune is accepted, the DVMRP-enabled device sets a negative cache prune entry for three seconds. If the traffic has not stopped after three seconds, the device sends another prune and doubles the cache entry. This method is called exponential back-off. The more prunes that are dropped, the longer the back-off becomes.

After the prune lifetime expires (two hours), the prune transmission process is repeated.

When receiving a prune, the upstream device:

1. Decides if the sending neighbor is known.
   - If the neighbor is unknown, it discards the received prune.
   - If the neighbor is known, the receiving device proceeds as follows.
2. Ensures the prune message contains at least the correct amount of data.
3. Copies the source address, group address, and prune time-out value, and, if it is available in the packet, the netmask value to determine the route to which the prune applies.
4. Determines if there is active source information for the source network, multicast group (S,G) pair.
   - If there is not, then the device ignores the prune.
   - If there is, then the device proceeds as follows.
5. Verifies that the prune was received from a dependent neighbor for the source network.
   - If it was not, then the device discards the prune.
   - If it was, then the device proceeds as follows.
6. Determines if a prune is currently active from the same dependent neighbor for this S,G pair.
   - If not active, creates a state for the new prune and sets a timer for the prune lifetime
   - If active, resets the timer to the new time-out value.
7. Determines if all dependent downstream devices on the interface from which the prune was received have now sent prunes.
   - If they have not, removes the interface from all forwarding cache entries for this group instantiated using the route to which the prune applies.
   - If they have, determines if there are group members active on the interface and if this device is the designated forwarder for the network.

Graft Messages

Leaf devices send graft messages when the following occur:

- A new local member joins a group that has been pruned upstream and this device is the designated forwarder for the source.
- A new dependent downstream device appears on a pruned branch.
- A dependent downstream device on a pruned branch restarts.
- A graft retransmission timer expires before a graft ACK is received.

Graft messages are sent upstream hop-by-hop until the multicast tree is reached. Since there is no way to tell whether a graft message was lost or the source has stopped sending, each graft message is acknowledged hop-by-hop.

When sending grafts, the downstream device does the following:

1. Verifies a prune exists for the source network and group.
2. Verifies that the upstream device is capable of receiving prunes (and therefore grafts).

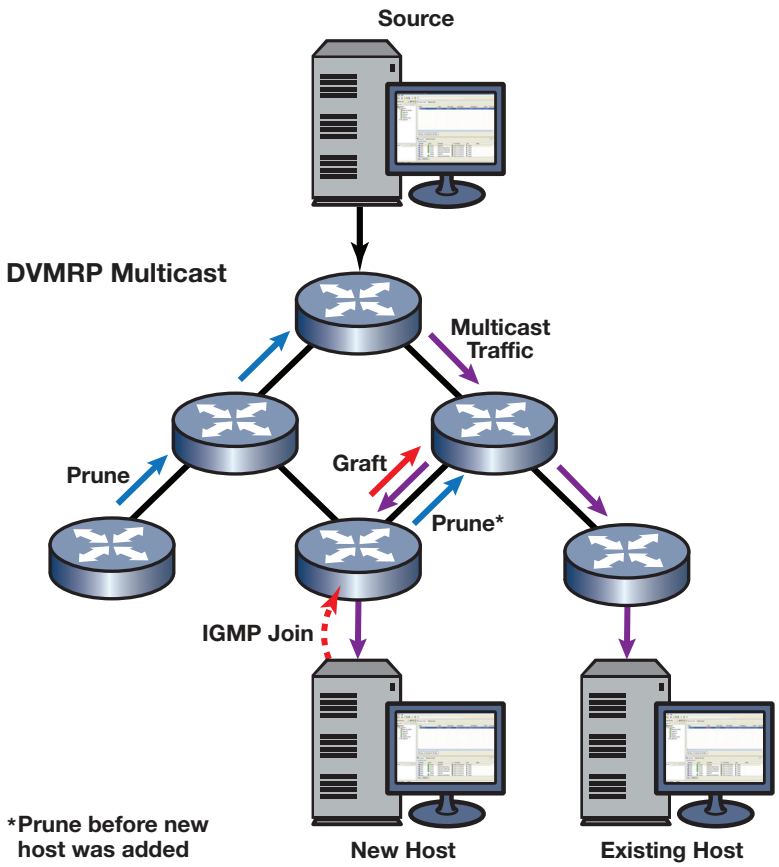
3. Adds the graft to the retransmission timer list awaiting an acknowledgment.
4. Formulates and transmits the graft packet.

When receiving grafts, the upstream device does the following:

1. Verifies whether the neighbor is known.
   - If unknown, discards the received graft.
   - If known, proceeds as follows.
2. Ensures the graft message contains at least the correct amount of data.
3. Sends back a graft ACK to the sender.
4. If the sender was a downstream dependent neighbor from which a prune had previously been received:
   - Removes the prune state for this neighbor.
   - If necessary, updates any forwarding cache entries based on this (source, group) pair to include this downstream interface.

Figure 27-3 shows the DVMRP pruning and grafting process.

Figure 27-3 DVMRP Pruning and Grafting



Protocol Independent Multicast (PIM)

Overview

PIM is made up of a collection of multicast routing protocols optimized for different multicast environments. The S-Series platform supports two PIM modes: sparse mode (SM) and dense mode (DM).

PIM dynamically builds a distribution tree for forwarding multicast data on a network. It is designed for use where there may be many devices communicating at the same time, and any one of the devices could be the sender at any particular time. Scenarios for using PIM multicasting include desktop video conferencing and telephone conference calls.

PIM relies on IGMP technology to determine group memberships and uses existing unicast routes to perform reverse path forwarding (RPF) checks, which are, essentially, a route lookup on the source. Its routing engine then returns the best interface, regardless of how the routing table is constructed. In this sense, PIM is independent of any routing protocol. It can perform RPF checks using protocol-specific routes (for example, OSPF routes), static routes, or a combination of route types.

Note: IGMP must be enabled for PIM to operate.

PIM-SM is a multicast routing protocol optimized for a network in which multicast recipients are sparsely distributed throughout the network. PIM-SM assumes that the network contains subnets that will not want a given multicast packet. Given this assumption, routers must explicitly tell their upstream neighbors about their interest in particular groups and sources. PIM-SM creates multicast distribution trees based upon the sending and receiving of PIM Join and Prune messages to join or leave the distribution tree.

PIM-SM by default uses shared trees, which are multicast distribution trees rooted at some selected node called the Rendezvous Point (RP) and used by all sources sending to the multicast group. PIM-SM is a soft-state protocol. All state periodically times out after receiving the control message that instantiated it. To keep the state alive, all PIM Join messages are periodically retransmitted.

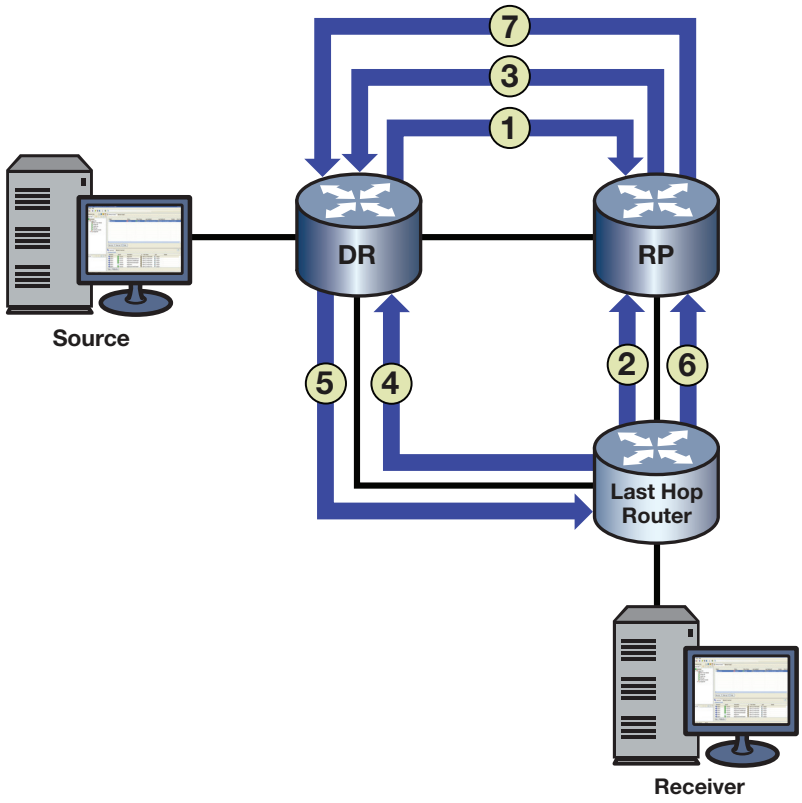
PIM-DM is a multicast routing protocol optimized for a network in which the receivers for any multicast group are assumed to be densely distributed throughout the network. PIM-DM assumes that most, or at the very least, many subnets in the network will want any given multicast packet. PIM-DM uses source-based trees rather than the shared (RP-based) tree used by PIM-SM. PIM-DM does not use the concept of an RP in which all sources first send packets to a single router. PIM-DM initially sends multicast data to all hosts in the network. Routers that do not have any interested hosts then send PIM Prune messages to remove themselves from the tree. Because PIM-DM does not use RPs, it is easier to implement and deploy than PIM-SM. It is an efficient protocol when most receivers are interested in the multicast data, but does not scale well across larger domains in which most receivers are not interested in the data.

PIM-SM

PIM-SM, a shared distribution tree technology, designates a router as the rendezvous point (RP), which is the root of a shared tree for a particular group. All sources send packets to the group via the RP (that is, traffic flows from the sender to the RP, and from the RP to the receiver). By maintaining one RP-rooted tree instead of multiple source-rooted trees, bandwidth is conserved.

Figure 27-4 illustrates the PIM traffic flow.

Figure 27-4 PIM Traffic Flow



1. The source's DR registers (encapsulates) and sends multicast data from the source directly to the RP via a unicast routing protocol (number 1 in figure). The RP de-encapsulates each register message and sends the resulting multicast packet down the shared tree.
2. The last-hop router (that is, the receiver's DR) sends a multicast group (*,G) join message upstream to the RP, indicating that the receiver wants to receive the multicast data (number 2 in figure). This builds the RP tree (RPT) between the last-hop router and the RP.
3. The RP sends an S,G join message to the source (number 3 in figure). It may send the join message immediately, or after the data rate exceeds a configured threshold. This allows the administrator to control how PIM-SM uses network resources.
4. The last-hop router joins the shortest path tree (SPT) and sends an S,G join message to the source. (number 4 in figure).This builds the SPT.
5. Native multicast packets (that is, non-registered packets) are sent from the source's DR to the receiver on its SPT (number 5 in figure), while registered multicast packets continue to be sent from the source's DR to the RP.
6. A prune message is sent from the last-hop router to the RP (number 6 in figure).
7. A prune message (*register-stop*) is sent from the RP to the source's DR (number 7 in figure). Once traffic is flowing down the SPT, the RPT is pruned for that given S,G.

When receivers go away, prunes are sent (S,G prune messages towards the source on the SPT, and *,G prune messages towards the RP on the RPT). When new receivers appear, the process begins again.

PIM Support on Extreme Networks Devices

Extreme Networks devices support version 2 of the PIM protocol as described in RFC 4601 and *draft-ietf-pim-sm-v2-new-09*.

The PIM specifications define several modes or methods by which a PIM router can build the distribution tree. Extreme Networks devices support sparse mode (PIM-SM), dense mode (PIM-DM) and source-specific multicast (PIM-SSM).

PIM-SM uses only those routers that need to be included in forwarding multicast data. PIM-SM uses a host-initiated process to build and maintain the multicast distribution tree. Sparse mode routers use bandwidth more efficiently than other modes, but can require more processing time when working with large numbers of streams.

PIM-SSM is a subset of the PIM-SM protocol. PIM-SSM is disabled by default and must be explicitly enabled. PIM-SSM builds trees that are rooted in just one source, offering a more secure and scalable model for a limited amount of applications such as broadcasting of content. PIM-SSM is not independent of PIM-SM. PIM-SM must be enabled on all interfaces that use PIM-SSM. In PIM-SSM, an IP datagram is transmitted by a source S to an SSM destination address G, and receivers can receive this datagram by subscribing to channel (S,G). The destination address range for PIM SSM is 232.0.0.0/8 for IPv4 and ff3x:0000/32 where (x = 4,5,8, or E) for IPv6.

PIM-SSM does not require an RP candidate or BSR candidate. In a mixed PIM-SM and PIM-SSM configuration, the RP candidate and BSR candidate need to be configured for the PIM-SM group address range only. Enable IGMP on all PIM-SSM interfaces and enable IGMP querying on the PIM-SSM receiver interface. PIM-SSM requires IGMPv3 and MLDv2 at the edge of the network to process the source-specific IGMP and MLD joins.

PIM-DM creates a source-based distribution tree with minimal configuration for networks containing receivers for most PIM enabled network interfaces for any given multicast data stream.

Key Features

Key features of PIM-SM are the following:

- Uses IGMP to propagate group membership information
- Sends hello messages to determine neighbor presence and configuration
- Sends join/prune messages to determine the need to retain multicast route information for a particular group on an interface
- Sends assert messages to resolve conflicts that occur regarding inbound interfaces
- Uses routes in the Multicast Routing Information Base (MRIB) to perform its reverse path forwarding check

Key features of PIM-SSM are the following:

- Protects against Denial of Service Attacks from unwanted sources
- Is easier to provision and maintain due to the single source address that a receiver can request data from
- Provides the ideal mechanism for internet broadcasts that originate from a single source and go to multiple receivers
- Does not require unique multicast addresses; it depends upon the receiver request for the destination address of the broadcast

Key features of PIM-DM are the following:

- Ease of configuration
- Operational and overhead efficiencies when a high density of network PIM enabled router interfaces have receivers attached for a given multicast data stream

PIM-SM Message Types

Extreme Networks PIM-SM-enabled devices use the following message types:

- Hello — These messages announce the sender's presence to other PIM-SM devices. The hello packet includes options such as:
  - Hold time — the length of time to keep the sender reachable
  - Designated router (DR) priority — used to designate which PIM-SM device will act on behalf of sources and receivers in the PIM-SM domain
- Register — These messages are used by a source's DR to encapsulate (register) multicast data, and send it to the rendezvous point (RP) — a PIM-SM router designated as the root of a shared tree.
- Register-Stop — These messages are used by the RP to tell the source's DR to stop registering traffic for a particular source.
- Join/Prune (J/P) — These messages contain information on group membership received from downstream routers.

  PIM-SM adopts RPF technology in the join/prune process. When a multicast packet arrives, the router first judges the correctness of the arriving interfaces:
  - If the packet is a source address/multicast group (S,G) entry (on the shortest path tree (SPT)), then the correct interface is the reverse path forwarding (RPF) interface towards the source.
  - If the packet is not an S,G entry (on the RP tree (RPT)), then the correct interface is the RPF interface towards the RP.

  A router directly connected to the hosts is often referred to as a leaf router or DR. The leaf router is responsible for sending the prune messages to the RP, informing it to stop sending multicast packets associated with a specific multicast group. When the RP receives the prune message, it will no longer forward the multicast traffic out the interface on which it received the prune message.
- Assert — These messages indicate that the device received a data packet on its outbound (receiving) interface for the group. They report the metric or distance to the source or RP to help the device identify the most direct path to the root of the tree. If multiple routers claim to have the most direct path to the source or RP, each device sends its own assert message and the router with the best metric wins. The other device will then remove that link from its outbound interface list for the group.
- Bootstrap — These messages are sent by the PIM-SM router that has been elected as the bootstrap router (BSR) to inform all PIM-SM routes of the RP/group mappings.
- Candidate RP message — These messages are sent by the configured candidate RP routers to the BSR to inform the BSR of its RP/group candidacy.

PIM-SSM Message Types

The PIM-SSM implementation is a subset of PIM-SM protocol. PIM-SM and PIM-SSM can coexist on a single router and are both implemented using the PIM-SM protocol.

Extreme Networks PIM-SSM enabled devices use the following PIM-SM message types:

- Hello — These messages announce the sender's presence to other PIM-SM devices. The hello packet includes options such as:
  - Hold time — the length of time to keep the sender reachable
  - Designated router (DR) priority — used to designate which PIM-SM device will act on behalf of sources and receivers in the PIM-SM domain

- Join/Prune (J/P) — These messages contain information on group membership received from downstream routers.
- PIM-SM adopts RPF technology in the join/prune process. When a multicast packet arrives, the router first judges the correctness of the arriving interfaces:
  - If the packet is a source address/multicast group (S,G) entry (on the shortest path tree (SPT)), then the correct interface is the reverse path forwarding (RPF) interface towards the source.
- Assert — These messages indicate that the device received a data packet on its outbound (receiving) interface for the group. They report the metric or distance to the source to help the device identify the most direct path to the root of the tree. If multiple routers claim to have the most direct path to the source, each device sends its own assert message and the router with the best metric wins. The other device will then remove that link from its outbound interface list for the group.

PIM-DM Message Types

The PIM-DM-enabled devices use the following message types:

- Hello — These messages announce the sender's presence to other PIM-DM devices. The hello packet includes options such as:
  - Hold time — the length of time to keep the sender reachable
- Join/Prune (J/P) — These messages contain information on group membership received from downstream routers.
- Graft – These messages are sent upstream when new group membership is added to a pruned branch, instructing the upstream router to forward multicast data for the specified source to the downstream router.
- Graft ACK – These messages are sent to the downstream router, acknowledging the reception of a graft message from the downstream router.
- State Refresh – These messages are generated periodically by the PIM-DM router directly connected to a source and sent to neighbor routers. State refresh message minimize network overhead by conveying prune state.

Anycast-RP

The S-Series supports anycast-RP. Anycast-RP provides a means of avoiding a single point of failure through fast convergence when a PIM RP router fails. It also provides for RP load balancing. The relationship between a source or receiver and the PIM RP router is a one-to-one relationship. The relationship between a source or receiver and an Anycast-RP set of routers is a one-to-many relationship, where one of multiple anycast configured RPs is selected by the routing protocol to be the source or receiver PIM RP router.

Anycast-RP provides for the selection of a set of routers to be identified as anycast RPs by

- Configuring each member of the anycast-RP set as either a static RP or a PIM candidate RP using the same loopback anycast IP address as the RP IP address
- Configuring:
  - A loopback interface with the same IP address for each anycast-RP router in the set
  - Either a second loopback interface or another hardware interface to be configured with a unique address for this peer of the anycast-RP set

Each anycast-RP router is configured with the same anycast-RP address and all the peer-addresses of each router in the anycast-RP router set. A unique peer address is used to allow each member of the anycast-RP set to identify all other members of the set. Each anycast-RP and peer-address combination is configured in its own command line entry using the **ip pim anycast-rp** command.

The routing protocol determines which member of the anycast-RP router set will function as the PIM RP router. Should the PIM RP router fail, the routing protocol determines the next anycast-RP router that will become the new PIM RP router, based upon the routing protocol's routing criteria.

Figure 27-5 on page 27-17 illustrates an Anycast-RP configuration example.

RP1

- Create and enable VLAN 10 with IP interfaces
- Configure the underlying unicast routing protocol (OSPF)
- Enable IGMP on VLAN 10
- Configure interface loopback 1 with the anycast-RP address 1.1.1.1/32
- Configure interface loopback 2 with the peer-address 10.0.0.1/32
- Configure 1.1.1.1 as either a static RP using the **ip pim rp-address** command or an RP candidate using the **ip pim rp-candidate** command
- Configure RP 1.1.1.1 as an anycast-RP set with the peer-addresses for RP1, RP2, and RP3 using the following commands:
  - **ip pim anycast-rp 1.1.1.1 10.0.0.1**
  - **ip pim anycast-rp 1.1.1.1 20.0.0.1**
  - **ip pim anycast-rp 1.1.1.1 30.0.0.1**

RP2

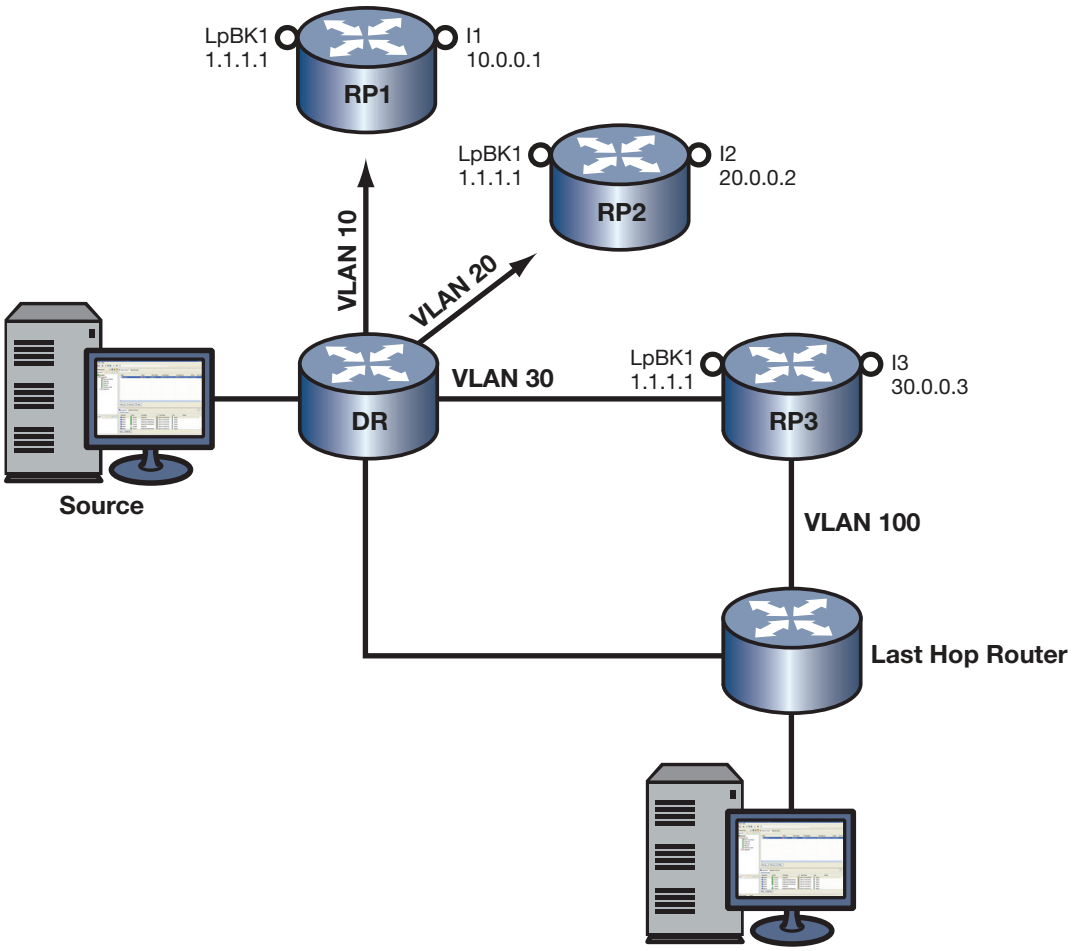
- Create and enable VLAN 20 with IP interfaces
- Configure the underlying unicast routing protocol (OSPF)
- Enable IGMP on VLAN 20
- Configure interface loopback 1 with the anycast-RP address 1.1.1.1/32
- Configure interface loopback 2 with the peer-address 20.0.0.1/32
- Configure 1.1.1.1 as either a static RP using the **ip pim rp-address** command or an RP candidate using the **ip pim rp-candidate** command
- Configure RP 1.1.1.1 as an anycast-RP set with the peer-addresses for RP1, RP2, and RP3 using the following commands:
  - **ip pim anycast-rp 1.1.1.1 10.0.0.1**
  - **ip pim anycast-rp 1.1.1.1 20.0.0.1**
  - **ip pim anycast-rp 1.1.1.1 30.0.0.1**

RP3

- Create and enable VLAN 30 with IP interfaces
- Configure the underlying unicast routing protocol (OSPF)
- Enable IGMP on VLAN 30
- Configure interface loopback 1 with the anycast-RP address 1.1.1.1/32

- Configure interface loopback 2 with the peer-address 30.0.0.1/32
- Configure 1.1.1.1 as either a static RP using the **ip pim rp-address** command or an RP candidate using the **ip pim rp-candidate** command
- Configure RP 1.1.1.1 as an anycast-RP set with the peer-addresses for RP1, RP2, and RP3 using the following commands:
  - **ip pim anycast-rp 1.1.1.1 10.0.0.1**
  - **ip pim anycast-rp 1.1.1.1 20.0.0.1**
  - **ip pim anycast-rp 1.1.1.1 30.0.0.1**

Figure 27-5 Anycast-RP Configuration



With all anycast-RPs configured, the routing protocol selects RP3 as the RP for this domain based upon its routing criteria. Should RP3 fail, the routing protocol will determine which of the remaining routers in the anycast-RP set will take over as RP. Should the failed router return to an operational state, the routing protocol will determine whether a new PIM RP will be selected based upon current conditions.

PIM-DM

PIM-DM is a source-based tree technology. Multicast data is sent to all hosts in the network. Routers that do not have any interested hosts send PIM Prune messages to remove themselves from the tree associated with the multicast data source. If a router connects to a LAN with

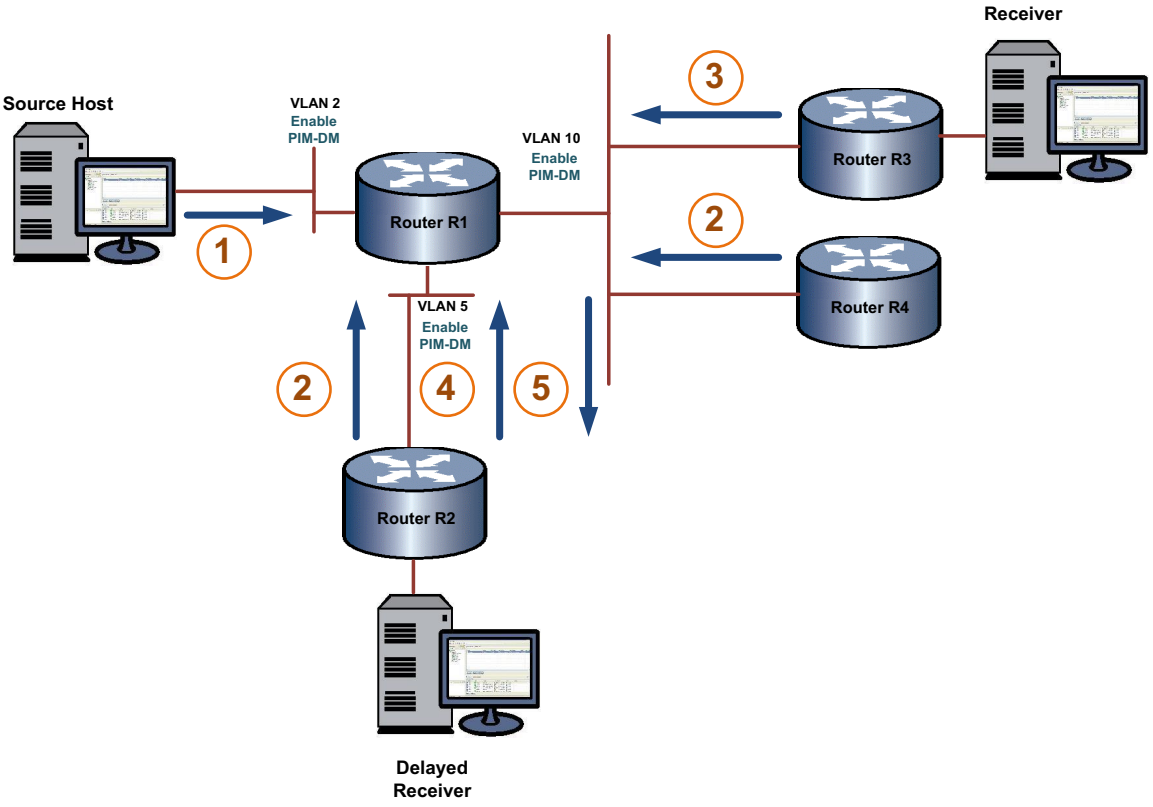
multiple routers, and that LAN contains both non-interested and interested routers for a given multicast data source:

- Non-interested routers will send a prune message upstream towards the source to inform the sending router that it does not have any receivers for the data stream
- Interested routers, having received a prune message, will send a join message upstream towards the source to inform the sending router that it has receivers for the data stream

Should a non-interested router become interested in a given mulitcast data stream, a graft message is sent upstream towards the source, informing the upstream router of its interest. The upstream router returns a graft ACK message acknowledging reception of the graft message. If no acknowledgment is received prior to the expiration of the graft retry timer (GRT), the interested router sends another graft message upstream.

Figure 27-6 illustrates the PIM-DM traffic flow.

Figure 27-6 PIM-DM Traffic Flow



1. The source sends multicast data on VLAN 2. R1 floods the data out all interfaces enabled for PIM-DM.
2. When R2 and R4 receive the multicast data, at this time no downstream receivers exist for the data's multicast group. At this time, both R2 and R4 send prune messages on the upstream interfaces indicating they have no receivers for the source data.
3. R3 receives the prune sent by R4 and responds by sending a join out its upstream interface indicating that it does have a receiver for the source data.
4. A receiver connected to R2 joins the multicast group for the source data. R2 responds by sending a graft message out its upstream interface informing R1 that it should now send its source data out its downstream interfaces to R2.

5. When R1 receives the graft message, a graft ACK message is returned to R2. R2 initiates a graft retry timer upon sending the graft message to R1. If R2 does not receive a graft ACK message from R1 before the timer expires, a new graft message is sent to R1.

When receivers go away, prunes are sent (S,G prune messages are sent towards the source). When new receivers appear, the process begins again.

PIM Terms and Definitions

Table 27-1 lists terms and definitions used in PIM configuration.

Table 27-1 PIM Terms and Definitions

Term	Definition
Bootstrap Router (BSR)	A PIM router responsible for collecting, within a PIM domain, the set of potential rendezvous points (RPs) and distributing the RP set information to all PIM routers within the domain. The BSR is dynamically elected from the set of candidate BSRs. RP set information includes group-to-RP mappings.
Candidate Bootstrap Router (Candidate-BSR)	A small number of routers within a PIM domain are configured as candidate BSRs, and each C-BSR is given a BSR priority. All C-BSRs multicast bootstrap messages (BSMs) containing their priority to the ALL-PIM-ROUTERS group. When a C-BSR receives a bootstrap message from a C-BSR with a higher priority, it stops sending. This continues until only one C-BSR remains sending bootstrap messages, and it becomes the elected BSR for the domain.
Rendezvous Point (RP)	The root of a group-specific distribution tree whose branches extend to all nodes in the PIM domain that want to receive traffic sent to the group. RPs provide a place for receivers and senders to meet. Senders use RPs to announce their existence, and receivers use RPs to learn about new senders of a group. The RP router, for the group, is selected by using the hash algorithm defined in RFC 2362.
Candidate Rendezvous Point (Candidate-RP)	PIM routers configured to participate as RPs for some or all groups. C-RPs send C-RP Advertisement messages to the BSR. The messages contain the list of group prefixes for which the C-RP is willing to be the RP. Once the PIM-SM routers receive the BSR's message, the routers use a common hashing algorithm to hash the C-RP address, group, and mask together to identify which router will be the RP for a given group. A C-RP router must also learn which PIM-SM router is the BSR. Each designated candidate-BSR (C-BSR) asserts itself as the BSR, then defers once it receives a preferable BSR message. Eventually, all C-RPs send their messages to a single BSR, which communicates the *Candidate RP-set* to all PIM-SM routers in the domain.
dense mode	PIM dense mode (DM) uses a source-based tree to distribute multicast data. DM does not require routers on the network to explicitly request interest in a given data stream and assumes that most routers in any given network will be interested in order to maintain efficient operation.
Static RP	If a BSR is not used to distribute RP set information, RP-to-group mappings are configured statically on each router. Static RP configuration and use of bootstrap routers are mutually exclusive. You should not configure both in a PIM-SM domain because such configuration could result in inconsistent RP sets. Statically configured RP set information will take precedence over RP set information learned from a BSR.

Table 27-1 PIM Terms and Definitions (continued)

Term	Definition
Anycast-RP	Anycast-RP provides a means of fast convergence when a PIM RP router fails. All members of the anycast-RP set share the same IP address configured on a loopback interface of each set member. A peer-address associated with the member specifies a unique IP address that identifies the router and can be either a loopback or physical interface.
Designated Router (DR)	A designated router is elected from all the PIM routers on a shared network. DRs are responsible for encapsulating multicast data from local sources into PIM-SM register messages and for unicasting them to the RP. The router with the highest priority wins the DR election. In the case of a tie, the router with the highest IP address wins.
PIM Domain	A contiguous set of routers that implement PIM and are configured to operate within a common boundary defined by PIM multicast border routers.
PIM Multicast Border Router (PMBR)	A router that connects a PIM domain to other multicast routing domains.
sparse mode	PIM sparse mode (SM) uses a host-initiated process to build and maintain the multicast distribution tree, using only those routers that need to be included in forwarding multicast data. Sparse mode routers use bandwidth more efficiently than other modes, but can require more processing time when working with large numbers of streams
source-specific multicast	PIM source-specific multicast (SSM) is a modular switch only subset of the PIM-SM protocol that builds trees rooted in just one source and is used by applications such as content broadcasting.

Configuring Multicast

This section provides the following information about configuring multicast.

Configuring IGMP

IGMP is configured at the switch level in any command mode on the S-Series devices. At Layer 2, IGMP can be enabled for VLANs, regardless of whether it is enabled on routed interfaces. If, however, IGMP is enabled on a routed interface, and the routed interface is a routed VLAN, then IGMP must also be enabled at the switch level.

IGMP Configuration Commands

Table 27-2 lists the IGMP configuration commands for S-Series devices.

Table 27-2 IGMP Configuration Commands

Task	Command
Enable IGMP on one or more VLANs.	**set igmp enable** *vlan-list*

Table 27-2 IGMP Configuration Commands (continued)

Task	Command
Disable IGMP on one or more VLANs.	**set igmp disable** *vlan-list*
Enable IGMP querying on one or more VLANs.	**set igmp query-enable** *vlan-list*
Disable IGMP querying on one or more VLANs.	**set igmp query-disable** *vlan-list*
Determine what action to take with multicast frames when the multicast group table is full.	**set igmp grp-full-action** *action*
Configure IGMP settings on one or more VLANs.	**set igmp config** *vlan-list* {[**query-interval** *query-interval*] **[igmp-version** *igmpversion*] [**max-resp-time** *max-resp-time*] [**robustness** *robustness*] [**last-mem-int** *last-mem-int*] [**fast-leave** *fast-leave*] [**rtr-alert-checking** *rtr-alert-checking*] [**filter-id** *filter-id*] [**filter-status** {enable | disable}]}
Remove IGMP configuration settings for one or more VLANs.	**set igmp delete** *vlan-list*
Change the IGMP classification of received IP frames.	**set igmp protocols** [**classification** *classification*] [**protocol-id** *protocol-id*] [**modify**]
Clear the binding of IP protocol ID to IGMP classification.	**clear igmp protocols** [**protocol-id** *protocol-id*]
Creates a new static IGMP entry or to adds one or more new include or exclude ports to an existing entry.	**set igmp static** *group vlan-list* [**modify**] [**include-ports** *include-ports*] [**exclude-ports** *exclude-ports*]
Create an input filter to apply to the VLAN.	**set igmp input-filter** *filter-id rule-id* **start-ip** *ip-address* **end-ip** *ip-address* **protocol-action** {deny | allow} **flow-action** {drop | flood | allow}
Clear an input filter.	**clear igmp input-filter** *filter-id* [*rule-id*]
Set the action taken when the first few frames of a multicast stream are received (that is, before the stream is added to the MLD database).	**set igmp unknown-input-action** {routers | flood | discard}

Basic IGMP Configurations

Procedure 27-1 describes the basic steps to configure IGMP on S-Series devices. This procedure assumes that the VLANs on which IGMP will run have been configured and enabled with IP interfaces.

Procedure 27-1 Basic IGMP Configuration

Step	Task	Command
1.	In switch mode, configure IGMP for each VLAN interface.	**set igmp config** *vlan-list* {[**query-interval** *query-interval*] **[igmp-version** *igmpversion*] [**max-resp-time** *max-resp-time*] [**robustness** *robustness*] [**last-mem-int** *last-mem-int*] [**fast-leave** *fast-leave*] [**rtr-alert-checking** *rtr-alert-checking*] [**filter-id** *filter-id*] [**filter-status** {enable | disable}]}
2.	In switch mode, enable IGMP on each VLAN interface.	**set igmp enable** *vlan-list*

Procedure 27-1 Basic IGMP Configuration

Step	Task	Command
3.	In switch mode, enable IGMP querying on each of the VLANs specified in step 2.	**set igmp query-enable** *vlan-list*

For more information on IGMP CLI commands, refer to your device's *CLI Reference Guide*.

Example IGMP Configuration

```
S Chassis(su)->set igmp enable 2, 3
S Chassis(su)->set igmp query-enable 2, 3
```

IGMP Display Commands

Table 27-3 lists Layer 2 IGMP show commands for S-Series devices.

Table 27-3 Layer 2 IGMP Show Commands

Task	Command
Display the status of IGMP on one or more VLANs.	**show igmp enable** *vlan-list*
Display the IGMP query status of one or more VLANs.	**show igmp query** *vlan-list*
Display the action to be taken with multicast frames when the multicast IGMP flow table is full.	**show igmp flow-full-action**
Display IGMP configuration information for one or more VLANs.	**show igmp config** *vlan-list*
Display IGMP information regarding multicast group membership.	**show igmp groups** [**group** *group*] [**vlan-list** *vlan-list*] [**sip** *sip*] [**-verbose**]
Display static IGMP ports for one or more VLANs or IGMP groups.	**show igmp static** *vlan-list* [**group** *group*]
Display the binding of IP protocol id to IGMP classification.	**show igmp protocols**
Display IGMP information for a specific VLAN.	**show igmp vlan** [*vlan-list*]
Display IGMP reporter information.	**show igmp reporters** [**portlist** *portlist*] [**group** *group*] [**vlan-list** *vlan-list*] [**sip** *sip*]
Display IGMP flow information.	**show igmp flows** [**portlist** *portlist*] [**group** *group*] [**vlan-list** *vlan-list*] [**sip** *sip*]
Display IGMP counter information.	**show igmp counters**
Display configuration information for input filters.	**show igmp input-filter** [*filter-id]* [*rule-id*]
Display the action taken when the first frames of a multicast stream are received.	**show igmp unknown-input-action**

Table 27-4 lists Layer 3 IGMP show commands for S-Series devices.

Table 27-4 Layer 3 IGMP Show Commands

Task	Command
Display IGMP information regarding multicast group membership.	**show ip igmp groups**

Table 27-4 Layer 3 IGMP Show Commands

Task	Command
Display multicast-related information about a specific interface or all interfaces.	**show ip igmp interface** [**vlan** *vlan-id*]

Configuring DVMRP

DVMRP Configuration Commands

Table 27-5 lists the DVMRP configuration commands for S-Series devices.

Table 27-5 DVMRP Configuration Commands

Task	Command
Enable or disable DVMRP on an interface.	**ip dvmrp** **no ip dvmrp**
Configure the metric associated with a set of destinations for DVMRP reports.	**ip dvmrp metric** *metric*

Basic DVMRP Configuration

By default, DVMRP is disabled globally on Extreme Networks S-Series devices and attached interfaces.

Procedure 27-2 describes the basic steps to configure DVMRP on S-Series devices. This procedure assumes that the VLANs on which DVMRP will run have been configured and enabled with IP interfaces.

Procedure 27-2 Basic DVMRP Configuration

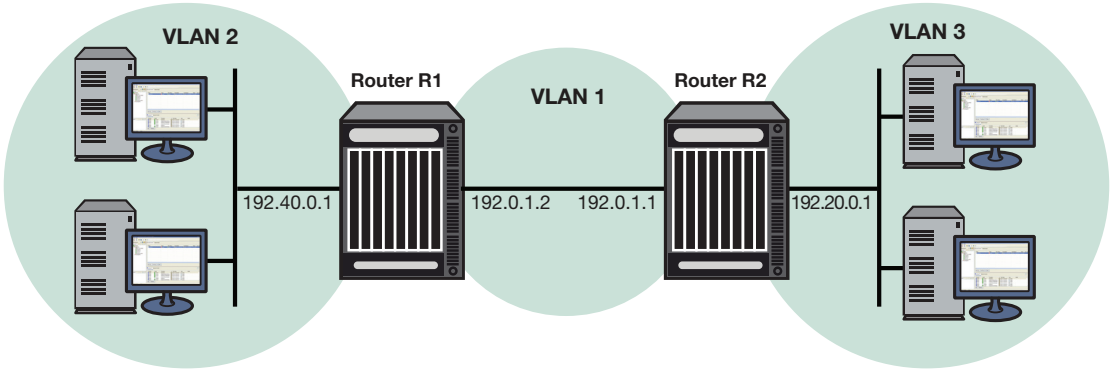
Step	Task	Command
1.	Configure IGMP for each VLAN interface.	**set igmp config** *vlan-list* {[**query-interval** *query-interval*] [**igmp-version** *igmpversion*] [**max-resp-time** *max-resp-time*] [**robustness** *robustness*] [**last-mem-int** *last-mem-int*]}
2.	Enable IGMP on each VLAN interface.	**set igmp enable** *vlan-list*
3.	Enable DVMRP on each of the VLANs specified in step 2.	**ip dvmrp**

Example DVMRP Configuration

Figure 27-7 illustrates the DVMRP configuration of two S-Series devices shown in the example below. This example assumes the following:

- VLANs have been configured and enabled with IP interfaces
- IGMP has been enabled on the VLANs

Figure 27-7 DVMRP Configuration on Two Routers



Router R1 Configuration

For the VLAN 1 interface, which provides connection to Router R2, an IP address is assigned and DVMRP is enabled. For the VLAN 2 interface, which provides connection to the host network, an IP address is assigned and DVMRP is enabled.

```
R1(su)->config
R1(su-config)->interface vlan 1
R1(su-config-intf-vlan.0.1)->ip address 192.0.1.2 255.255.255.0
R1(su-config-intf-vlan.0.1)->ip dvmrp
R1(su-config-intf-vlan.0.1)->no shutdown
R1(su-config-intf-vlan.0.1)->exit
R1(su-config)->interface vlan 2
R1(su-config-intf-vlan.0.2)->ip address 192.40.0.1 255.255.255.0
R1(su-config-intf-vlan.0.2)->ip dvmrp
R1(su-config-intf-vlan.0.2)->no shutdown
R1(su-config-intf-vlan.0.2)->exit
```

Router R2 Configuration

For the VLAN 1 interface, which provides connection to the Router R1, an IP address is assigned and DVMRP is enabled. For the VLAN 3 interface which provides connection to the host network, an IP address is assigned and DVMRP is enabled.

```
R2(su)->config
R2(su-config)->interface vlan 1
R2(su-config-intf-vlan.0.1)->ip address 192.0.1.1 255.255.255.0
R2(su-config-intf-vlan.0.1)->ip dvmrp
R2(su-config-intf-vlan.0.1)->no shutdown
R2(su-config-intf-vlan.0.1)->exit
R2(su-config)->interface vlan 3
R2(su-config-intf-vlan.0.3)->ip address 192.20.0.1 255.255.255.0
R2(su-config-intf-vlan.0.3)->ip dvmrp
R2(su-config-intf-vlan.0.3)->no shutdown
R2(su-config-intf-vlan.0.3)->exit
```

Displaying DVMRP Information

Table 27-6 lists the DVMRP show commands for S-Series devices.

Table 27-6 DVMRP Show Commands

Task	Command
Display information about the routes in the DVMRP routing table.	**show ip dvmrp route**

Table 27-6 DVMRP Show Commands

Task	Command
Display the IP multicast routing table.	**show ip mroute** [*unicast-source-address* | *multicast-group-address*] [**summary**]

Refer to the *Extreme Networks S-Series CLI Reference* for an example of each command's output.

Configuring PIM

PIM Configuration Commands

Table 27-7 lists the PIM configuration commands for Extreme Networks S-Series devices.

Table 27-7 IPv4 PIM Sparse Mode Commands

Task	Command
Enable PIM-SM on a routing interface. Use the **no** command to disable PIM-SM.	**ip pim sparse-mode** **no ip pim sparse-mode**
Enable PIM-SSM in router configuration mode. Use the **no** command to disable PIM-SSM.	**ip pim ssm** {**default** | *group-address group-mask*} **no ip pim ssm** {**default** | *group-address group-mask*}
Enable the router to announce its candidacy as a BootStrap Router (BSR). Use the **no** command to remove the router as a BSR candidate.	**ip pim bsr-candidate** *pim-interface-address* [**priority** *priority*] **no ip pim bsr-candidate**
Set the priority for which a router will be elected as the designated router (DR). Use the **no** command to disable the DR functionality.	**ip pim dr-priority** *priority* **no ip dr-priority**
Set a static rendezvous point (RP) for a multicast group, specifying a specific group or a group-list. Use the **no** command to remove the static RP configuration.	**ip pim rp-address** *rp-address* {*group-address group-mask* | **group-list** *group-list*} **no ip rp-address** *rp-address* {*group-address group-mask* | **group-list** *group-list*}
Enable the router to advertise itself as a PIM candidate rendezvous point (RP) to the BSR specifying either a specific group or a group-list. Use the **no** command to remove the router as an RP candidate.	**ip pim rp-candidate** *pim-interface-address* {*group-address group-mask* | **priority** *priority* | **group-list** *group-list* [**priority** *priority*]} **no ip pim rp-candidate** *pim-interface-address* {*group-address group-mask* | **group-list** *group-list* [**priority** *priority*]}
Enable control of whether static RP configurations will override dynamic RP information learned for IPv4 groups.	**ip pim static-rp-override** **no ip pim static-rp-override**
Filter PIM neighbors by specifying a standard ACL containing neighbors to allow.	**ip pim neighbor-filter** *neighbor-filter* **no ip pim neighbor-filter** *neighbor-filter*
Configure an anycast Rendezvous Points (RP) set member for a multicast group. Use the **no** command to remove the specified anycast member.	**ip pim anycast-rp** *anycast-address peer-address* **no ip anycast-rp** *anycast-address peer-address*
Set the multicast graceful-restart period, which is the period of time in which a restarting router and its neighbors can continue to forward multicast packets during the failover.	**ip pim graceful-restart** [**period** *value*] **no ip pim graceful-restart**

Table 27-7 IPv4 PIM Sparse Mode Commands

Task	Command
Set PIM multicast to either load share over ECMP paths or have a single deterministic next hop for ECMP paths.	**ip pim multipath** {**hash** | **highest-nexthop**} **no ip pim multipath** {**hash** | **highest-nexthop**}

Table 27-8 lists the PIM IPv6 sparce mode configuration commands for Extreme Networks S-Series devices.

Table 27-8 IPv6 PIM Sparce Mode Commands

Task	Command
Enable PIM-SM on a routing interface. Use the **no** command to disable PIM-SM.	**ipv6 pim sparse-mode** **no ipv6 pim sparse-mode**
Optionally enable PIM-SSM in router configuration mode. Use the no command to disable PIM-SSM	**ipv6 pim ssm** {**default** | *group-address/length*} **no ipv6 pim ssm** {**default** | *group-address*/length}
Enable the router to announce its candidacy as a BootStrap Router (BSR). Use the **no** command to remove the router as a BSR candidate.	**ipv6 pim bsr candidate bsr** *interface-address* [**priority** *priority*] **no ipv6 pim bsr candidate bsr** *interface-address*
Set the priority for which a router will be elected as the designated router (DR). Use the **no** command to disable the DR functionality.	**ipv6 pim dr-priority** *priority* **no ipv6 pim dr-priority** *priority*
Set a static rendezvous point (RP) for a multicast group, specifying a specific group or a group-list. Use the **no** command to remove the static RP configuration.	**ipv6 pim rp-address** *rp-address* **group-list** *group-list* **no ipv6 pim rp-address** *rp-address* **group-list** *group-list*
Enable the router to advertise itself as a PIM candidate rendezvous point (RP) to the BSR specifying a group-list. Use the **no** command to remove the router as an RP candidate.	**ipv6 pim bsr candidate rp** *pim-interface-address* {[**group-list** *group-list*] [**priority** *priority*]} **no ipv6 pim bsr candidate bsr** *pim-interface-address* {[**group-list** *group-list*] [**priority** *priority*]}
Enable control of whether static RP configurations will override dynamic RP information learned for IPv6 groups.	**ipv6 pim static-rp-override** **no ipv6 pim static-rp-override**
Filter PIM neighbors by specifying a standard ACL containing neighbors to allow.	**ipv6 pim neighbor-filter** *neighbor-filter* **no ipv6 pim neighbor-filter** *neighbor-filter*
Configure an anycast Rendezvous Points (RP) set member for a multicast group. Use the **no** command to remove the specified anycast member.	**ipv6 pim anycast-rp** *anycast-address peer-address* **no ipv6 pim anycast-rp** *anycast-address peer-address*
Set the multicast graceful-restart period, which is the period of time in which a restarting router and its neighbors can continue to forward multicast packets during the failover.	**ipv6 pim graceful-restart** [**period** *value*] **no ipv6 pim graceful-restart**
Set PIM multicast to either load share over ECMP paths or have a single deterministic next hop for ECMP paths.	**ipv6 pim multipath** {**hash** | **highest-nexthop**} **no ipv6 pim multipath** {**hash** | **highest-nexthop**}

Table 27-9 lists the PIM dense mode configuration commands for Extreme Networks S-Series devices.

Table 27-9 PIM Dense Mode Commands

Task	Command
Enable PIM-DM on a routing interface. Use the **no** command to disable PIM-DM.	**ip** | **ipv6 pim dense-mode** **no ip** | **ipv6 pim dense-mode**
Optionally set the interval between PIM dense mode state refresh messages.	**ip** | **ipv6 pim state-refresh origination-interval** *interval* **no ip** | **ipv6 pim state-refresh origination-interval** *interval*
Set PIM multicast to either load share over ECMP paths or have a single deterministic next hop for ECMP paths.	**ip** | **ipv6 pim multipath** {**hash** | **highest-nexthop**} **no ip** | **ipv6 pim multipath** {**hash** | **highest-nexthop**}

Basic PIM-SM Configurations

The following describes a basic PIM configuration. PIM-SSM is a simplified version of PIM-SM. PIM-SSM does not require either a BSR or an RP. In a PIM-SSM configuration there is no need for a candidate BSR, a candidate RP, or a static RP. In a mixed PIM-SSM and PIM-SM configuration, the candidate BSR, candidate RP, and the static RP need only be configured for the non-PIM-SSM address ranges.

By default, PIM-SM and PIM-SSM are disabled globally on S-Series devices and attached interfaces. Basic PIM configuration includes the following steps:

1. Creating and enabling VLANs with IP interfaces.
2. Configuring the underlying unicast routing protocol (for example, OSPF).
3. Enabling IGMP on the VLANs. Enable IGMP for interfaces with IGMP reporters. Enable IGMP version 3 on interfaces using PIM-SSM
4. Configuring PIM-SM and/or PIM-SSM on the VLANs.

Procedure 27-3, which describes the basic steps the configure PIM-SM on an S-Series device, assumes the following:

- VLANs have been configured and enabled with IP interfaces.
- The unicast routing protocol has been configured.
- IGMP has been enabled on the devices and VLANs that will be connected with hosts. For information on enabling IGMP, see "Configuring IGMP" on page 27-20.

Note: PIM-SSM and PIM-SM can coexist in a network. A candidate BSR, candidate RP, and static RP addresses can be configured in a PIM-SSM configuration, but are not required. Along with IGMP, PIM-SSM must be enabled on the source host interface and be reachable by the PIM-SSM destination addresses.

Procedure 27-3 Basic PIM Sparse Mode Configuration

Step	Task	Command(s)
1.	If desired, change the DR priority of one or more interfaces on the Extreme Networks S-Series router from the default value of 1 in interface configuration mode. The highest priority PIM router on a shared network is elected the DR for that network.	**IPv4:** **ip pim dr-priority** *priority* **IPv6:** **ipv6 pim dr-priority** *priority*
2.	If the dynamic BSR RP set distribution method is used on the network, configure at least one PIM router as a candidate BSR in interface configuration mode. Note that the Extreme Networks S-Series router does not act as a BSR without being explicitly configured to do so.	**IPv4:** **ip pim bsr-candidate** *pim-interface* [**priority** *priority*] **IPv6:** **ipv6 pim bsr candidate bsr** *interface-address* [**priority** *priority*]
3.	If the dynamic BSR RP set distribution method will be used on the network, configure at least one PIM router as a Candidate Rendezvous Point in global configuration mode. Note that the Extreme Networks S-Series router does not act as an RP without being explicitly configured to do so.	**IPv4:** **ip pim rp-candidate** *pim-interface group-address group-mask* [**priority** *priority*] **IPv6:** **ipv6 pim bsr candidate rp** *pim-interface-address* {[**group-list** *group-list*] [**priority** *priority*]}
4.	If static RP set distribution is desired, configure the static RP set information in global configuration mode. The RP set information must be the same on all PIM routers in the network. **Note:** Static RP set distribution cannot be combined with BSR RP set distribution in the same PIM domain. Routers with statically configured RP set information discard RP set information learned from a BSR.	**IPv4:** **ip pim rp-address** *rp-address group-address group-mask* **IPv6:** **ipv6 pim rp-address** *rp-address* **group-list** *group-list*
5.	Configure PIM-SM and/or PIM/SSM on the S-Series router that will run PIM-SM. PIM-SM is configured on the interface. PIM-SSM is globally configured in global configuration mode. IPv6 PIM-SSM is enabled on the device by default with an address range of FF3E:0000/32.	**IPv4:** **ip pim sparse-mode** **ip pim ssm** **IPv6:** **ipv6 pim sparse-mode**

PIM IPv4 and IPv6 Display Commands

Table 27-10 lists the PIM IPv4 and IPv6 display commands for Extreme Networks S-Series devices.

Table 27-10 PIM IPv4 and IPv6 Display Commands

Task	Command
Display summary tables of PIM interfaces, neighbors, BSR, and group-to-RP mappings.	**show** {**ip** | **ipv6**} **pim**

Table 27-10 PIM IPv4 and IPv6 Display Commands

Task	Command
Display RP anycast information for all or a specified RP. (PIM-SM only).	**show** {**ip** | **ipv6**} **pim anycast-rp** [**rp-address** *rp-address*]
Display BootStrap Router (BSR) information. (PIM-SM only).	**show** {**ip** | **ipv6**} **pim bsr** [**detail**]
Display information about PIM interfaces that are currently up (not shutdown).	**show** {**ip** | **ipv6**} **pim interface** [*ifName*] [**brief**] [**detail**] [**statistics**]
Display the PIM multicast route (*,G and S,G) table.	**show** {**ip** | **ipv6**} **pim mrt** [**source** *source* | **group** *group*] [**interface**] [**detail**] [**brief**] [**summary**]
Display the PIM multicast route (*,G and S,G) table by type.	**show** {**ip** | **ipv6**} **mrt type** {**all** | **s-g** | **star-g**} [**source** *source* | **group** *group*] [**interface**] [**detail**] [**brief**] [**type** {**all** | **s-g** | **star-g**}] [**summary**]
Display information about discovered PIM neighbors.	**show** {**ip** | **ipv6**} **pim neighbor** [*ifName*] [**brief**] [**detail**] [**statistics**]
Display the active rendezvous points (RPs) that are cached with associated multicast routing entries. (PIM-SM only).	**show** {**ip** | **ipv6**} **pim rp** [**mapping**]
Display the rendezvous point (RP) selected for a specified group. (PIM-SM only).	**show** {**ip** | **ipv6**} **pim rp-hash** *group-address*
Display PIM statistics for this device.	**show** {**ip** | **ipv6**} **pim statistics**
Display the multicast routing table.	**show** {**ip** | **ipv6**} **mroute** [**source** *source* | **group** *group* | **interface** *interface*] [**brief**] [**summary**]
Display the multicast forwarding cache that was used to program the hardware flow.	**show** {**ip** | **ipv6**} **mcache** [**group** *group* | **source** *source*] [**interface**] [**verbose** | **brief** | **summary**] [**statistics**] [**-wide**]

Refer to the *Extreme Networks S-Series CLI Reference* for a description of the output of each command.

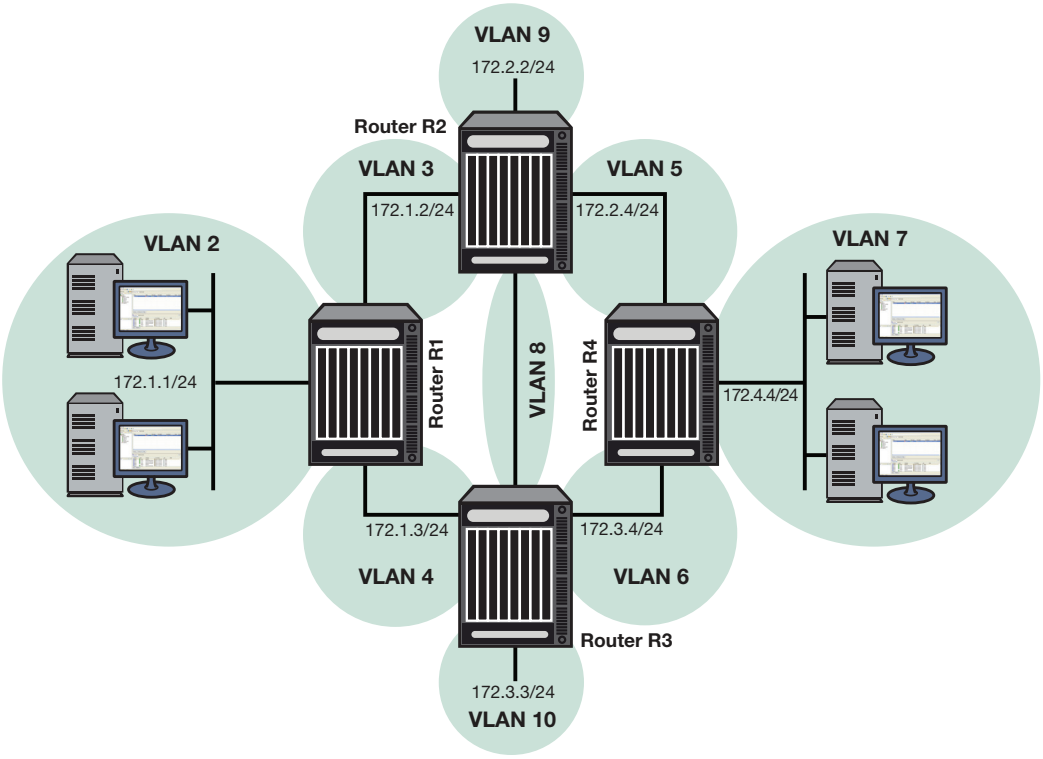
Example PIM Configuration

Figure 27-8 illustrates the PIM of four S-Series routers. For a PIM-DM configuration:

- Configure interfaces and enable IGMP as shown in the PIM-SM example script below
- Enable dense mode on all interfaces using the **ip pim dense-mode** command
- Optionally change the state refresh interval, using the **ip pim state-refresh origination-interval** command

The PIM-SM configuration is shown in the example scripts below. PIM-SM configuration includes configuring a preferred and a backup BSR for the topology, as well as two RPs for specific multicast groups and a backup RP for all groups.

Figure 27-8 PIM-SM Configuration with Bootstrap Router and Candidate RPs



Router R1 Configuration

On this router, IGMP is enabled on VLAN 2, which connects to hosts, and PIM-SM is enabled on all interfaces. IGMP is used to determine host group membership on directly attached subnets. Note that IGMP is enabled in switch mode on S-Series routers.

VLAN 2 is configured as the backup candidate RP for all multicast groups by using the default RP priority of 192. Note that the C-RP with the smallest priority value is elected.

Alternatively, you could configure a loopback interface as a candidate RP, to avoid the dependency on a particular interface.

```
R1(su-config)->router id 1.1.1.1
R1(su-config)->interface vlan 2
R1(su-config-intf-vlan.0.2)->ip address 172.1.1.1 255.255.255.0
R1(su-config-intf-vlan.0.2)->no shutdown
R1(su-config-intf-vlan.0.2)->exit

R1(su)->set igmp enable 2
R1(su)->set igmp enable 3
R1(su)->set igmp enable 4
R1(su)->set igmp query-enable 2

R1(su-config)->ip pim rp-candidate 172.1.1.1 224.0.0.0 240.0.0.0
R1(su-config)->interface vlan 2
R1(su-config-intf-vlan.0.2)->ip pim sparse-mode
R1(su-config-intf-vlan.0.2)->exit
R1(su-config)->interface vlan 3
R1(su-config-intf-vlan.0.3)->ip address 172.1.2.1 255.255.255.0
R1(su-config-intf-vlan.0.3)->no shutdown
R1(su-config-intf-vlan.0.3)->ip pim sparse-mode
R1(su-config-intf-vlan.0.3)->exit
```

```
R1(su-config)->interface vlan 4
R1(su-config-intf-vlan.0.4)->ip address 172.1.3.1 255.255.255.0
R1(su-config-intf-vlan.0.4)->no shutdown
R1(su-config-intf-vlan.0.4)->ip pim sparse-mode
R1(su-config-intf-vlan.0.4)->exit
```

Router R2 Configuration

On this router, PIM-SM is enabled on all interfaces. VLAN 9 is configured as a candidate BSR and is assigned a priority higher than the default of 0. Note that the C-BSR with the largest priority value is elected.

VLAN 9 is also configured as a candidate RP for the multicast group 224.2.2.0/24. Its priority is set to 2, which will most likely make it the elected RP for that particular group, since the C-RP with the smallest priority value is elected. (Note that Router R3 has an RP candidate priority value of 3 for that group.)

Again, alternatively, you could configure a loopback interface as a candidate BSR or RP, to avoid the dependency on a particular interface.

```
R2(su)->set igmp enable 3
R2(su)->set igmp enable 9
R1(su)->set igmp enable 8
R1(su)->set igmp enable 5

R2(su-config)->router id 1.1.1.2
R2(su-config)->ip pim bsr-candidate vlan 9 priority 2

R2(su-config)->interface vlan 3
R2(su-config-intf-vlan.0.3)->ip address 172.1.2.2 255.255.255.0
R2(su-config-intf-vlan.0.3)->no shutdown
R2(su-config-intf-vlan.0.3)->ip pim sparse-mode
R2(su-config-intf-vlan.0.3)->exit

R2(su-config)->interface vlan 9
R2(su-config-intf-vlan.0.9)->ip address 172.2.2.2 255.255.255.0
R2(su-config-intf-vlan.0.9)->no shutdown
R2(su-config-intf-vlan.0.9)->ip pim sparse-mode
R2(su-config-intf-vlan.0.9)->exit
R2(su-config)->ip pim rp-candidate 172.2.2.2 224.2.2.0 255.255.255.0 priority 2

R2(su-config)->interface vlan 8
R2(su-config-intf-vlan.0.8)->ip address 172.2.3.2 255.255.255.0
R2(su-config-intf-vlan.0.8)->no shutdown
R2(su-config-intf-vlan.0.8)->ip pim sparse-mode
R2(su-config-intf-vlan.0.8)->exit

R2(su-config)->interface vlan 5
R2(su-config-intf-vlan.0.5)->ip address 172.2.4.2 255.255.255.0
R2(su-config-intf-vlan.0.5)->no shutdown
R2(su-config-intf-vlan.0.5)->ip pim sparse-mode
R2(su-config-intf-vlan.0.5)->exit
```

Router R3 Configuration

On this router, PIM-SM is enabled on all interfaces. VLAN 10 is configured as a backup candidate BSR, by leaving its priority at the default of 0.

VLAN 10 is also configured as a backup candidate RP for multicast group 224.2.2.0/24, by setting its priority value slightly higher (3) than the priority configured on R2 for the same group (2) (since the C-RP with the smallest priority value is elected).

```
R3(su)->set igmp enable 4
```

```
R3(su)->set igmp enable 8
R3(su)->set igmp enable 10
R3(su)->set igmp enable 6
R3(su)->configure

R3(su-config)->router id 1.1.1.3
R3(su-config)->ip pim bsr-candidate vlan 10

R3(su-config)->interface vlan 4
R3(su-config-intf-vlan.0.4)->ip address 172.1.3.3 255.255.255.0
R3(su-config-intf-vlan.0.4)->no shutdown
R3(su-config-intf-vlan.0.4)->ip pim sparse-mode
R3(su-config-intf-vlan.0.4)->exit

R3(su-config)->interface vlan 8
R3(su-config-intf-vlan.0.8)->ip address 172.2.3.3 255.255.255.0
R3(su-config-intf-vlan.0.8)->no shutdown
R3(su-config-intf-vlan.0.8)->ip pim sparse-mode
R3(su-config-intf-vlan.0.8)->exit

R3(su-config)->interface vlan 10
R3(su-config-intf-vlan.0.10)->ip address 172.3.3.3 255.255.255.0
R3(su-config-intf-vlan.0.10)->no shutdown
R3(su-config-intf-vlan.0.10)->ip pim sparse-mode
R3(su-config-intf-vlan.0.10)->exit
R3(su-config)->ip pim rp-candidate 172.3.3.3 224.2.2.0 255.255.255.0 priority 3

R3(su-config)->interface vlan 6
R3(su-config-intf-vlan.0.6)->ip address 172.3.4.3 255.255.255.0
R3(su-config-intf-vlan.0.6)->no shutdown
R3(su-config-intf-vlan.0.6)->ip pim sparse-mode
R3(su-config-intf-vlan.0.6)->exit
```

Router R4 Configuration

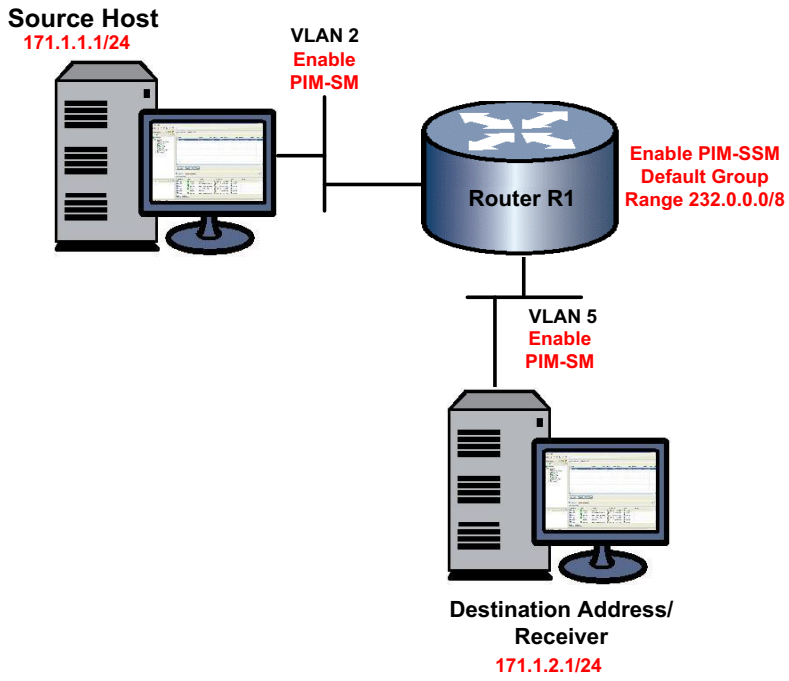
This router does not play any special role in PIM-SM, except that it has hosts directly connected to it. IGMP is enabled on the interface that connects to hosts and PIM-SM is enabled on all interfaces.

```
R3(su)->set igmp enable 5
R3(su)->set igmp enable 6
R3(su)->set igmp enable 7
R3(su)->configure
R4(su-config)->router id 1.1.1.4
R4(su-config)#interface vlan 5
R4(su-config-intf-vlan.0.5)->ip address 172.2.4.4 255.255.255.0
R4(su-config-intf-vlan.0.5)->no shutdown
R4(su-config-intf-vlan.0.5)->ip pim sparse-mode
R4(su-config-intf-vlan.0.5)->exit

R4(su-config)->interface vlan 6
R4(su-config-intf-vlan.0.6)->ip address 172.3.4.4 255.255.255.0
R4(su-config-intf-vlan.0.6)->no shutdown
R4(su-config-intf-vlan.0.6)->ip pim sparse-mode
R4(su-config-intf-vlan.0.6)->exit

R4(su-config)->interface vlan 7
R4(su-config-intf-vlan.0.7)->ip address 172.4.4.4 255.255.255.0
R4(su-config-intf-vlan.0.7)->no shutdown
R4(su-config-intf-vlan.0.7)->ip pim sparse-mode
R4(su-config-intf-vlan.0.7)->exit
```

Example PIM-SSM Configuration

Figure 27-9 illustrates the PIM-SSM configuration of a single router shown in the example scripts below. PIM-SSM is enabled on router R1 with the default group range of 232.0.0.0/8. VLANs connected to the source host and receiver are configured on the router. PIM-SM and IGMP are enabled on all interfaces. IGMP query is enabled on the receiver interface.

Figure 27-9 PIM-SSM Configuration



Router R1 Configuration

On this router:

- Enable PIM-SSM with the default group range
- Configure VLAN 2 with the source host IP address 171.1.1.1/24, and enable PIM-SM on the interface
- Configure VLAN 5 with the receiver IP address 171.1.2.1/24, and enable PIM-SM on the interface
- Enable IGMP version 3 on VLAN 2 and VLAN 5. IGMP is used to determine host group membership on directly attached subnets. PIM-SSM requires IGMP version 3. Note that IGMP version 2 is enabled by default in switch mode on S-Series routers.
- Enable IGMP querying on the receiver interface (VLAN 5)

```
R1(su-config)->router id 1.1.1.1
R1(su-config)->ip pim ssm default
R1(su-config)->interface vlan 2
R1(su-config-intf-vlan.0.2)->ip address 171.1.1.1 255.255.255.0
R1(su-config-intf-vlan.0.2)->ip pim sparse-mode
R1(su-config-intf-vlan.0.2)->no shutdown
R1(su-config-intf-vlan.0.2)->exit
R1(su-config)->interface vlan 5
R1(su-config-intf-vlan.0.5)->ip address 171.1.2.1 255.255.255.0
```

```
R1(su-config-intf-vlan.0.2)->ip pim sparse-mode
R1(su-config-intf-vlan.0.5)->no shutdown
R1(su-config-intf-vlan.0.5)->exit
R1(su-config)->exit
R1(su)->set igmp enable 2,5
R1(su)->set igmp query-enable 5
R1(su)->set igmp config 2,5 igmp-version 3
```

28

MSDP Configuration

This document describes the Multicast Source Discovery Protocol (MSDP) and its configuration on S-Series devices. MSDP is based on multicast technology and Protocol Independent Multicast Sparse Mode (PIM-SM), described in Chapter 27, **Multicast Configuration**.

Note: This feature requires licenses for the following S-Series platforms:

- For the S130 platform: **S-EOS-L3-S130** (S130 class I/O and SSA130)
- For the S150 platform: **S-EOS-L3-S150** (S150 class I/O and SSA150)

The S155/S140/S180 platforms and SSA180 platform are fully entitled to all features and do not require a license.

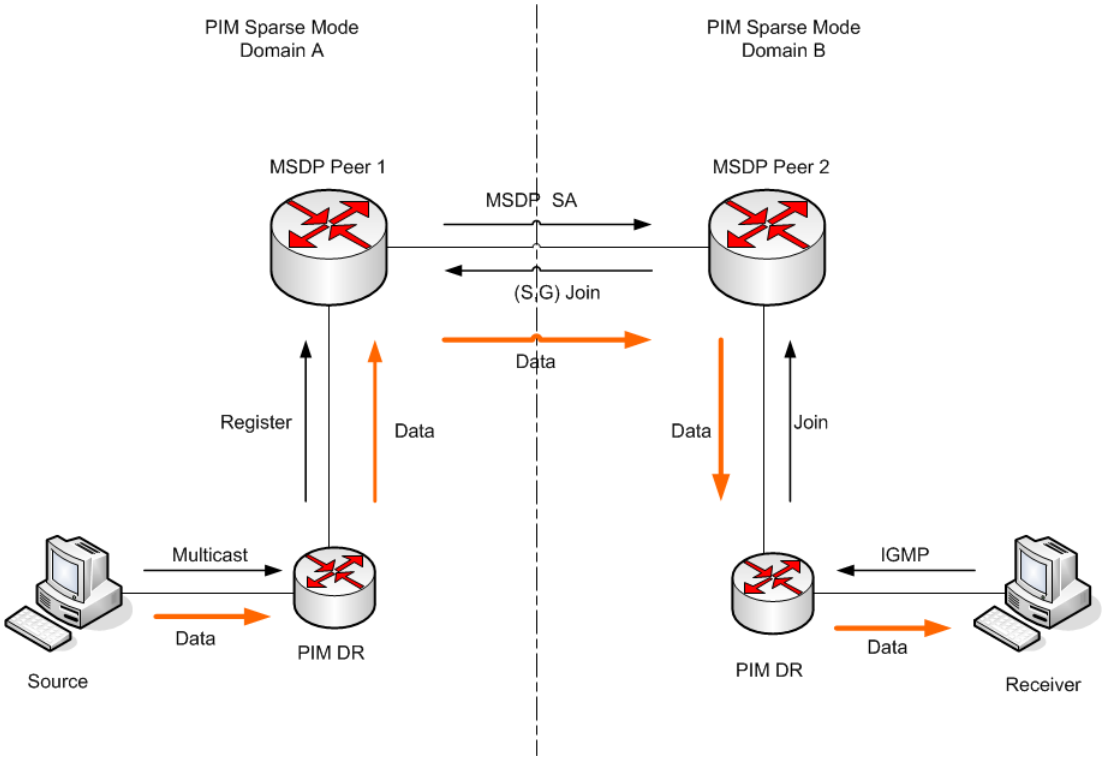
MSDP Overview

Multicast Source Discovery Protocol (MSDP) connects multiple Protocol Independent Multicast sparse mode (PIM-SM) domains. Where standard multicast distribution is limited to a single PIM domain, MSDP establishes Transmission Control Protocol (TCP) connections between rendezvous points (RPs) in different PIM-SM domains and allows multicast sources to be known to all RPs. Once an RP knows a multicast source from a different domain, it can perform multicast distribution across both PIM-SM domains.

MSDP uses TCP connections to establish MSDP peering relationships. The TCP connections between PIM RPs depend on an underlying routing system such as BGP or MBGP for inter-domain operation. MSDP and PIM exchange information about multicast sources. When MSDP learns a new source from an incoming Source-Active (SA) message, it notifies PIM of the new multicast source and multicast group. If there is an entry for that group in the PIM mroute entry, then a join is activated.

Figure 28-1 illustrates MSDP operating between two RPs in different PIM-SM domains, serving as MSDP peers. PIM uses MSDP to register a source with the RPs of other domains.

Figure 28-1 Rendezvous Points as MSDP Peers



When an MSDP peer (MSDP Peer 1 in Figure 28-1) receives a PIM register message from a PIM designated router (DR) within its own domain (PIM Domain A in Figure 28-1), MSDP is notified of multicast source information when MSDP is configured on this router. MSDP originates a Source-Active (SA) message which it immediately forwards to all MSDP peers. The SA message includes the source, the group, and the address of the RP or the originator ID.

Each MSDP peer receives and forwards the SA message away from the originating RP. In order to avoid SA flooding, MSDP forwards SA messages only from an MSDP Reverse Path Forwarding (RPF) peer. The RPF peer is the next hop toward the originating RP of the SA message (as defined in the BGP or MBGP routing table for the domain). If the MSDP peer receives the same SA message from a non-RPF peer toward the originating RP, it drops the message. Otherwise, it forwards the message on to all its MSDP peers except for the RPF peer.

When MSDP receives an SA message from an MSDP peer, it passes all multicast source information to the RP (RP2 in Figure 28-1) configured in this router. The RP2 determines if any group members in this PIM domain (PIM Domain B in Figure 28-1) are interested in the group the SA message describes. If so, the RP2 triggers an (S, G) join toward the RP1 in the domain A. Multicast distribution is then established across multiple domains. RP1 in PIM Domain A forwards this (S,G) multicast flow to RP2 in PIM domain B, which distributes to all receivers in domain B according to its multicast routing table.

Note: It is not a requirement for RPs/MSDP peers to be at the edge of domains, as shown in Figure 28-1. Non-RP PIM routers may be in the path between the RPs in their respective domains. MSDP messages and RP-issued S,G joins simply traverse these routers.

Source Active Messages

In addition to keep-alive messages, MSDP peers send each other SA messages. SA messages convey group addresses, source addresses, and RP addresses among MSDP peers so that the RPs in different domains learn active source information from outside. MSDP SA messages are sent out periodically by the originating RP.

MSDP routers cache SA messages they receive from their MSDP peers. They also cache SA messages received from RPs within their own PIM domain. The SA-cache can reduce message storms. Entries in cache time out after 135 seconds and are removed from the cache. SA messages can also be cleared manually.

MSDP uses filters to control SA source information. Inbound filters determine if an SA message is to be accepted or which peer it is to be accepted from. If no SA filters are configured, the MSDP router receives all SA messages from its RPF peers and forwards them to the other MSDP peers. Outbound policy determines which peer the SA messages are sent to and for which source/group to advertise. The default is to advertise any source to any MSDP peer. Extended access control lists are used in SA filters to filter incoming SA messages and to prevent outgoing messages from being forwarded.

MSDP Mesh Groups

When a group of MSDP speakers in a domain are meshed together, they can be configured as a mesh-group. An MSDP mesh group reduces flooding by ensuring that a mesh-group member does not have to forward SA messages to other group members, because the originator will forward it to all group members. If a member R of a mesh-group M receives an SA message from an MSDP peer that is also a member of mesh-group M, R does not flood or forward these messages. If the SA message passes the peer-RPF check, then R forwards the SA message to all members of mesh-group M.

If a member R of a mesh-group M receives an SA message from an MSDP peer that is not a member of mesh-group M, then R does the peer-RPF check first for the SA message. If the peer-RPF check fails, R drops it. If the SA message passes the peer-RPF check, then R forwards the SA message to all members of mesh-group M.

Each MSDP peer in the mesh group must be configured as such (a peer in the mesh group).

Configuring MSDP

Procedure 28-1 lists the MSDP configuration commands for S-Series devices.

Procedure 28-1 MSDP Configuration

Step	Task	Command
1.	Enable MSDP by configuring an MSDP peer to the local router. To delete an MSDP peer, use the **no** form of this command.	**ip msdp peer** *peer-address* **connect-source** *type-number* [**remote-as** *as-number*]
2.	Configure an Originator ID for the router that originates SA messages. This configures the RP address in SA messages to be the address of the originating router's interface. To remove the Originator ID, use the **no** form of this command.	**ip msdp originator-id** *interface-id*
3.	Configure MSDP filters that enable MSDP to control how multicast sources and groups are learned and advertised. Configure an incoming filter list (**sa-filter in**) for Source-Active messages received from a specified MSDP peer. Configure an outgoing filter list (**sa-filter out**) for Source-Active messages sent to a specified MSDP peer. To remove a filter, use the **no** form of this command.	**ip msdp sa-filter in** *peer-address* **[list** *access-list-name***]** **ip msdp sa-filter out** *peer-address* **[list** *access-list-name***]**

Procedure 28-1 MSDP Configuration (continued)

Step	Task	Command
4.	(Optional) Configure a BSR border router to limit BSR messages in a PIM domain. BSR messages from other PIM domains can cause a wrong RP selection for this domain. This command prevents this border router from receiving BSR messages outside the PIM domain and sending BSR messages out. To remove the border router, use the **no** form of this command.	**ip pim bsr-border**
5.	Configure an MSDP mesh group by running this command for each MSDP peer in the group. To remove an MSDP peer from the mesh group, use the **no** form of this command.	**ip msdp mesh-group** *group-name peer-address*
6.	Administratively shut down an MSDP peer. When a peer is shut down, the TCP connection is terminated and not restarted.	**ip msdp shutdown** *peer-address*
7.	Clear the MSDP TCP connection to the specified MSDP peer and reset all MSDP message counters.	**clear ip msdp peer** [*peer-address*]
8.	Clear the SA cache of all entries.	**clear ip msdp sa-cache**
9.	Clear statistics counters for one or all of the MSDP peers.	**clear ip msdp statistics** [*peer-address*]

MSDP Display Commands

Table 28-1 lists MSDP show commands for S-Series devices.

Table 28-1 MSDP Show Commands

Task	Command
Display detailed information about an MSDP peer.	**show ip msdp peer [***peer-address***]**
Display (S, G) state learned from MSDP peers.	**show ip msdp sa-cache**
Display all MSDP peer status.	**show ip msdp summary**
Display local SA messages (those generated on this router).	**show ip msdp sa local**
Display all messages, including local and learned.	**show ip msdp sa all**

For more information on MSDP CLI commands, refer to your device's *CLI Reference Guide*.

Example MSDP Configuration

The following shows an example S-Series device configuration that sets up an inter-domain MSDP peering relationship.

Define the loop back or VLAN interfaces for PIM RP / MSDP peers. In the example, loop.0.1 is used as PIM RP/ MSDP peer.

[On router A:]

```
interface loop.0.1
  ip address 121.1.1.1 255.255.255.0 primary
  no shutdown
```

```
  exit
```

[On router B:]

```
interface loop.0.1
  ip address 122.2.2.2 255.255.255.0 primary
  no shutdown
  exit
```

Define a PIM RP for each domain.

[On router A:]

```
ip pim bsr-candidate 121.1.1.1
 ip pim rp-candidate 121.1.1.1 224.0.0.0 240.0.0.0
```

[On router B:]

```
ip pim bsr-candidate 122.2.2.2
 ip pim rp-candidate 122.2.2.2 224.0.0.0 240.0.0.0
```

Configure BGP for each domain.

[On router A:]

```
router bgp 5100
  bgp router-id 121.1.1.1
  neighbor 122.2.2.2 remote-as 5200
  neighbor 122.2.2.2 ebgp-multihop 10
  neighbor 122.2.2.2 update-source 121.1.1.1
  redistribute connected
  exit
```

[On router B:]

```
router bgp 5200
  bgp router-id 122.2.2.2
  neighbor 121.1.1.1 remote-as 5100
  neighbor 121.1.1.1 ebgp-multihop 10
  neighbor 121.1.1.1 update-source 122.2.2.2
  redistribute connected
  exit
```

Configure OSPF for each domain so that unicast routes work for PIM. This is basic for PIM.

Configure MSDP peers between the two domains.

[On router A:]

```
ip msdp peer 122.2.2.2 connect-source loop.0.1  remote-as 5200
```

[On router B:]

```
ip msdp peer 121.1.1.1 connect-source loop.0.1  remote-as 5100
```

Configuring Anycast RP in MSDP

Anycast RP is a solution for fast convergence of RP routing in the event of an RP router failure. All members of an Anycast RP set are registered peers, and when a source registers with one RP in the set, it sends an SA message to the other RPs in the set so that they all have the available information about active sources.

Anycast RP has been available for PIM RPs in Extreme Networks, as described in "Anycast-RP" on page 27-15. With MSDP configured in your network, the Anycast RP solution can be applied to MSDP RPs in the same way. Please note that MSDP Anycast RP and PIM Anycast RP are mutually exclusive, there can be only one or the other Anycast RP solution operating in a domain. If you already have PIM Anycast RP configured and you want to employ MSDP Anycast RP, you must disable the PIM Anycast RP before configuring and enabling MSDP Anycast RP.

The following is an example configuration of router A and router B as Anycast RPs in an MSDP domain.

Define a loop back interface as the anycast RP address for each router. In the example, loop.0.8 is used as the anycast RP address.

```
interface loop.0.8
  ip address 8.8.8.8 255.255.255.255 primary
  no shutdown
```

Define another loop back interface or VLAN interface to be MSDP peer address. In the example, loop.0.1 is used as the MSDP peer address for both routers.

[On router A:]

```
interface loop.0.1
  ip address 121.1.1.1 255.255.255.0 primary
  no shutdown
  exit
```

[On router B:]

```
interface loop.0.1
  ip address 122.2.2.2 255.255.255.0 primary
  no shutdown
  exit
```

Define another loop back interface or VLAN interface to be MSDP peer address. In the example, loop.0.1 is used as the MSDP peer address for both routers.

[On router A:]

```
interface loop.0.1
  ip address 121.1.1.1 255.255.255.0 primary
  no shutdown
  exit
```

[On router B:]

```
interface loop.0.1
  ip address 122.2.2.2 255.255.255.0 primary
  no shutdown
  exit
```

Configure MSDP peer and originator-ID.

[On router A:]

```
ip msdp originator-id loop.0.1
 ip msdp peer 122.2.2.2 connect-source loop.0.1
```

[On router B:]

```
ip msdp originator-id loop.0.1
 ip msdp peer 121.1.1.1 connect-source loop.0.1
```

Define an RP address for the whole PIM-SM domain.

```
ip pim rp-address 8.8.8.8 224.0.0.0 240.0.0.0
```

Verify that unicast protocols such as OSPF or BGP work. This is necessary for all PIM-SM.

29

Multi-Topology Configuration

This chapter describes Multi-Topology configuration on S-Series devices. Multiple topologies are particularly useful in multicast and that is the focus of the current implementation. Multicast and Protocol Independent Multicast Sparse Mode (PIM-SM) are described in Chapter 27, **Multicast Configuration**.

Multiple Topology Overview

Separate routing topologies provide the capability to route different types of traffic on different paths through the network. This capability can be used, for example, to route voice and video traffic differently: video traffic could be routed using 10 Gb connections, while voice traffic might be routed on 1 Gb connections.

The S-Series router software has the capability to accept user-defined topologies for select routing purposes, such as routing of multicast traffic. A separate multicast topology can be configured for PIM to use a specific routing protocol such as BGP, OSPF, or OSPFv3. As well as the default “base” topology, two additional multicast topologies (one for each address family, IPv4 and/or IPv6) may be configured for each VRF on the device. A separate multicast topology has route tables for all interfaces configured for direct connection to the VRF, as well as any routes provided by the routing protocol(s) activated in that topology. Additional topologies, such as a unicast or multiple multicast topologies, cannot be configured in any VRF.

Each VRF on a device has a base topology that:

- is always present
- cannot be removed
- routes to all active interfaces in the VRF
- includes additional topologies as subsets of the base

A multicast topology is therefore a subset of the base topology and is not completely separate from it. One multicast topology may be configured per address family (IPv4, IPv6), per VRF. When a new topology is created, a separate route table and route table manager (RTM) instance is created for it. Within a multicast topology, PIM interacts only with the route table in that VRF for the configured address family. Once a topology has been created, it is available for use by all protocols that support multi-topology (besides PIM, currently limited to BGP, OSPF and OSPFv3). Each protocol must be separately configured to use the new multicast topology.

Configured OSPF and OSPFv3 parameters (such as timer values and authentication) apply to all instances of that protocol active on the interface for which they are configured, and override any values set in the global configuration of the protocol.

A multicast topology must be created before any routing protocol is configured to use it. By default, routing protocols operate in the base topology and must be explicitly configured to participate in a multicast topology. For multiple topologies using OSPF protocols, it is necessary to configure a separate instance of the protocol for each topology. When an instance of an OSPF protocol is activated, it joins the base topology by default. It remains joined to the base topology even when joined to a multicast topology, unless explicitly removed.

In any VRF there are only the base topology and an IPv4 multicast and/or IPv6 multicast topologies. If you create a process and specify a new multicast topology for it (for example, “address-family ipv4 multicast"), it is then in both unicast and multicast topologies. This is because its route table is still joined to the base (unicast), but now is configured to receive multicast routes from the base as well. To make the process be in the multicast topology only, you must also enter a "no address-family ipv4 unicast" command within that process. Once a multicast topology has been created, you can configure BGP and OSPF or OSPFv3 to join that topology.

In regards to OSPF use in Multi-Topology: you can run a single instance of OSPF connected to the unicast and multicast topology, or you can run multiple instances of OSPF each connected to either a unicast or multicast topology. Since unicast is the default topology you must explicitly disconnect from it if you wish to have an instance connected only to the multicast. It is also possible to have the same interface connect to multiple OSPF processes with the addition of an instance number and the OSPF configuration done under the interface.

```
interface vlan.0.100
ip ospf 1 area 0.0.0.1 instance 1
ip ospf 2 area 0.0.0.0 instance 2
exit
```

Redistribution and distance commands under router ospf apply to the base topology.

Redistribution and distance commands under the address-family command under router ospf apply to the multicast topology.

Configuring a Multicast Topology

To configure a topology, enter the router configuration mode of the router. Then configure an address-family (IPv4 or IPv6) and the topology itself that will apply to multicast traffic on this router.

Global Mode Topology Configuration

Configure a global topology instance for each topology on a router. Procedure 29-1 lists the multicast topology configuration commands for S-Series devices.

Procedure 29-1 Global Mode Topology Configuration

Step	Task	Command
1.	In router configuration mode, enter address family configuration mode for the address-family to which this topology will apply. To exit address family mode, use the **exit** command. To delete an address family, and the topology, use the **no** form of this command.	**address-family** {*ipv4* | *ipv6*} **multicast**

Procedure 29-1 Global Mode Topology Configuration (continued)

Step	Task	Command
2.	Configure a global topology instance. To remove this topology from the router configuration, use the **no** form of this command.	**topology** *topology-name*

Multicast Topology Configuration

Configure a topology for each router-address family combination ina VRF. You can apply protocols separately to each topology.

Table 29-1 lists the [BGP/OSPF/OSPFv3 protocol] multicast topology configuration commands for S-Series devices.

Table 29-1 Multicast Topology Configuration Table

Task	Command
In router config-*protocol* mode, enter the address family configuration mode for the address-family to which this topology will apply. Also enter the **multicast** or **unicast** sub-address family for this topology. To exit address family mode, use the **exit** command. To delete an address family, and the topology, use the **no** form of this command.	**address-family** {*ipv4* | *ipv6*} [**multicast | unicast]**
Configure static routing in the multicast topology in router configuration mode. To remove a static route from the router configuration, use the **no** form of this command.	**[ip | ipv6] mroute**

The following configuration example creates an IPv4 multicast topology named v4multi:

```
address-family ipv4 multicast
topology v4multi
exit
```

Create an instance of OSPF and join to the multicast topology:

```
router ospf 1
address-family ipv4 multicast
exit
exit
```

With this configuration, OSPF instance 1 is joined with both the "unicast" or base topology and the multicast topology v4multi, meaning that OSPF will contribute routes to both route tables. In order to have OSPF instance 1 joined only with the multicast topology, it is necessary to enter the "no address-family ipv4 unicast" command in OSPF configuration mode:

```
router ospf 1
router-id 192.168.200.1
address-family ipv4 multicast
exit
log-adjacency
```

```
no address-family ipv4 unicast
exit
```

Multi-Topology Display Commands

Table 29-2 lists Multi-Topology show commands for S-Series devices. The <*topology-name*> parameter value is the configured address family topology name.

Table 29-2 Multi-Topology Show Commands

Task	Command
Display topology information about an IPv4 BGP router.	**show ip bgp** multicast
Display topology information about an IPv6 BGP router.	**show ipv6 bgp** multicast
Display topology information about an IP OSPF router.	**show ip ospf interface topology** <*topology-name*>
Display topology information about an IPv6 OSPF router.	**show ipv6 ospf interface topology** <*topology-name*>
Display topology information about an IPv4 router.	**show ip route topology** <*topology-name*>
Display topology information about an IPv6 router.	**show ipv6 route topology** <*topology-name*>

For more information on Multi-Topology CLI commands, refer to your device's *CLI Reference Guide*.

30

Multicast Listener Discovery (MLD) Configuration

Using MLD in Your Network

Multicast Listener Discovery (MLD) enables each IPv6 router to discover the presence of nodes wishing to receive multicast packets on its directly attached links, and determines which multicast addresses are of interest to those neighboring nodes. MLD provides the discovered information to the routing protocol used by the router, in order to ensure that multicast packets are delivered to all links where there are interested receivers.

Multicast is a "one source to many destinations" method of simultaneously sending information over a network using the most efficient delivery strategy over each link. Only the end stations that explicitly indicate a need to receive a given multicast stream will receive it.

Applications that take advantage of multicast include video conferencing, streaming video, corporate communications, distance learning, and distribution of software, stock quotes, and news. See "Multicast Configuration" on page 27-1 for a detailed multicast discussion.

The S-Series supports MLD version 1 (RFC 2710) and version 2 (RFC 3810). MLD defaults to version 2.

Unlike unicast and broadcast, multicast uses network infrastructure efficiently because only one copy of the source traffic is sent throughout the network, going only to interested receivers, minimizing the burden placed on the sender, network, and receiver. The routers in the network take care of replicating the packet, where necessary, to reach multiple receivers. If a router decides that there are no interested users downstream from itself, it prunes the stream back to the next router. Thus, unwanted streams are not sent to the pruned routers, saving bandwidth and preventing unwanted packets from being sent.

Implementing MLD

You can implement the MLD multicast protocol on Extreme Networks devices using simple CLI commands as described in this chapter and the Multicast Listener Discovery (MLD) Commands chapter of the *Extreme Networks S-Series CLI Reference*. A basic configuration process involves the following tasks:

1. Configuring the VLANs and IP interfaces on which you want to transmit multicast.

2. Enabling the multicast protocol(s) on configured interfaces.

For PIM for IPv6 to operate, the Multicast Listener Discovery (MLD) protocol must be enabled. You must also configure a unicast routing protocol, such as OSPFv3. For both DVMRP and PIM for IPv4 to operate, IGMP must be enabled. See "Configuring IGMP" on page 27-20 for IGMP configuration information.

Understanding MLD

Multicast allows a source to send a single copy of data using a single IP address from a well-defined range for an entire group of recipients (an MLD group). A source sends data to an MLD group by simply setting the destination IP address of the datagram to be the MLD group address. Sources do not need to register in any way before they can begin sending data to a group, and do not need to be members of the group themselves. Routers between the source and recipients use the group address to route the data, forwarding duplicate data packets only when the path to recipients diverges.

Hosts that wish to receive IPv6 data from the MLD group join the group by sending a message to a multicast router on a local interface, using MLD.

Multicast routers communicate among themselves using a multicast routing protocol, such as DVMRP of PIM-SM. These protocols calculate a multicast distribution tree of recipients to ensure that:

- Multicast traffic reaches all recipients that have joined the MLD group
- Multicast traffic does not reach networks that do not have any such recipients (unless the network is a transit network on the way to other recipients)
- The number of identical copies of the same data flowing over the same link is minimized.

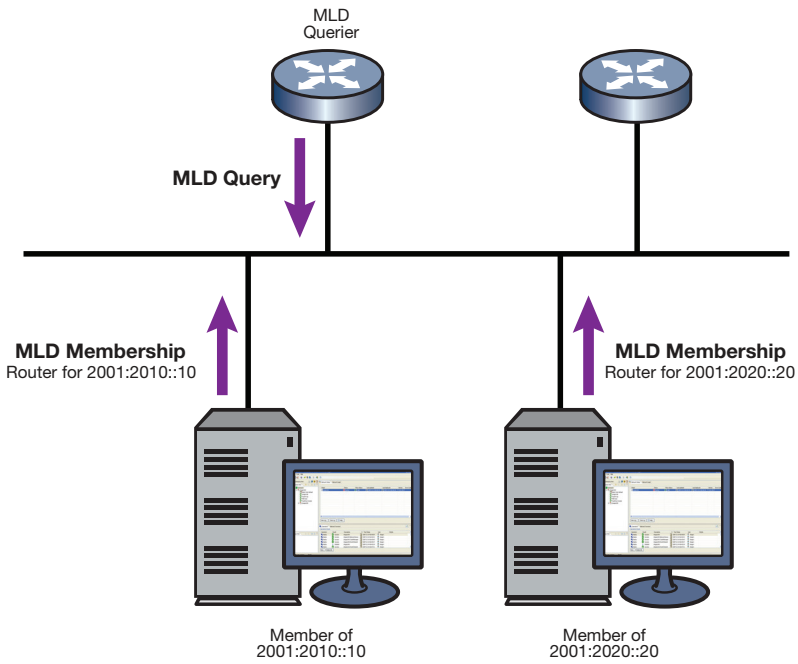
Group membership management is fundamental to the multicasting process. An arbitrary group of receivers can express interest in receiving a particular multicast stream, regardless of the physical or geographical boundaries of its members.

The purpose of IPv6 multicast group management is to optimize a switched network's performance so multicast packets will only be forwarded to those ports containing MLD group hosts or multicast switch devices instead of flooding to all ports in the subnet (VLAN).

MLD uses three key components to control multicast membership:

- **Source** — A server that sends an IPv6 multicast data stream with a particular MLD destination IPv6 and MAC address. A server may not have direct MLD involvement, as it often does not receive a multicast stream, but only sends a multicast stream.
- **Querier** — A device that periodically sends out queries in search of multicast hosts on a directly connected network. If multiple queriers are present on the LAN, the querier with the lowest IP address assumes the role.
- **Host** — A client end station that sends one of two MLD messages to a querier:
  - Join message — Indicates the host wants to receive transmissions associated to a particular multicast group.
  - Leave message — Indicates the host wants to stop receiving the multicast transmissions.

Figure 30-1 MLD Querier Determining Group Membership



As shown in Figure 30-1, an MLD-enabled device can periodically ask its hosts if they want to receive multicast traffic. If there is more than one device on the LAN performing IPv6 multicasting, one of these devices is elected querier and assumes the responsibility of querying the LAN for group members.

Based on the group membership information learned from MLD, a device can determine which (if any) multicast traffic needs to be forwarded to each of its ports. At Layer 3, multicast switch devices use this information, along with a multicast routing protocol, to support IPv6 multicasting across the Internet.

MLD provides the final step in IPv6 multicast delivery. It is only concerned with forwarding multicast traffic from the local switch device to group members on a directly attached subnetwork or LAN segment.

MLD neither alters nor routes any IPv6 multicast packets. Since MLD is not concerned with the delivery of IPv6 multicast packets across subnetworks, an external IPv6 multicast device is needed if IPv6 multicast packets have to be routed across different subnetworks.

MLD Support on Extreme Networks Devices

Extreme Networks devices implement MLD versions 1 and 2, which includes interoperability with version 1 hosts.

Depending on your Extreme Networks device, MLD can be configured independently at the switch level (Layer 2) and at the router level (Layer 3).

Extreme Networks devices support MLD as follows:

- Passively snooping on the MLD query and MLD report packets transferred between IPv6 multicast switches and IPv6 multicast host groups to learn MLD group members. Each Layer 2 device records which ports MLD packets are received on, depending on the kind of MLD message, so multicast data traffic is not flooded across every port on the VLAN when it is received by the switch.

MLD snooping is disabled by default on Extreme Networks devices. You can enable it using the **set mld enable** command on Extreme Networks S-Series devices as described in "Configuring MLD" on page 30-5.

- MLD actively sends MLD query messages to learn locations of multicast switches and member hosts in MLD groups within each VLAN.

Example: Sending a Multicast Stream

Figure 30-2 Sending a Multicast Stream with No Directly Attached Hosts

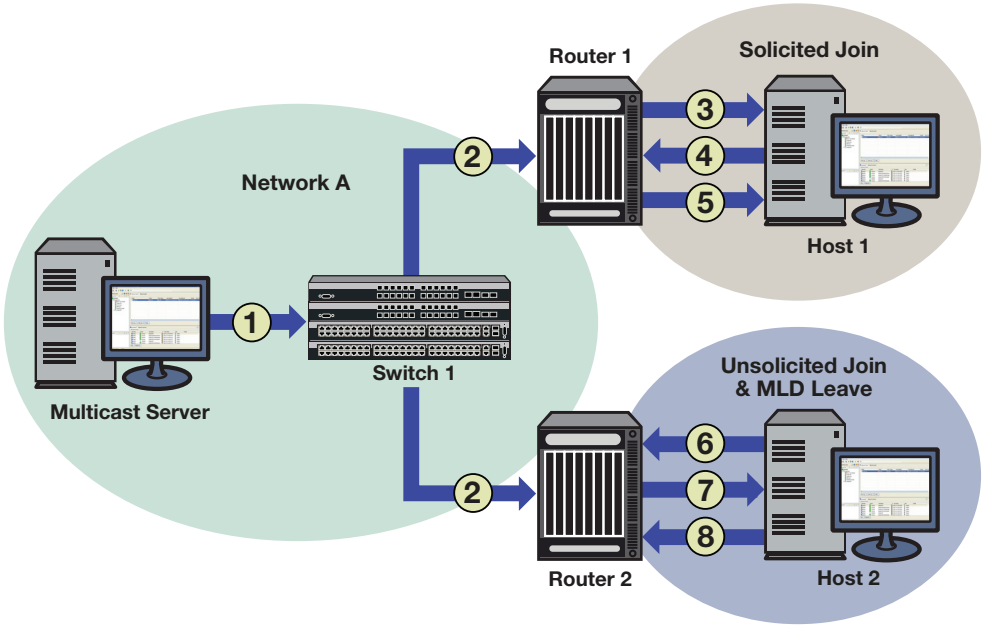


Figure 30-2 provides an example of MLD processing on Extreme Networks devices when there are no directly attached hosts.

1. A single IP multicast server, with no directly attached hosts, sends a multicast stream into the network via Switch 1.
2. Because MLD snooping is disabled, Switch 1 floods the multicast stream to all ports which are linked to Router 1 and Router 2.

   Each router performs an MLD forwarding check to see if there are any hosts that want to join the multicast group on its locally attached network. Each router drops multicast packets until a host joins the group using one of the following messages:

   – **solicited join** (sent in response to an MLD query produced by the router's interface)

     In Figure 30-2, this type of exchange occurs between Router 1 and Host 1 when:

     (3) Router 1 sends a query to potential Host 1.

     (4) Host 1 responds with a join message.

     (5) Router 1 forwards the multicast stream.

   – **unsolicited join** (sent as a request without receiving an MLD query first)

     In Figure 30-2, this type of exchange occurs between Router 2 and Host 2 when:

     (6) Host 2 sends a join message to Router 2.

(7) Router 2 forwards the multicast stream to Host 2.

(8) When it no longer wants to receive the stream, Host 2 can do one of the following:

-Send a leave message to Router 2.

-Time out the MLD entry by not responding to further queries from Router 2.

Configuring MLD

MLD is configured in switch mode on Extreme Networks S-Series devices. At Layer 2, MLD can be enabled for VLANs, regardless of whether it is enabled on routed interfaces. If, however, MLD is enabled on a routed interface, and the routed interface is a routed VLAN, then MLD must also be enabled at the switch level.

MLD Configuration Commands

Table 30-1 lists the MLD configuration commands for Extreme Networks S-Series devices.

Table 30-1 MLD Configuration Commands

Task	Command
Enable MLD on one or more VLANs.	**set mld enable** *vlan-list*
Disable MLD on one or more VLANs.	**set mld disable** *vlan-list*
Enable MLD querying on one or more VLANs.	**set mld query-enable** *vlan-list*
Disable MLD querying on one or more VLANs.	**set mld query-disable** *vlan-list*
Determine what action to take with multicast frames when the MLD flow table is full.	**set mld flow-full-action** *action*
Configure MLD settings on one or more VLANs.	**set mld config** *vlan-list* {[**query-interva**l *query-interval*] [**mld-version** *mldversion*] [**max-resp-time** *max-resp-time*] [**robustness** *robustness*] [**last-mem-int** *last-mem-int*] [**fast-leave** *fast-leave*] [**filter-id** filter-id] [**filter-status** {enable | disable}]}
Remove MLD configuration settings for one or more VLANs.	**set mld delete** *vlan-list*
Create a new static MLD entry or add one or more new ports to an existing entry.	**set mld static** *group vlan-list* [**modify** {[**include-ports**] [**exclude-ports**]}]
Remove a static MLD entry or remove one or more ports from an existing entry.	**clear mld static** *group vlan-list* [**modify** {[**include-ports**] [**exclude-ports**]}]
Change the MLD classification of received IP frames.	**set mld protocols classification** *classification* **protocol-id** *protocol-id* [**modify**]
Clear the binding of IP protocol ID to MLD classification.	**clear mld protocols protocol-id** *protocol-id*
Enable port fast leave on the specified port or range of ports.	**set mld portFastLeave** *port-list*
Disable port fast leave on the specified port or range of ports.	**clear mld portFastLeave** *port-list*
Create an input filter to apply to the VLAN.	**set mld input-filter** *filter-id rule-id* **start-ip** *ip-address* **end-ip** *ip-address* **protocol-action** {deny | allow} **flow-action** {drop | flood | allow}

Table 30-1 MLD Configuration Commands (continued)

Task	Command
Clear an input filter.	**clear mld input-filter** *filter-id* [*rule-id*]
Set the action taken when the first few frames of a multicast stream are received (that is, before the stream is added to the MLD database).	**set mld unknown-input-action** {routers | flood | discard}

Basic MLD Configurations

Procedure 30-1 describes the basic steps to configure MLD on Extreme Networks S-Series devices. This procedure assumes that the VLANs on which MLD will run have been configured and enabled with IP interfaces.

Procedure 30-1 Basic MLD Configuration

Step	Task	Command
1.	In switch mode, configure MLD for each VLAN interface.	**set mld config** *vlan-list* {[**query-interval** *query-interval*] [**mld-version** *mldversion*] [**max-resp-time** *max-resp-time*] [**robustness** *robustness*] [**last-mem-int** *last-mem-int*]}
2.	In switch mode, enable MLD on each VLAN interface.	**set mld enable** *vlan-list*
3.	In switch mode, enable MLD querying on each of the VLANs specified in step 2.	**set mld query-enable** *vlan-list*

For more information on MLD CLI commands, refer to your device's *CLI Reference Guide*.

Example MLD Configuration

The following example enables MLD and MLD query on VLANs 2 and 3:

```
S Chassis->set mld enable 2,3
S Chassis->set mld query-enable 2,3
```

MLD Display Commands

Table 30-2 lists MLD show commands.

Table 30-2 MLD Show Commands

Task	Command
Display the status of MLD on one or more VLANs.	**show mld enable** *vlan-list*
Display the MLD query status of one or more VLANs.	**show mld query** *vlan-list*
Display the action to be taken with multicast frames when the multicast MLD flow table is full.	**show mld flow-full-action**
Display MLD configuration information for one or more VLANs.	**show mld config** *vlan-list*
Display MLD information regarding multicast group membership.	**show mld groups** [**group** *group*] [**vlan-list** *vlan-list*] [**sip** *sip*] [**-verbose**]

Table 30-2 MLD Show Commands (continued)

Task	Command
Display static MLD ports for one or more VLANs or MLD groups.	**show mld static** [**group** *group*] [**vlan-list** *vlan-list*]
Display the binding of IP protocol ID to MLD classification.	**show mld protocols**
Display MLD information for a specific VLAN.	**show mld vlan** [*vlan-list*]
Display MLD reporter information.	**show mld reporters** [**portlist** *portlist*] [**group** *group*] [**vlan-list** *vlan-list*] [**sip** *sip*]
Display MLD flow information.	**show mld flows** [**portlist** *portlist*] [**group** *group*] [**vlan-list** *vlan-list*] [**sip** *sip*]
Display MLD counter information.	**show mld counters**
Display the number of MLD flows set on the Extreme Networks S-Series device.	**show mld number-flows**
Display configuration information for input filters.	**show mld input-filter** [*filter-id]* [*rule-id*]
Display the action taken when the first frames of a multicast stream are received.	**show mld unknown-input-action**

31

System Logging Configuration

This chapter provides the following information about configuring and monitoring Syslog on Extreme Networks S-Series devices.

Using Syslog in Your Network

Syslog, short for System Logging, is a standard for forwarding log messages in an IP network that is typically used for network system management and security auditing. The term often applies to both the actual Syslog protocol, as well as the application sending Syslog messages.

As defined in RFC 3164, the Syslog protocol is a client/server-type protocol which enables a station or device to generate and send a small textual message (less than 1024 bytes) to a remote receiver called the Syslog server. Messages are transmitted using User Datagram Protocol (UDP) packets and are received on UDP port 514. These messages inform about simple changes in operational status or warn of more severe issues that may affect system operations.

When managed properly, logs are the eyes and ears of your network. They capture events and show you when problems arise, giving you information you need to make critical decisions whether you are building a policy rule set, fine tuning an Intrusion Detection System, or validating which ports should be open on a server. However, since it's practically impossible to wade through the volumes of log data produced by all your servers and network devices, Syslog's ability to place all events into a single format so they can be analyzed and correlated makes it a vital management tool. Because it is supported by a wide variety of devices and receivers across multiple platforms, you can use it to integrate log data from many different types of systems into a central repository.

Efficient Syslog monitoring and analysis reduces system downtime, increases network performance, and helps tighten security policies. It can help you:

- Troubleshoot switches, firewalls and other devices during installation and problem situations.
- Perform intrusion detection.
- Track user activity.

Syslog On S-Series Switches

By default, Syslog is operational on S-Series devices at startup. All generated messages are eligible for logging to local destinations and to remote servers configured as Syslog servers. Using simple CLI commands, you can adjust device defaults to configure the following:

- Message sources—which system applications on which modules should log messages
- Message destinations—will messages be sent to the local console, the local file system, or to remote Syslog servers? Which facility (functional process) will be allowed to send to each destination?

The following section provides an overview of Syslog features and functions supported on Extreme Networks devices and their default configurations. Later sections will provide instructions on changing default settings to suit your network logging needs.

Syslog Overview

Developers of various operating systems, processes, and applications determine the circumstances that will generate system messages and write those specifications into their programs. Messages can be generated to give status, either at a certain period of time, or at some other interval, such as the invocation or exit of a program. Messages can also be generated due to a set of conditions being met. Typically, developers quantify these messages into one of several broad categories, generally consisting of the facility that generated them, along with an indication of the severity of the message. This allows system administrators to selectively filter the messages and be presented with the more important and time sensitive notifications quickly, while also having the ability to place status or informative messages in a file for later review.

Switches must be configured with rules for displaying and/or forwarding event messages generated by their applications. In addition, Syslog servers need to be configured with appropriate rules to collect messages so they can be stored for future reference. This document will describe how to complete these key configuration steps on S-Series platforms.

If C2 security mode is enabled, while in Read-Write user mode you can not:

- Create, modify, or clear a server logging configuration
- Create, modify, or clear a default logging configuration
- Create, modify, or clear a logging application configuration

Configuring Syslog Message Disposition

The Syslog implementation on S-Series devices uses a series of system logging messages to track device activity and status. These messages inform users about simple changes in operational status or warn of more severe issues that may affect system operations. Logging can be configured to display messages at a variety of different severity levels about application-related error conditions occurring on the device.

You can decide to have all messages stored locally, as well as to have all messages of a high severity forwarded to another device. You can also have messages from a particular facility sent to some or all of the users of the device, and displayed on the system console. For example, you may want all messages that are generated by the mail facility to be forwarded to one particular Syslog server. However you decide to configure the disposition of the event messages, the process of having them sent to a Syslog collector generally consists of:

- Determining which messages at which severity levels will be forwarded.
- Defining one or more remote receivers (Syslog servers/console displays).

Filtering by Severity and Facility

Syslog daemons determine message priority by filtering them based on a combined facility and severity code. Severity indicates the seriousness of the error condition generating the Syslog message. This is a value from 1 to 8, with 1 indicating highest severity. Facility categorizes which functional process is generating an error message. The Extreme Networks implementation uses the eight facility designations reserved for local use: **local0 – local7** defined in RFC 3164. You can modify these default facility and severity values to control message receipt and aid in message sorting on target servers.

For example, you can configure all router messages to go to Server 1 using facility local1, while all SNMP messages go to Server 1 using facility local2.

The following sections provide greater detail on modifying key Syslog components to suit your enterprise.

Syslog Components and Their Use

Table 31-1 describes the Extreme Networks implementation of key Syslog components.

Table 31-1 Syslog Terms and Definitions

Term	Definition	Extreme Networks Usage
Facility	Categorizes which functional process is generating an error message. Syslog combines this value and the severity value to determine message priority.	Extreme Networks uses the eight facility designations reserved for local use: **local0** – **local7**. Default is **local4**, which allows the message severity portion of the priority code to be visible in clear text, making message interpretation easiest. For more information about facility designations, refer to RFC 3164.
Severity	Indicates the severity of the error condition generating the Syslog message. The lower the number value, the higher will be the severity of the condition generating the message.	Extreme Networks devices provide the following eight levels: **1** - emergencies (system is unusable) **2** - alerts (immediate action required) **3** - critical conditions **4** - error conditions **5** - warning conditions **6** - notifications (significant conditions) **7** - informational messages **8** - debugging messages The default Syslog configuration allows applications (log message sources) to forward messages at a severity level of 6, and destinations (console, file system, or remote Syslog servers) to log messages at a severity level of 8.

Note: Numerical values used in Extreme Networks syslog CLI and the feature's configuration MIB range from 1-8. These map to the RFC 3164 levels of 0-7 respectively. Syslog messages generated report the RFC 3164 specified level values.

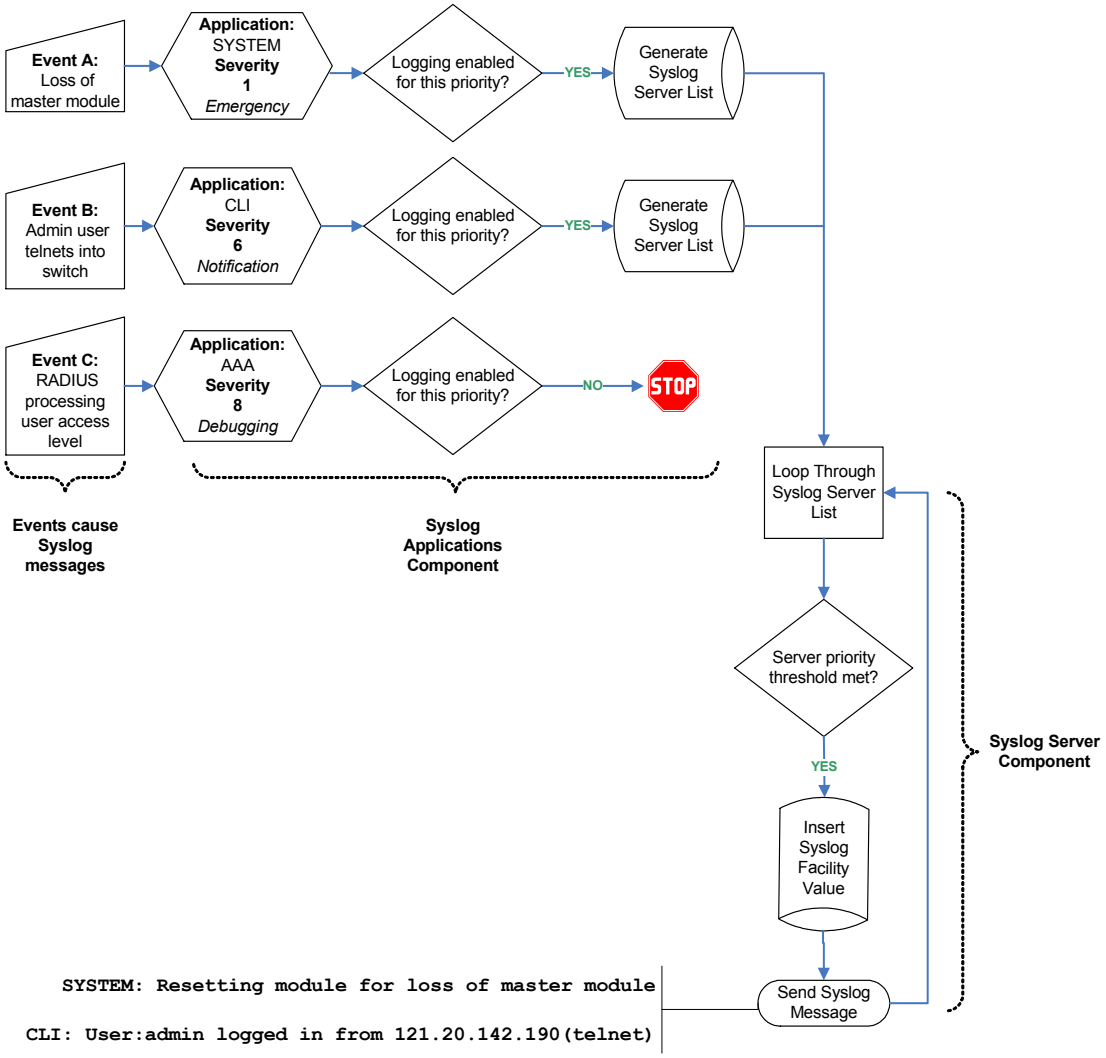
Table 31-1 Syslog Terms and Definitions (continued)

Term	Definition	Extreme Networks Usage
Application	Client software applications running on devices that can generate Syslog messages.	Extreme Networks supported applications and their associated CLI mnemonic values include: **CLI** - Command Line Interface **SNMP** - Simple Network Management Protocol **Webview** - Extreme Networks Web-based system management **System** - System messages **RtrFe** - Router Forwarding Engine **Trace** - Trace logging **RtrLSNat** - Load Share Network Address Translation **FlowLimt** - Flow limiting **UPN** - User Personalized Networks **AAA** - Authentication, Authorization and Accounting Use the **show logging application all** command to list supported applications and the corresponding CLI numeric or mnemonic values you can use to configure application logging on your devices.
Syslog server	A remote server configured to collect and store Syslog messages.	Extreme Networks devices allow up to 8 server IP addresses to be configured as destinations for Syslog messages. By default, Syslog server is globally enabled, with no IP addresses configured, at a severity level of 8.

Basic Syslog Scenario

Figure 31-1 on page 31-5 shows a basic scenario of how Syslog components operate on an Extreme Networks switch. By default, all applications running on the Extreme Networks switch are allowed to forward Syslog messages generated at severity levels 6 through 1. In the configuration shown, these default settings have not been changed.

Figure 31-1 Basic Syslog Scenario



Default application settings in the example in Figure 31-1 have not been modified. Therefore, an emergency message triggered by a system reset due to loss of the master module is forwarded to Syslog destinations. The CLI-related message notifying that a user has logged in remotely is also forwarded. Configured Syslog server(s) will receive all forwarded messages since their default severity threshold is at 8 (accepting messages at all severity levels).

Any messages generated by applications at severity levels 7 and 8 are not forwarded in this example. For instance, forwarding does not occur for an AAA authentication-related debugging message with information about RADIUS access level processing for a particular user. If at some point in time it becomes necessary, for example, to log all AAA authentication-related message activity and to save it to a file so authentication details can be tracked, the administrator can allow that specific application to forward debugging messages to a Syslog server, as well as to the console and persistent file storage.

For more information on how to configure these basic settings, refer to "Syslog Command Precedence" on page 31-7, and the "Configuration Examples" on page 31-12.

Interpreting Messages

Every system message generated by the S-Series platforms follows the same basic format:

```
<facility/severity> time stamp address application [slot] message text
```

Example

This example shows Syslog informational messages, displayed with the **show logging buffer** command. It indicates that messages were generated by facility code 16 (local4) at severity level 5 from the CLI application on IP address 10.42.71.13.

```
S Chassis(rw)->show logging buffer
<165>Sep  4 07:43:09 10.42.71.13 CLI[5]User:rw logged in from 10.2.1.122 (telnet)
<165>Sep  4 07:43:24 10.42.71.13 CLI[5]User: debug failed login from 10.4.1.100
(telnet)
```

Table 31-2 describes the components of these messages.

Table 31-2 Syslog Message Components

Component	Description	Example Code
Facility/Severity	Combined code indicating the facility generating the message and the severity level used to determine message priority. Facility codes 16 - 23 are Syslog designations for local0 - local7, the Extreme Networks supported designations for local use. For a complete list of facility codes, refer to RFC 3164.	<165> = Numerical code indicating a message from facility local4 at severity 5.
Time stamp	Month, date, and time the Syslog message appeared.	Sep 4 07:43:09
Address	IP address of the client originating the Syslog message.	10.42.71.13
Application	Client process generating the Syslog message.	CLI
Slot/Module	Slot location of the device module generating the Syslog message.	(5) = Slot 5 in the chassis.
Message text	Brief description of error condition.	User: debug failed login from 10.4.1.100 (telnet)

Configuring Syslog

Syslog Command Precedence

Table 31-3 lists basic Syslog commands and their order of precedence on Extreme Networks switches.

Table 31-3 Syslog Command Precedence

Syslog Component	Command	Function
Logging defaults	**set logging default** { [**facility** *facility*] [**severity** *severity*] [**port** *port*] }	Sets default parameters for facility code, severity level and/or UDP port for all Syslog servers and local destinations. Settings will be applied when Syslog servers are configured without specifying values with the **set logging server** command. This command overrides factory defaults.
Server settings	**set logging server** *index* **ip-addr** *ip-addr* [**facility** *facility*] [**severity** *severity*] [**descr** *descr*] [**port** *port*] **state enable** | **disable**	During or after new server setup, specifies a server index, IP address, and operational state for a Syslog server. Optionally, this command specifies a facility code, severity level at which messages will be accepted, text string description, and/or UDP port for the specified server. This command overrides system defaults for the specified server. If not specified with this or the **set logging default** command, optional server parameters will be set to the system defaults listed in Table 31-4 on page 8.
Application settings	**set logging application** {[*mnemonic*|**all**]} [**level** *level*] [**servers** *servers*]	Sets the severity level at which one or all applications will send messages to Syslog servers. If not specified, settings will apply to all configured servers and severity level will not be changed from system defaults.

About Server and Application Severity Levels

By default, client applications will forward Syslog messages at severity levels 6 through 1, and servers will log messages at all severity levels (8 through 1). You can use the procedures described in this chapter to change these parameters, fine tuning the scope of message logging and modifying the Syslog behavior between one or more client applications and one or more servers.

Configuring Syslog Server(s)

Use the following commands to configure one or more servers as destinations for Syslog messages and verify the configuration:

1. Add a Syslog server to the device's server list:

   set logging server *index* **ip-addr** *ip-addr* **state enable**

   Index is a value from 1 to 8 that specifies the server table index number for this server.

2. (Optional) Verify the server configuration:

   show logging server [*index*]

   If *index* is not specified, information for all configured Syslog servers will be displayed.

Example

This sample output from the **show logging server** command shows that two servers have been added to the device's Syslog server list. These servers are using the default UDP port 514 to receive messages from clients and are configured to log messages from the local1 and local2 facilities, respectively. Logging severity on both servers is set at 5 (accepting messages at severity levels 5 through 1). Using the commands described in the next section, these settings can be changed on a per-server basis, or for all servers.

```
S Chassis(rw)->show logging server

   IP Address      Facility   Severity     Description    Port     Status
-------------------------------------------------------------------------
 1 132.140.82.111 local1     warning(5)    default         514    enabled
 2 132.140.90.84  local2     warning(5)    default         514    enabled
```

Modifying Syslog Server Defaults

Unless otherwise specified, the switch will use the default server settings listed in Table 31-4 for its configured Syslog servers:

Table 31-4 Syslog Server Default Settings

Parameter	Default Setting
facility	local4
severity	8 (accepting all levels)
descr	no description applied
port	UDP port 514

Use the following commands to change these settings either during or after enabling a new server.

Displaying System Logging Defaults

To display system logging defaults, or all logging information, including defaults:

show logging {**default**|**all**}

Modifying Default Settings

You can change factory default logging settings using one of the following methods.

- To specify logging parameters during or after new server setup:

  set logging server *index* **ip-addr** *ip-addr* [**facility** *facility*] [**severity** *severity*] [**descr** *descr*] [**port** *port*] **state enable**

  If not specified, optional server parameters will be set to the system defaults listed in Table 31-4. Refer back to Filtering by Severity and Facility and to Table 31-1 for more information. on how these parameters operate.

- To change default parameters for all servers:

  set logging default {[**facility** *facility*] [**severity** *severity*] [**port** *port*]}

Examples

This example shows how to configure the switch to forward messages from facility category local6 at severity levels 3, 2, and 1 to Syslog server 1 at IP address 134.141.89.113:

```
S Chassis(rw)->set logging server 1 ip-addr 134.141.89.113 facility local6
severity 3
```

This example shows how to change Syslog defaults so that messages from the local2 facility category at a severity level of 4 will be forwarded to all servers. These settings will apply to all newly-configured servers, unless explicitly configured with the **set logging server** command:

```
S Chassis(rw)->set logging default facility local2 severity 4
```

Reviewing and Configuring Logging for Applications

By default, all applications running on S-Series devices are allowed to forward messages at severity levels 6 through 1 to all configured destinations (Syslog servers, the console, or the file system).

Displaying Current Application Severity Levels

To display logging severity levels for one or all applications currently running on your device:

show logging application {*mnemonic*|**all**}

Example

This example shows output from the **show logging application all** command. A numeric and mnemonic value for each application is listed with the severity level at which logging has been configured and the server(s) to which messages will be sent. In this case, logging for applications has not been changed from the default severity level of 6. This means that notifications and messages with severity values 6 through 1 will be sent to configured servers.

```
S Chassis(rw)->Show logging application all
Application          Current Severity Level     Server List
-------------------------------------------------------------
 88     RtrAcl                   6              1-8
 89     CLI                      6              1-8
 90     SNMP                     6              1-8
 91     Webview                  6              1-8
 93     System                   6              1-8
 95     RtrFe                    6              1-8
 96     Trace                    6              1-8
 105    RtrLSNat                 6              1-8
 111    FlowLimt                 6              1-8
 112    UPN                      6              1-8
 117    AAA                      6              1-8
 118    Router                   6              1-8
 140    AddrNtfy                 6              1-8
 141    OSPF                     6              1-8
 142    VRRP                     6              1-8
 145    RtrArpProc               6              1-8
 147    LACP                     6              1-8
```

```
 148    RtrNat                    6              1-8
 151    RtrTwcb                   6              1-8
 158    HostDoS                   6              1-8

1(emergencies)  2(alerts)       3(critical)
4(errors)       5(warnings)     6(notifications)
7(information)  8(debugging)
```

Note: Mnemonic values are case sensitive and must be typed as they are listed in the **show logging application** command display for your device. Refer to Table 31-1 for sample CLI mnemonic values.

Modifying Severity Levels and Assigning Syslog Servers for Applications

Applications running on Extreme Networks devices will use the default Syslog settings unless otherwise configured by the **set logging server** or **set logging default** commands as previously described.

To modify the severity level at which log messages will be forwarded and the server(s) to which they will be sent for one or all applications:

set logging application {[*mnemonic*|**all**]} [**level** *level*] [**servers** *servers*]

Example

This example shows how to set the severity level for SSH (Secure Shell) to 5 so that warning conditions and messages of greater severity (levels 5 to 1) generated by that application will be sent to Syslog server 1.

```
S Chassis(rw)->set logging application SSH level 5 server 1
```

Enabling Console Logging and File Storage

S-Series devices allow you to display logging messages to the console and save to a persistent file. In addition, S-Series devices also provide the option of allowing you to display messages to the current console CLI session only.

Console logging allows you to view only as many messages as will fit on the screen. As new messages appear, old messages simply scroll off the console. While this is a temporary means of logging information, it allows you to track very specific activities quickly and easily. Console log messages can also be saved to a persistent file at two locations:

- slotX/logs/current.log — Location of current system log messages (up to 256k), where X specifies the slot location of the device.
- slotX/logs/old.log — Location of previous system log messages, where X specifies the slot location of the device. Current messages will be moved to the old.log when current.log file exceeds 256k.

Use the following commands to review and configure console logging and file storage.

Displaying to the Console and Saving to a File

To display log messages to the console and save to a persistent file:

set logging local console {**enable** | **disable**} **file** {**enable** | **disable**} **sfile** {**enable** | **disable**}

Note: The **set logging local** command requires that you specify both console and file settings. For example, **set logging local console enable** would not execute without also specifying **file enable** or **disable** or **sfile enable** or **disable** for a secure file.

Displaying to the Current CLI Session

To display logging to the current CLI console session on a S-Series device:

```
set logging here enable
```

This adds the current CLI session to the list of Syslog destinations, and will be temporary if the current CLI session is using Telnet or SSH.

Displaying a Log File

To display the contents of the persistent log file:

```
show file slotslotnumber/logs/current.log|old.log
```

Note: These log files may also be copied to another device using FTP or TFTP.

CLI and SNMP Audit Logging

CLI and SNMP audit logging provide Syslog messages for:

- CLI configuration changes
- SNMP configuration changes
- SNMP authentication failures

Note: In Release 7.41, CLI and SNMP audit logging messages were moved from other Syslog applications to the security application.

CLI and SNMP audit logging is turned on by configuring logging for the security application at level 7 (information), using the **set logging application security level 7** command. CLI audit logging messages display the user mode, the IP address of the station originating the CLI command and the type of client connection, the CLI command entered and a status indicating command success (OK) or failure (Fail). The logging output when entering this command would log a message similar to:

```
<166>May  3 16:44:12 10.21.130.166 Security[1]User:admin;
Source:134.141.90.54(ssh); Action:"set logging application security level 7 ";
Status:OK
```

SNMP audit logging messages display the SNMP user, the IP address of the station originating the SET, the MIB leaf modified and the setting value, and a status indicating success (OK) or failure (Fail).

SNMP authentication failure messages display the SNMP user, the IP address of the station originating the request, and the failure reason.

Configuration Examples

Enabling a Server and Console Logging

Procedure 31-1 shows how you would complete a basic Syslog configuration. In this example, the default application severity level has not been modified, allowing all applications to forward messages to configured destinations. One Syslog server is configured on IP address 10.1.1.2, logging all messages. Console logging is enabled, but persistent file storage is not.

Procedure 31-1 Configuring a Server and Console Logging

Step	Task	Command(s)
1.	Configure Syslog server 1 and accept default settings (listed in Table 31-4 on page 31-8).	**set logging server 1 ip-addr 10.1.1.2 state enable**
2.	(Optional) Verify that application logging settings are at default values for the enabled server.	**show logging application all**
3.	Enable console logging and disable file storage.	**set logging local console enable file disable**

Note: The **set logging local** command requires that you specify both console and file settings. For example, **set logging local console enable** would not execute without also specifying **file enable** or **disable**.

Adjusting Settings to Allow for Logging at the Debug Level

Procedure 31-2 shows how you would adjust the previous Syslog configuration so that all AAA-related authentication messages (level 8) could be forwarded to Server 2 at IP address 10.1.1.3, displayed on the console and saved to persistent file storage. This would enable all Syslog messaging capabilities for this particular application. Since the severity for this new server has not changed from the default of level 8, there is no need to adjust this setting.

Procedure 31-2 Adjusting Settings for an Application

Step	Task	Command(s)
1.	Configure Syslog server 2 and accept default settings (listed in Table 31-4 on page 31-8).	**set logging server 2 ip-addr 10.1.1.3 state enable**
2.	Set the severity level for the AAA application to level 8.	**set logging application** AAA **level** 8 **servers** 2
3.	Enable console logging and file storage.	**set logging local console enable file enable**

32

Network Monitoring Configuration

This document describes the network monitoring features and their configuration on Extreme Networks S-Series devices.

Using Network Monitoring in Your Network

S-Series network monitoring features support for:

- Console/Telnet based monitoring:
  - Display contents and determine the size of the command history buffer
  - Close an active console port or Telnet session
- Network Diagnostics:
  - Determine the availability of another node on the network (ping)
  - Display a hop-by-hop path through an IP network (traceroute)
  - Query name servers (nslookup)
- Display of switch network connections:
  - Display statistics for the active connections on the switch
  - Display switch users
  - Send a message to switch user
- SMON statistics:
  - Monitor SMON priority and VLAN statistics counting
- RMON:
  - Record statistics measured by the RMON probe for each monitored interface on the device.
  - Record periodic statistical samples from a network.
  - Periodically gather statistical samples from variables in the probe and compares them with previously configured thresholds.
  - Record statistics associated with each host discovered on the network.

- Control the generation and notification of events from the device.
- Generate tables that describe hosts that top a list ordered by one of their statistics.
- Record statistics for conversations between two IP addresses.
- Allow packets to be matched by a filter equation.
- Allow packets to be captured upon a filter match.

Network Monitoring Overview

This section provides an overview of network monitoring configuration.

Console/Telnet History Buffer

The history buffer lets you recall your previous CLI input. The size of the history buffer determines how many lines of previous CLI input are available for recall. By default, the size of this buffer is 20 lines. The configured size can be displayed. The contents of the buffer can be displayed.

Use the **set history** command in any command mode to set the size of the history buffer to a value between **1** - **100** lines.

```
S Chassis(rw)->set history 25
```

Use the **show history** command in any command mode to display the currently configured size of the history buffer.

```
S Chassis(rw)->show history
History size currently set to: 25
S Chassis(rw)->
```

Use the **history** command in any command mode to display the contents of the history buffer.

```
S Chassis(rw)->history
 1 history
 2 show gvrp
 3 show vlan
 4 show igmp
S Chassis(rw)->
```

Network Diagnostics

S-Series network diagnostics provide for:

- Pinging another node on the network to determine its availability
- Performing a traceroute through the IP network to display a hop-by-hop path from the device to a specific destination host
- Querying name servers to translate hostnames to IP addresses or IP addresses to hostnames

Use the **ping** command, in any command mode, to determine whether the specified node is available.

```
S Chassis(rw)->ping -c 10 127.0.0.1
PING 127.0.0.1 (localhost) 64 bytes of data.
64 bytes from 127.0.0.1 (localhost): icmp_seq=0 ttl=64 time=1.58 ms
```

```
64 bytes from 127.0.0.1 (localhost): icmp_seq=1 ttl=64 time=1.52 ms
64 bytes from 127.0.0.1 (localhost): icmp_seq=2 ttl=64 time=1.57 ms
64 bytes from 127.0.0.1 (localhost): icmp_seq=3 ttl=64 time=2.26 ms
64 bytes from 127.0.0.1 (localhost): icmp_seq=4 ttl=64 time=1.42 ms
```

Use the **traceroute** command, in any command mode, to display a hop-by-hop path through an IP network from the device to a specific destination host.

```
S Chassis(rw)->traceroute 192.167.252.17
traceroute to 192.167.252.17 (192.167.252.17), 30 hops max, 40 byte packets
  1  matrix.extremenetworks.com (192.167.201.40)  20.000 ms  20.000 ms  20.000 ms
  2  14.1.0.45 (14.1.0.45)  40.000 ms  10.000 ms  20.000 ms
  3  192.167.252.17 (192.167.252.17)  50.000 ms  0.000 ms  20.000 ms
```

Use the **nslookup** command, in any command mode, to query name servers, translating hostnames to IP addresses or IP addresses to hostnames.

```
S Chassis(su)->nslookup -x 127.0.0.1
Name: localhost
Address: 127.0.0.1
```

Switch Connection Statistics

Switch connection statistics can be displayed for:

- ICMP
- IP
- TCP
- UDP

Use the **show netstat** command to display switch connection statistics. Use the **stats** option to display statistics for all supported protocols.

```
S Chassis(rw)->show netstat stats
Ip:
    26034 total packets received
    25824 with invalid addresses
    0 forwarded
    0 incoming packets discarded
    187 incoming packets delivered
    6391 requests sent out
    21 dropped because of missing route
Icmp:
    14 ICMP messages received
    0 input ICMP message failed
    ICMP input histogram:
        destination unreachable: 14
    6184 ICMP messages sent
    0 ICMP messages failed
    ICMP output histogram:
        destination unreachable: 1
```

```
        echo request: 6183
Tcp:
    2 active connection openings
    2 passive connection openings
    0 failed connection attempts
    0 connection resets received
    4 connections established
    153 segments received
    153 segments send out
    0 segments retransmitted
    0 bad segments received
    0 resets sent
Udp:
    42 packets received
    1 packets to unknown port received
    0 packet receive errors
    57 packets sent
S Chassis(rw)->
```

Users

The network monitoring feature supports the display of information about the active console port or Telnet session(s) logged in to the switch. It also provides for the ability to send a message to one or all users with active sessions.

Use the **show users** command to display information for active console port or Telnet sessions on the switch.

```
S Chassis(rw)->show users
  Session  User  Location
  -------- ----- -------------------------
* console  admin console (via com.1.1)
  telnet   rw    134.141.192.18
S Chassis(rw)->
```

Use the **tell** command to send a message to one or all users on the switch.

```
S Chassis(rw)->tell rw@134.141.192.18 "System reset in 15 minutes"
```

User rw@134.141.192.18 will receive:

```
Message from admin@console: "System reset in 15 minutes"
```

RMON

RMON (Remote Network Monitoring) is an industry standard specification that provides comprehensive network fault diagnosis, planning, and performance tuning information and allows for interoperability between SNMP management stations and monitoring agents. RMON extends the SNMP MIB capability by defining additional MIBs that generate a much richer set of data about network usage. These MIB "groups" each gather specific sets of data to meet common network monitoring requirements.

RMON statistics and history can be configured to gather packet counts in both the receive and transmit, receive only, or transmit only directions.

Table 32-1 lists:

- The nine RMON monitoring groups supported on S-Series devices
- Each group's function
- The elements it monitors
- The group's associated commands

Table 32-1 RMON Monitoring Group Functions and Commands

RMON Group	What It Does...	What It Monitors...	CLI Command(s)
Statistics	Records statistics measured by the RMON probe for each monitored interface on the device.	Packets dropped, packets sent, bytes sent (octets), broadcast and multicast packets, CRC errors, oversized and undersized packets, fragments, jabbers, and counters for packets.	**show rmon stats** **set rmon stats** **clear rmon stats**
History	Records periodic statistical samples from a network.	Sample period, number of samples and item(s) sampled.	**show rmon history** **set rmon history** **clear rmon history**
Alarm	Periodically gathers statistical samples from variables in the probe and compares them with previously configured thresholds. If the monitored variable crosses a threshold, an event is generated.	Alarm type, interval, starting threshold, stop threshold.	**show rmon alarm** **set rmon alarm properties** **set rmon alarm status** **clear rmon alarm**
Event	Controls the generation and notification of events from the device.	Event type, description, last time event was sent.	**show rmon event** **set rmon event properties** **set rmon event status** **clear rmon event**
Host	Records statistics associated with each host discovered on the network.	Host address, packets and bytes received and transmitted, and broadcast, multicast and error packets.	**show rmon host** **set rmon host properties** **set rmon host status** **clear rmon host**
Host TopN	Generates tables that describe hosts that top a list ordered by one of their statistics. These rate-based statistics are samples of one of their base statistics over an interval specified by the management station.	Statistics, top host(s), sample stop and start period, rate base, and duration.	**show rmon topN** **set rmon topN properties** **set rmon topN status** **clear rmon topN**

Table 32-1 RMON Monitoring Group Functions and Commands (continued)

RMON Group	What It Does...	What It Monitors...	CLI Command(s)
Matrix	Records statistics for conversations between two IP addresses. As the device detects a new conversation, it creates a new matrix entry.	Source and destination address pairs and packets, bytes and errors for each pair.	**show rmon matrix** **set rmon matrix properties** **set rmon matrix status** **clear rmon matrix**
Filter	Allows packets to be matched by a filter definition. These matched packets form a data stream or "channel" that may be captured or may generate events.	Packets matching the filter definition.	**show rmon channel** **set rmon channel** **clear rmon channel** **show rmon filter** **set rmon filter** **clear rmon filter**
Packet Capture	Allows packets to be captured upon a filter match.	Packets matching the filter definition.	**show rmon capture** **set rmon capture** **clear rmon capture**

SMON Priority and VLAN Statistics Counting

SMON is a set of RMON MIB extensions for switch monitoring. The S-Series supports the enabling and display of SMON Ethernet priority and VLAN statistics counters. SMON is described by RFC 2613. An SMON session for a specified port or range of ports is first created and then enabled before statistics are collected.

The S-Series platform supports a maximum of 16 SMON switch-wide VLAN sessions and 128 SMON priority sessions. Resources available to SMON priority and VLAN sessions are shared with other SMON tasks and port mirroring. Packets that match multiple SMON sessions may be counted in only one of them. Depending upon your configuration needs, you may not be able to configure the maximum number of supported SMON VLAN or priority sessions.

Use the **set smon priority create** and **set smon vlan create** commands to create priority and VLAN SMON sessions for the specified port(s) on the switch.

Use the **set smon priority enable** and **set smon vlan enable** commands to enable existing priority and VLAN SMON sessions for the specified port(s) on the switch.

The following example:

- Creates an SMON priority session for port ge.1.1
- Enables SMON priority for port ge.1.1
- Displays statistics for priority 0 for the enabled port:

```
S Chassis(rw)->set smon priority create ge.1.1
S Chassis(rw)->set smon priority enable ge.1.1
S Chassis(rw)->show smon priority ge.1.1 priority 0
Show Priority Statistics
------------------------
Interface = ge.1.1
Owner     = none
Creation  = 0 days 0 hours 37 minutes 36 seconds
```

```
Status    = enabled
--------------------

Priority 0 Packets              Octets
----------
Total      3477                 256168
Overflow   0                    0

S Chassis(rw)->
```

The following example:

- Creates an SMON VLAN session for port ge.1.1
- Enables SMON VLAN monitoring for port ge.1.1
- Displays statistics for VLAN 1 for the enabled port:

```
S Chassis(rw)->set smon vlan create ge.1.1
S Chassis(rw)->set smon vlan enable ge.1.1
S Chassis(rw)->show smon vlan vlan 1
Show VLAN Statistics
--------------------
Interface = ge.1.1
Owner     = none
Creation  = 20 days 1 hours 44 minutes 27 seconds
Status    = enabled
--------------------

VLAN 1                Packets               Octets
Total                 3728                  433041
Overflow              0                     0
NonUnicast            2660                  174336
NonUnicast Overflow   0                     0

S Chassis(rw)->
```

Configuring Network Monitoring

This section provides details for the configuration of network monitoring on the S-Series products.

Table 32-2 lists network monitoring parameters and their default values.

Table 32-2 Default Network Monitoring Parameters

Parameter	Description	Default Value
history buffer	The number of lines of CLI input that are placed in a buffer for redisplay.	20 lines
buckets	The number of RMON history entries to maintain.	50 entries

Table 32-2 Default Network Monitoring Parameters (continued)

Parameter	Description	Default Value
interval	The period between RMON history or alarm sampling.	history = 1800 seconds alarm = 3600 seconds
owner	The RMON management station entity for a statistics or alarm context.	monitor
type	The RMON alarm monitoring method or property, RMON event, or TopN counter type.	alarm = absolute event = none topN = inpackets
startup	The RMON alarm type generated when an event is first enabled.	rising
rthresh	The RMON minimum threshold for causing a rising alarm.	0 events
fthresh	The RMON maximum threshold for causing a falling alarm.	0 events
revent	The RMON index event number to be triggered when the rising threshold is crossed.	0
fevent	The RMON index event number to be triggered when the falling threshold is crossed.	0
alarm, event, topN, matrix or host status	Whether an entry is enabled or disabled.	disabled
channel action	The RMON channel entry action.	packets are accepted on filter matches
channel control	The RMON channel flow of data control state.	off
channel event status	The event to be triggered when the channel is on and a packet is accepted.	ready
channel description	A user configured description of the channel.	none.
capture action	The RMON capture entry action when the buffer is full.	lock
capture offset	The RMON capture first octet from each packet to retrieve.	0
capture asksize	The RMON capture requested maximum octets to save in the buffer.	1
capture slice	The RMON capture maximum number of octets from each packet to be saved to the buffer.	100
capture loadsize	The RMON capture maximum number of cotets from each packet to be downloaded from the buffer.	100

To optionally change the size of the history buffer, use the **set history** command, specifying the size of the history buffer. The **default** option configures the specified history buffer setting to persist for all future sessions. Otherwise, the setting only affects this session.

This example shows how to set the size of the command history buffer to 25 lines and make this the default setting:

```
S Chassis(rw)->set history 25 default
```

To optionally send a message to one or all active users on this switch, use the tell command, specifying an individual destination or all users. The **dest** option specifies the user and location in the user@location format.

This example shows how to tell user rw@134.141.192.18 about a system reset:

```
S Chassis(rw)->show users
  Session  User  Location
  -------- ----- -------------------------
* console  admin console (via com.1.1)
  telnet   rw    134.141.192.18
S Chassis(rw)->tell rw@134.141.192.18 "System reset in 15 minutes"
```

User rw@134.141.192.18 will receive:

```
Message from admin@console: "System reset in 15 minutes"
```

Table 32-3 describes network diagnostics commands.

Table 32-3 Network Diagnostics Commands

Task	Command
To determine the availability of another node on the network: • **-s** *bytes* – (Optional) Specifies the number of data bytes to be sent. • **-c** *count* – (Optional) Number of ping packets. • **-n** – (Optional) Avoids any communications with nameservers. • **-p** *pattern* – (Optional) Specify up to a 16 bit hexadecimal pattern to fill outgoing packet with (ex. -p ff). • **-t** *hops* – (Optional) Specifies the maximum number of hops for the ping. • **-I** *interface* – (Optional) Source IP Interface. • **-S** *ip-address* – (Optional) Source IP address. • **-Q** *service-type* – (Optional) Specifies the Type of Service in the IPv4 header or the traffic class in the IPv6 header. • **-r** – (Optional) Bypass the normal routing tables and send directly to a host on an attached network. • **-i** – (Optional) Specifies the time in milliseconds to wait for ping timeouts and between sending ping packets. • **-v** – (Optional) Forces ping to a specific ip version. • **-V** *router* – (Optional) Specify a virtual router name or number for this ping. *host* – Specifies the IP address or a hostname of the receiving device.	**ping** [**-s** *bytes*] [**-c** *count*] [**-n**] [**-p** *pattern*] [**-t** *milliseconds*] [**-I** *interface*] [**-S** *ip-address*] [**-Q** *service-type*] [**-r**] [**-i** *milliseconds*] [**-v** {4 | 6}] [**-V** *router*] *host*

Table 32-3 Network Diagnostics Commands (continued)

Task	Command
To display a hop-by-hop path from the device to a specific destination host: **-d** *ip-address* – (Optional) Performs a reverse lookup (finds a hostname that matches the specified IP address). **-F** – (Optional) Specifies that the traceroute packet should not be fragmented. **-f** *first-TTL* – (Optional) Specifies the maximum Time-To-Live (TTL) used in the first outgoing probe packets. **-I** – (Optional) Specifies that ICMP echo requests should be used instead of UDP datagrams. **-i** *source-interface* – (Optional) Specifies the IP source interface (for example vlan.0.5 for VLAN 5). **-m** *max-ttl* – (Optional) Specifies the maximum Time-To-Live (TTL) for outgoing packets. **-n** *host-ip-address* – (Optional) Specifies that name server contact should be avoided. **-p** *udp-dest-port* – (Optional) Specifies the initial UDP destination port. For each sent probe the UDP destination port is increased by one. **-q** *number-of-probes* – (Optional) Specifies the number of probes to send out for each hop. **-r** – (Optional) Specifies that normal host routing tables should be bypassed. **-s** *source-ip-address* – (Optional) Specifies the source IP address for the traceroute probes. **-t** *tos* – (Optional) Specifies the Type-of-Service (ToS) for IPv4 or the traffic class for IPv6. **-v** *version* – (Optional) Forces traceroute to use either IPv4 or IPv6. **-V** *router* – (Optional) Specifies the virtual router to use for this traceroute. **-w** *period* (Optional) Specifies the time in seconds to wait for a response to a probe. **-x** – (Optional) Specifies that traceroute should not calculate checksum. **host** *host* – Specifies an IP address or a host to find a route to.	**traceroute [-d** *ip-address***] [-F] [-f** *first_ttl***] [-I] [-i** *interface***] [-m** *max_ttl***] [-n] [-p** *port***] [-q** *nqueries***] [-r] [-s** *source-address***] [-t** *tos***] [-v {4 | 6}] [-V** *router***][-w** *waittime***] [-x]** *host*
To query a name server to translate hostnames to IP addresses or IP addresses to hostnames: **-x** – (Optional) Specifies that a reverse lookup should be performed. If this parameter is used, then you must specify an IP address as the host variable. **-v** {**4** | **6**} – (Optional) Specifies the IP version for this name server lookup. *host* – Specifies the host name, or an IP address, in the case of a reverse lookup.	**nslookup** [**-x**] [**-v** {**4** | **6**}] *host*

Procedure 32-1 describes how to configure SMON. SMON commands can be entered from any command mode.

Procedure 32-1 Configuring SMON

Step	Task	Command(s)
1.	Optionally, first create and then enable an SMON session for the collection of Ethernet priority statistics for the specified port(s).	**set smon priority** {**create** | **enable**} port-string [owner]
2.	Optionally, first create and then enable an SMON session for the collection of VLAN statistics for the specified ports and VLANS	**set smon vlan** {**create** | **enable**} *port-string* [*owner*]

Procedure 32-2 describes how to configure RMON. RMON commands can be entered from any command mode.

Procedure 32-2 Configuring Remote Network Monitoring

Step	Task	Command(s)
1.	Optionally, configure RMON to to create entries that record statistics measured by the RMON probe for each specified interface. • *index* – Specifies the index number for this entry • *port-string* – assigns this entry to a specific port • **owner** – (Optional) Specifies the management station owner for this entry • **direction** – (Optional) Specifies the direction in which RMON statistics are collected	**set rmon stats** *index* [*port-string*] [*owner*] [**direction** {**rx+tx** | **rx** | **tx**}]
2.	Optionally, specify the maximum number and period for recorded statistical samples from a network. • *index* – Specifies the index number for this entry • *port-string* – assigns this entry to a specific port • **bucket** – (Optional) Specifies the maximum number of entries to maintain • **interval** – (Optional) Specifies the period between samples in seconds • **owner** – (Optional) Specifies the management station owner for this entry • **direction** – (Optional) specifies the direction in which RMON history is collected	**set rmon history** *index* [*port-string*] [**buckets** *buckets*] [**interval** *interval*] [**owner** *owner*] [**direction** {**rx+tx** | **rx** | **tx**}]

Procedure 32-2 Configuring Remote Network Monitoring (continued)

Step	Task	Command(s)
3.	Configure RMON probe variable thresholds that will trigger an alarm if crossed by a sampled probe. • *index* - Specifies the entry for this set of alarm properties • **interval** - (Optional) Specifies the period between samples in seconds • **object** - (Optional) Specifies the MIB object to be monitored • **type** - (Optional) Specifies a monitoring method • **startup** - (Optional) Specifies the alarm type generated when this event is first enabled • **rthresh** - (Optional) Specifies the minimum threshold that will cause a rising alarm • **fthresh** - (Optional) Specifies the minimum threshold that will cause a falling alarm • **revent** - (Optional) Specifies the index number of the RMON event to be triggered when the rising threshold is crossed • **fevent** - (Optional) Specifies the index number of the RMON event to be triggered when the falling threshold is crossed • **owner** - (Optional) Specifies the management station owner for this entry	**set rmon alarm properties** *index* [**interval** *interval*] [**object** *object*] [**type** {**absolute** | **delta**}] [**startup** {**rising** | **falling** | **either**}] [**rthresh** *rthresh*] [**fthresh** *fthresh*] [**revent** *revent*] [**fevent** *fevent*] [**owner** *owner*]
4.	Enable a configured alarm entry.	**set rmon alarm status** *index* **enable**
5.	Configure RMON probe variable thresholds that will trigger an event if crossed by a sampled probe. • *index* - Specifies the entry for this set of event properties • **description** - (Optional) Specifies a text string description for this event • **type** - (Optional) Specifies the event notification type for this entry • **community** - (Optional) Specifies an SNMP community name to use if the message type is set to trap • **owner** - (Optional) Specifies the management station owner for this entry	**set rmon event properties** *index* [**description** *description*] [**type** {**none** | **log** | **trap** | **both**}] [**community** *community*] [**owner** *owner*]
6.	Enable a configured event entry.	**set rmon event status** *index* **enable**

Procedure 32-2 Configuring Remote Network Monitoring (continued)

Step	Task	Command(s)
7.	Configure RMON to record statistics associated with each host discovered on the network. • *index* - Specifies the entry value for this set of host properties • *port-string* - Specifies the port on which RMON will monitor hosts • *owner* - (Optional) Specifies the management station owner for this entry	**set rmon host properties** *index port-string* [*owner*]
8.	Configure an RMON topN properties entry for the generation of tables that describe hosts that top a list ordered by one of their statistics. • *index* - Specifies the entry number for this set of RMON topN properties • **hindex** - (Optional) Specifies the host table index number • **rate** - (Optional) Specifies the counter type to activate: InPackets, OutPackets, InOctets, OutOctets, OutErrors, Broadcast packets, and Multicast packets • **duration** - (Optional) Specifies the sampling interval in seconds • **size** - (Optional) Specifies the maximum number of entries to maintain • **owner** - (Optional) Specifies the management station that configured this entry	**set rmon topn properties** *index* [**hindex** *hindex*] [**rate** {**inpackets** | **outpackets** | **inoctets** | **outoctets** | **errors** | **bcast** | **mcast**}] [**duration** *duration*] [**size** *size*] [**owner** *owner*]
9.	Enable an RMON topN entry.	**set rmon topN status** *index* **enable**
10.	Configure an RMON matrix properties entry for recording statistics for conversations between two IP addresses. • *index* - Specifies the entry value for this set of matrix properties • *port-string* - Specifies the port on which RMON will monitor conversations • *owner* - (Optional) Specifies the management station owner for this entry	**set rmon matrix properties** *index port-string* [owner]
11.	Enable an RMON matrix entry.	**set rmon matrix status** *index* **enable**

Procedure 32-2 Configuring Remote Network Monitoring (continued)

Step	Task	Command(s)
12.	Configure an RMON channel entry to match packets by a filter equation. • *index* - Specifies the entry value for this channel entry • *port-string* - Specifies the port on which RMON will monitor traffic • **accept** - (Optional) Specifies the filters action for this entry • **control** - (Optional) Enables or disables control of the flow of data through this channel • **onevent** - (Optional) Specifies the index of the RMON event that turns this channel on • **offevent** - (Optional) Specifies the index of the RMON event that turns this channel off • **event** - (Optional) Specifies the event to be triggered when the channel is on and a packet is accepted • **estatus** - (Optional) Specifies the event status • **description** - (Optional) Specifies a description for this channel • *owner* - (Optional) Specifies the management station owner for this entry	**set rmon channel** *index port-string* [**accept** {**matched** | **failed**}] [**control** {**on** | **off**}] [**onevent** *onevent*] [**offevent** *offevent*] [**event** *event*] [**estatus** {**ready** | **fired** | **always**}] [**description** *description*] [**owner** *owner*]
13.	Configure an RMON filter entry. • *index* - Specifies the entry value for this filter entry • *port-string* - Specifies the channel on which RMON will monitor this filter entry • **offset** - Specifies the offset from the beginning of the packet to look for matches • **status** - (Optional) Specifies packet status bits that are to be matched • **smask** - (Optional) Specifies the mask applied to status to indicate which bits are significant • **snotmask** - (Optional) Specifies the inversion mask that indicates which bits should be set or not set • **data** - (Optional) Specifies the data to be matched • **dmask** - (Optional) Specifies the mask applied to data to indicate which bits are significant • **dnotmask** - (Optional) Specifies the inversion mask that indicates which bits should be set or not set • *owner* - (Optional) Specifies the management station owner for this entry	**set rmon filter** *index channel_index* [**offset** *offset*] [**status** *status*] [**smask** *smask*] [**snotmask** *snotmask*] [**data** *data*] [**dmask** *dmask*] [**dnotmask** *dnotmask*] [**owner** *owner*]

Procedure 32-2 Configuring Remote Network Monitoring (continued)

Step	Task	Command(s)
14.	Configure RMON capture to capture packets upon a filter match. • *index* - Specifies an entry number for this capture entry • *channel* - Specifies the channel to which this capture entry will be applied • **action** - (Optional) Specifes buffer behavior when it is full • **slice** - (Optional) Specifies the maximum number of octets from each packet to be saved in a buffer • **loadsize** - (Optional) Specifies the maximum number of octets from each packet to be downloaded from the buffer • **offset** - (Optional) Specifies the number octets from each packet to be retrieved • **asksize** - (Optional) Specifies the maximum number of octets that will be saved in the buffer • **owner** - (Optional) Specifies the name of the management station that configured this entry **Note:** Configuring RMON capture causes hardware based forwarding to be disabled, resulting in all traffic from the port to be forwarded by the CPU.	**set rmon capture** *index* {*channel* [**action** {**lock** | **wrap**}] [**slice** *slice*] [**loadsize** *loadsize*] [**offset** *offset*] [**asksize** *asksize*] [**owner** *owner*]}

Table 32-4 describes how to manage network monitoring.

Table 32-4 Managing Network Monitoring

Task	Command
To disconnect from a console or Telnet session:	**disconnect** {*ip-address* | **console**}
To disable SMON priority counters collection for the specified port(s), without clearing the created session:	**set smon priority disable** *port-string* [owner]
To disable SMON VLAN counters collection for the specified selected VLANs, specified using VTAP port strings, without clearing the created session:	**set smon vlan disable** *port-string* [*owner*]
To clear an existing SMON priority counters session for the specified port(s):	**clear smon priority** [*port-string*]
To clear an existing SMON VLAN counters session for the specified port(s):	**clear smon vlan** [*port-string*]
To delete one or more RMON statistics entries:	**clear rmon stats** {*index* | **to-defaults**}
To delete one or more RMON statistics entries:	**clear rmon stats** {*index-list* | **to-defaults**}
To delete one or more RMON history entries:	**clear rmon history** {*index-list* | **to-defaults**}
To delete an RMON alarm entry:	**clear rmon alarm** *index*

Table 32-4 Managing Network Monitoring (continued)

Task	Command
To delete an RMON event entry and any associated log entries:	**clear rmon event** *index*
To delete an RMON host entry:	**clear rmon host** *index*
To delete an RMON topN entry:	**clear rmon topN** *index*
To delete an RMON matrix entry:	**clear rmon matrix** *index*
To delete an RMON channel entry:	**clear rmon channel** *index*
To delete an RMON filter entry:	**clear rmon filter** *index*
To delete an rmon capture entry:	**clear rmon capture** *index*

Table 32-5 describes how to display network monitoring information and statistics.

Table 32-5 Displaying Network Monitoring Information and Statistics

Task	Command
To display the contents of the CLI history buffer:	**history**
To display the current history buffer size setting:	**show history**
To display switch connection statistics for all or the specified protocol:	**show netstat** [**icmp** | **ip** | **stats** | **tcp** | **udp**]
To display information for the active console port or Telnet sessions on the switch:	**show user**
To display SMON priority statistics counters for all or the specified port(s) and priorities:	**show smon priority** [*port-string*] [**priority** *priority*]
To display SMON VLAN statistics counters for all or the specified VLAN(s):	**show smon vlan** [*port-string*] [**vlan** *vlan-id*]
To display RMON statistics for one or more ports:	**show rmon stats** [*port-string*] [**wide**] [**bysize**]
To display RMON history properties and statistics:	**show rmon history** [*port-string*] [**wide**] [**interval** {**30sec** | **5min** | **25min**}]
To display RMON alarm entries:	**show rmon alarm** [*index*]
To display RMON event entry properties:	**show rmon event** [*index*]
To display RMON properties and statistics associated with each host discovered on the network:	**show rmon host** [*port-string*] [**address** | **creation**]
To display RMON TopN properties and statistics:	**show rmon topN** [*index*]
To display RMON matrix properties and statistics:	**show rmon matrix** [*port-string*] [**source** | **dest**]
To display RMON channel entries for one or more ports:	**show rmon channel** [*port-string*]
To display one or more RMON filter entries	**show rmon filter** [**index** *index* | **channel** *channel*]
To display RMON capture entries and associated buffer control entries:	**show rmon capture** [*index*] [**nodata**]

33

NetFlow Configuration

This document describes the NetFlow feature and its configuration on Extreme Networks S-Series switch/routers.

Using NetFlow in Your Network

NetFlow is a flow-based data collection protocol that provides information about the packet flows being sent over a network. NetFlow collects data by identifying unidirectional IP packet flows between a single source IP address/port and a single destination IP address/port, using the same Layer 3 protocol and values found in a fixed set of IP packet fields for each flow. NetFlow collects identified flows and exports them to a NetFlow collector. Up to four NetFlow collectors can be configured on an S-Series device. A NetFlow management application retrieves the data from the collector for analysis and report generation.

Standard system feedback is simply not granular enough to provide for such network requirements as planning, user or application monitoring, security analysis, and data mining. For example, because of its ability to identify and capture network flows, NetFlow:

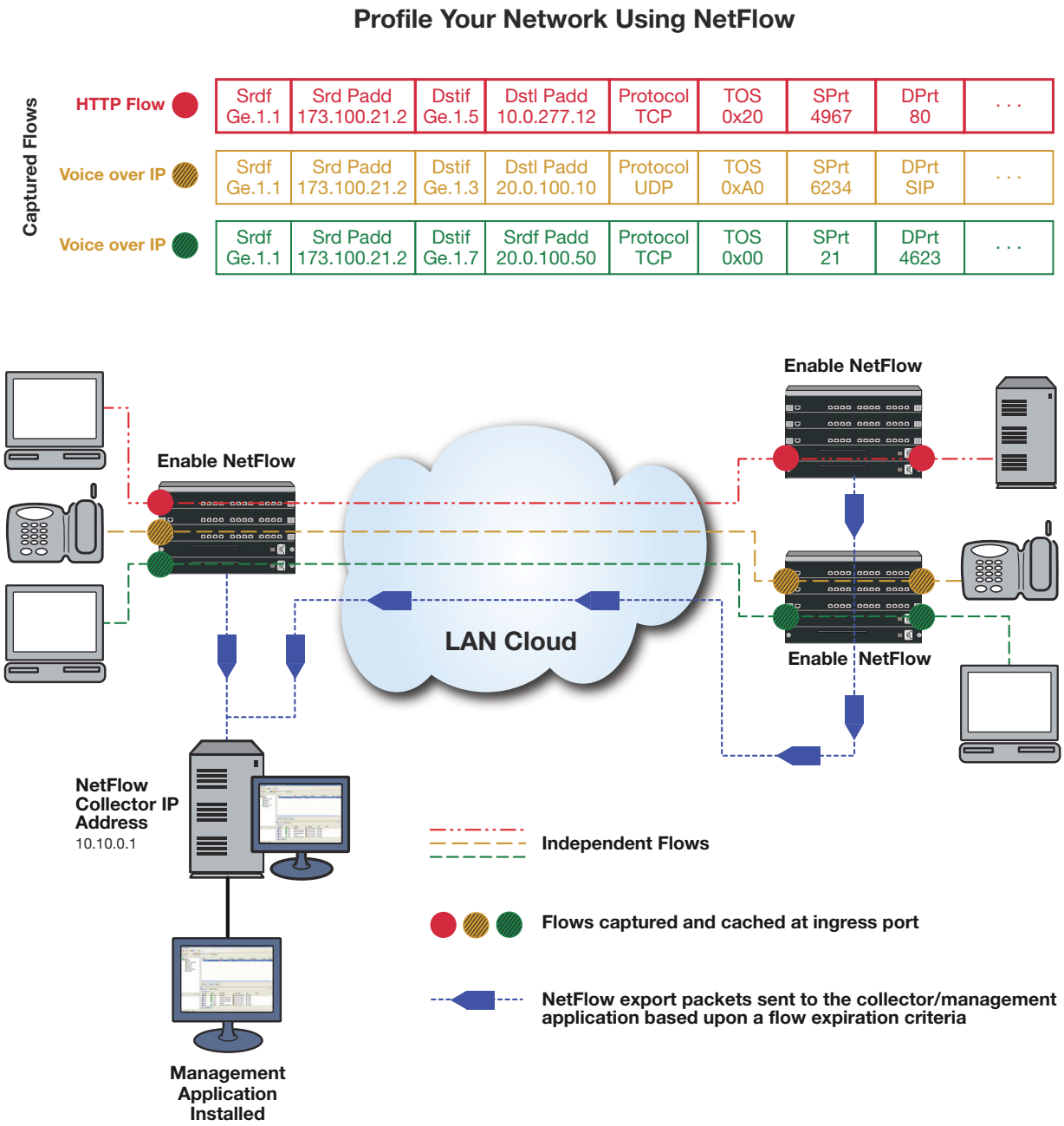
- Provides a means to profile all flows on your network over a period of time. A network profile provides the granularity of insight into your network necessary for such secure network functionality as establishing roles with policy and applying QoS to policy.
- Provides a means of isolating the source of DoS attacks allowing you to quickly respond with a policy, ACL, QoS change, or all of these to defeat the attack.
- Can identify the cause of an intermittently sluggish network. Knowing the cause allows you to determine whether it is an unexpected, but legitimate, network usage that might be rescheduled for low usage time blocks, or maybe an illegitimate usage of the network that can be addressed by speaking to the user.
- Can look into the flows that transit the network links, providing a means of verifying whether QoS and policy configurations are appropriately configured for your network.

- Can understand your network's flow characteristics, allowing for better planning when transitioning to new applications or services.

Implementing NetFlow

Having a profile of captured flows that transit your network over time is a crucial first step in implementing a secure network. This NetFlow profile provides you with a good understanding of the actual group and individual behaviors that make up the roles you set by policy and to which you apply QoS. A profile can also be very helpful during network planning exercises, such as projecting how a network might react to the introduction of a new application prior to actual implementation. Figure 33-1 illustrates an example of a NetFlow network profile setup.

Figure 33-1 NetFlow Network Profile Example



To complete a NetFlow network profile, enable NetFlow on all ports where packet flows aggregate. At the top of Figure 33-1 you will find an abbreviated sample of the independent flow records that are captured at each NetFlow-enabled port. These flow records will be retained locally in a cache until a flow expiration criteria has been met. As shown, when one of the flow expiration criteria is met, NetFlow export packets are then sent to the NetFlow collector server(s), where a collector and management application has been installed. The management application will process the records and generate useful reports. These reports provide you with a clear picture of the flows that traverse your network, based upon such data points as source and destination address, start and end time, application, and packet priority.

The following steps provide a high-level overview of a NetFlow implementation:

1. Determine the business or network purpose of the information NetFlow will provide you.
2. Choose a collector and management application(s), such as Extreme Networks SIEM, best suited for the purpose for which you are collecting the data. Install the application(s) on the NetFlow collector server(s).
3. Identify the paths used by the data to be collected by NetFlow.
4. Identify the "choke point" interfaces where the IP packet flows you want NetFlow to capture aggregate.
5. Enable NetFlow on the identified interfaces.
6. Identify up to four NetFlow collector servers by configuring the IP address for each collector.
7. Use the data reporting generated by the NetFlow management application to address the purpose determined in step 1.

Understanding Flows

The concept of a flow is critical to understanding NetFlow. A flow is a stream of IP packets in which the values of a fixed set of IP packet fields is the same for each packet in the stream. A flow is identified by a set of key IP packet fields found in the flow. Each packet containing the same value for all key fields is considered part of the same flow, until flow expiration occurs. If a packet is viewed with any key field value that is different from any current flow, a new flow is started based upon the key field values for that packet. The NetFlow protocol will track a flow until an expiration criteria has been met, up to a configured number of current flows.

The data captured for each flow is different, based on the NetFlow export version format supported by the network device. This data can include such items as packet count, byte count, destination interface index, start and end time, and next hop router. See "NetFlow Version 5 Record Format" on page 33-11 for NetFlow Version 5 template data field descriptions and "NetFlow Version 9 Templates" on page 33-12 for NetFlow Version 9 template data field descriptions.

Flow Expiration Criteria

Flow data records are not exported by the network switch to the NetFlow collector(s) until expiration takes place. There are two timers that affect flow expiration: the NetFlow active and inactive timers.

The active timer determines the maximum amount of time a long lasting flow will remain active before expiring. When a long lasting active flow expires, due to the active timer expiring, another flow is immediately created to continue the ongoing flow. It is the responsibility of the management application on the NetFlow collector to rejoin these multiple flows that make up a single logical flow. The active timer is configurable in the CLI (see "Configuring the Active Flow Export Timer" on page 33-6).

The inactive timer determines the length of time NetFlow waits before expiring a given flow once that flow has stopped. The inactive timer is a fixed value of 40 seconds and cannot be configured.

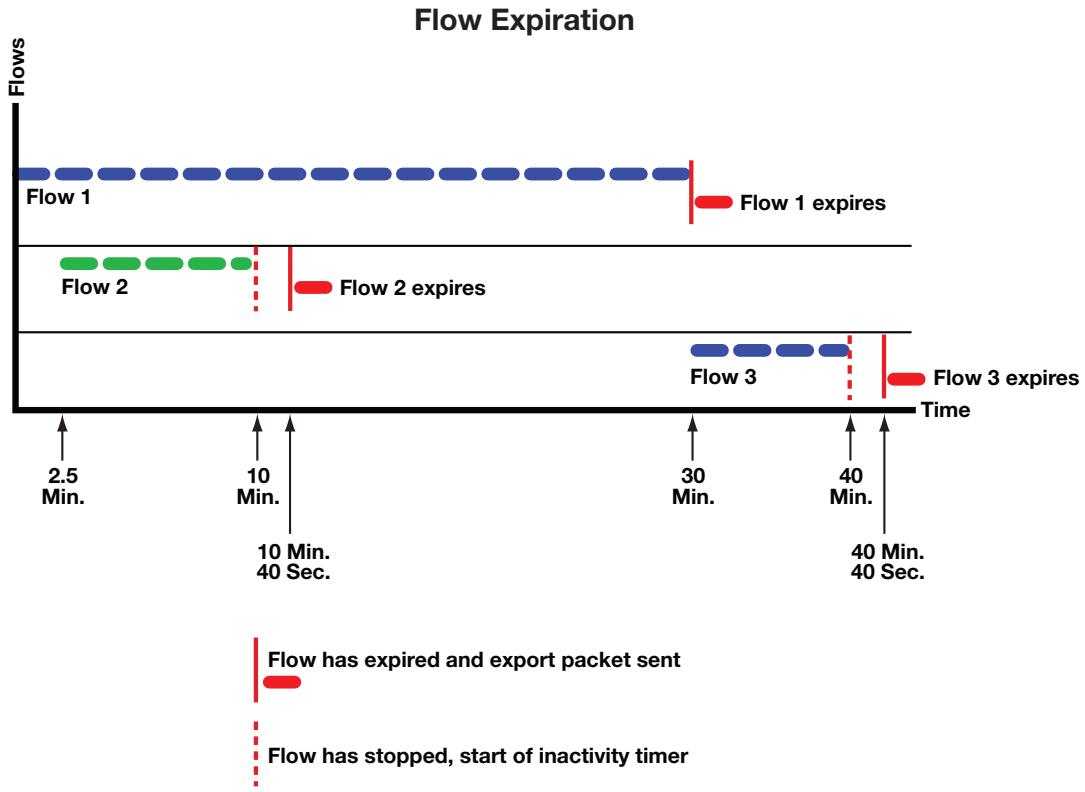
Rules for expiring NetFlow cache entries include:

- Flows which have been idle for 40 seconds (fixed value in firmware) are expired and removed from the cache.
- Long lived flows are expired and removed from the cache. (Flows are not allowed to live more than 30 minutes by default; the underlying packet conversation remains undisturbed).
- Flows associated with an interface that has gone down are automatically expired.

Figure 33-2 provides a graphic depiction of how these timers interact. Flows 1 and 3 show a single long lasting logical flow. Flow 1 times out and expires at 30 minutes, the active timer length. Because the flow expires, an export packet is sent to the NetFlow collector. Flow 3 continues this long lasting flow for another 10 minutes. At time 40 minutes the flow ends. The 40 second inactive timer initiates and expires at 40 minutes and 40 seconds resulting in an export packet to the NetFlow collector for flow 3. At the NetFlow collector, the management application joins the two flows into a single logical flow for purposes of analysis and reporting.

Flow 2 is a 7.5-minute flow that never expires the active timer. It begins at 2.5 minutes and ends at 10 minutes. At 10 minutes the inactive timer commences and expires the flow at 10 minutes and 40 seconds. At this time, NetFlow sends an export packet for the flow to the NetFlow collector for processing.

Figure 33-2 Flow Expiration Timers



Deriving Information from Collected Flows

On each collection server, a third-party NetFlow collector application correlates the received records and prepares them for use by the NetFlow management application. (In some cases the collector and management applications are bundled in a single application.) The management application retrieves the flow records, combines flows that were broken up due to expiration rules, and aggregates flows based upon common values, before processing the data into useful reports viewable by the network administrator.

Correlated reports can be the basis for such information categories as:

- Understanding who is originating and receiving the traffic
- Characterizing the applications that are utilizing the traffic
- Examining flows by priority
- Characterizing traffic utilization by device
- Examining the amount of traffic per port

Configuring NetFlow on the S-Series

The S-Series modules support NetFlow. NetFlow is disabled by default on all devices at device startup.

Extreme Networks S-Series Implementation

The Extreme Networks S-Series flow-based architecture provides a powerful mechanism for collecting network flow statistics, with reporting capacity that scales with the addition of each S-Series module. For each flow, packet and byte count statistics are collected by the S-Series forwarding hardware. The flow report generation logic is distributed, permitting each module to report flows on its own ports.

The Extreme Networks S-Series implementation enables the collection of NetFlow data on both switched and routed frames, allowing S-Series modules in all areas of a network infrastructure to collect and report flow data. Routing does not need to be enabled to utilize NetFlow data collection. Flow detail depends on the content of the frame and the path the frame takes through the switch.

NetFlow can be enabled on all ports on an S-Series device, including fixed front panel ports, LAG ports and NEM ports. Router interfaces which map to VLANs may not be enabled directly.

NetFlow records are generated only for flows for which a hardware connection has been established. As long as the network connection exists (and NetFlow is enabled), NetFlow records will be generated. Flows that are switched in firmware (soft forwarded) will not have NetFlow records reported. For flows that are switched, the S-Series firmware reports the source and destination IfIndexes as physical ports. For flows that are routed, the S-Series firmware reports the source and destination ifIndexes as routed interfaces.

In the case of a LAG port, the module(s) that the physical ports are on will generate NetFlow records independently. They will however, report the source ifIndex as the LAG port. The Flow Sequence Counter field in the NetFlow Header is unique per module. The Engine ID field of the NetFlow Header is used to identify each unique module.

Configuring the Active Flow Export Timer

The active flow export timer, also referred to as the export interval, sets the maximum amount of time an active flow will be allowed to continue before expiration for this system. Should the active timer expire and the flow terminate, the underlying flow continues as a separate flow. It is the responsibility of the management application to recognize the multiple flows as a single logical flow for analysis and reporting purposes. The active flow export timer defaults to 30 minutes.

Notes: Some NetFlow management applications expect to see export packets prior to some set interval that is often as low as 1 minute. Check the documentation for your management application and make sure that the export interval is configured for a value that does not exceed that value.

Use the **set netflow export-interval** command to change the active flow export timer value for each system.

Use the **clear netflow export-interval** command to reset the active flow export timer to its default value.

Configuring the NetFlow Collector IP Address

Expired NetFlow records are bundled into NetFlow export packets and sent to the NetFlow collector using the UDP protocol. Configuring the IP address of the NetFlow collector destination determines where expired NetFlow records for this system are sent. Up to four NetFlow collectors may be configured for each system. Multiple systems may share one or more NetFlow collectors. You can optionally specify the UDP port to be used on the NetFlow collector. By default, no NetFlow collector is configured on a system.

If you attempt to enter five collector destinations the following error displays:

```
Set failed.  If previously configured, you must "clear netflow export-destination"
first.
```

This message indicates that you have configured the maximum number of export destinations for the device. Remove a configured export destination using the **clear netflow export-destination** *ip-address* command before adding an additional export destination.

Use the **set netflow export-destination** command to configure the IP address of a NetFlow collector for this system and optionally set the UDP port.

Use the **clear netflow export-destination** command to clear the specified NetFlow collector configuration.

Configuring the NetFlow Export Version

The Extreme Networks S-Series supports NetFlow export versions 5 and 9. The default export version is 5.

The primary difference between the two versions is that version 5 is a fixed data record without multicast support, where version 9 is a flexible, extensible, template-based data record that provides the complete ifIndex value and 64-bit counters.

With NetFlow version 5, packets are made up of a series of data records and are exported to the collection server when the maximum number of NetFlow records is reached.

When transmitting NetFlow Version 5 reports, the S-Series module uses "NetFlow interface" indexes. Normally these would be actual MIB-2 ifIndex values, but the Version 5 record format limits the values to 2 bytes, which is not sufficient to hold 4-byte ifIndexes. NetFlow collector applications that use the in/out interface indexes to gather SNMP data about the interface (such as ifName) must translate the interface indexes using the Extreme Networks MIB etsysNetFlowMIB (1.3.1.6.1.4.1.5624.1.2.61).

With NetFlow version 9, packets are made up of templates containing a set of data records. Templates are sent after the period configured for the template timeout when a module or collection server first boots up. Data records for version 9 cannot be processed without an up-to-date template. Collectors ignore incoming packets until a template arrives. Templates are refreshed periodically based upon a packet refresh rate and timeout period. Setting the appropriate refresh rate for your S-Series device must be determined, since the default settings of a 20-packet refresh rate and a 30-minute timeout may not be optimal for your environment. See "Configuring NetFlow Export Version Refresh" on page 33-7.

NetFlow Version 9 records generated by S-Series modules use true MIB-2 ifIndex values since the template mechanism permits transmission of 4-byte ifIndexes. Version 9 also uses 8-byte packet and byte counters, so they are less likely to roll over. Check with your collector provider to determine if they provide the necessary support.

The current Extreme Networks Version 9 implementation:

- Does not support aggregation caches.
- Provides 15 IPv4 and 15 IPv6 predefined templates. The S-Series firmware automatically selects the appropriate template for each flow depending on whether the flow is routed or switched, whether it is a TCP/UDP packet or not, and contains fields appropriate to the data records supported in the template. See Table 33-5 on page 33-13 for a listing of the header fields supported by the NetFlow Version 9 templates. See Table 33-6 on page 33-13 for a listing of the base data record fields supported by all NetFlow Version 9 templates. See Table 33-7 on page 33-14 for a listing of the additional template specific data record fields supported by the NetFlow Version 9 templates. See Table 33-8 on page 33-14 for a listing of IPv4 and IPv6 Version 9 NetFlow templates by template ID and description.

Use the **set netflow export-version** {**5**|**9**} command to set the NetFlow export version.

Use the **clear netflow export-version** command to reset the export version to the default value of Version 5.

Configuring NetFlow Export Version Refresh

Version 9 template records have a limited lifetime and must be periodically refreshed. Templates are retransmitted when either:

- the packet refresh rate is reached, or
- the template timeout is reached.

Template refresh based on the timeout period is performed on every module. Since each S-Series module handles its own packet transmissions, template refresh based on number of export packets sent is managed by each module independently.

The refresh rate defines the maximum delay a new or restarted NetFlow collector would experience, before it learns the format of the data records being forwarded (from the template referenced by the data records). Refresh rates affect NetFlow collectors during their start up. Collectors must ignore incoming data flow reports until the required template is received.

The default behavior is for the template to be sent after 20 flow report packets are sent. Since data record packets are sent out per flow, a long FTP flow may cause the template timeout timer to expire before the maximum number of packets are sent. In any case a refresh of the template is sent at timeout expiration as well.

Setting the appropriate refresh rate for your S-Series device must be determined, because the default settings of a 20 flow report packet refresh rate and a 30-minute timeout may not be optimal for your environment. For example, a switch processing an extremely slow flow rate of, say, 20 flow report packets per half hour, would refresh the templates only every half hour using the default settings, while a switch sending 300 flow report packets per second would refresh the templates 15 times per second.

Enterasys recommends that you configure your S-Series device so it does not refresh templates more often than once per second.

Use the **set netflow template** to set the NetFlow export template refresh rate and timeout for this system.

Use the **clear netflow template** to reset the NetFlow export template refresh rate and timeout to the default values.

Configuring a NetFlow Port

NetFlow records are only collected on ports that are enabled for NetFlow.

Use the **set netflow port enable** command to enable NetFlow on the specified ports.

Use either the **set netflow port disable** or **clear netflow port** command to disable NetFlow on the specified ports.

Configuring the NetFlow Cache

Enabling the NetFlow Cache globally enables NetFlow on all S-Series modules for this system. When NetFlow recognizes a new flow on the ingress port, it creates a NetFlow record for that flow. The NetFlow record resides in the NetFlow cache for that port until an expiration event is triggered for that flow, at which time it is sent along with other expired flows in an export packet to the NetFlow collector for processing.

Use the **set netflow cache enable** command to enable NetFlow on this system.

Use either the **set netflow cache disable** or **clear netflow cache** command to globally disable NetFlow on this system.

Configuring Optional NetFlow Export Data

The export of optional source and destination MAC address and VLAN ID data is disabled by default. Including these export data options in the flow record makes the record larger and results in fewer records and exported packets.

If the **mac** option is enabled, both incoming source and destination MAC addresses are included in the export data for the collector.

If the **vlan** option is enabled, VLANs associated with both the ingress and egress interfaces are included in the export data for the collector.

Use the **set netflow export-data enable** {**mac** | **vlan**} command to enable either the MAC or VLAN export data.

Use the **set netflow export-data disable** {**mac** | **vlan**} command to disable either the MAC or VLAN export data.

Use the **clear netflow export-data** command to reset both MAC and VLAN optional export data configuration to the default value of disabled.

Displaying NetFlow Configuration and Statistics

Use the **show netflow** command to display the current configuration and export statistics for this system.

Use the **show netflow config** command to display the NetFlow configuration for a single or set of ports.

Use the **show netflow statistics export** command to display export statistics for this system.

Default NetFlow Settings for S-Series Systems

Table 33-1 provides a listing of the default NetFlow configuration settings for the S-Series systems.

Table 33-1 Default NetFlow Configuration Settings for S-Series Systems

Parameter	Description	Default Value
Cache Status	Whether NetFlow caching is globally enabled or disabled.	Disabled globally
Destination IP address	The IP address of the NetFlow collector which is the destination of the NetFlow UDP packets.	None
Export Interval	The time out interval when the NetFlow cache is flushed and the data is exported, if the maximum number of entries has not been reached.	30 minutes
Export Version	The NetFlow flow record format used when exporting NetFlow packets. Version can be either 5 or 9.	Version 5
Inactive flow timer	The number of seconds after a flow stops before NetFlow sends an export packet for that flow to the collector.	40 seconds (non-configurable)
Optional Export Data	Export data types that are disabled by default. These data types include source and destination MAC addresses and VLAN IDs associated with the ingress and egress interfaces for the flow.	Disabled
Port state	Whether NetFlow is enabled or disabled on a port.	Disabled
Refresh-rate	The number of flow report packets sent before NetFlow retransmits a template to the collector when using NetFlow Version 9.	20 flow report packets

Table 33-1 Default NetFlow Configuration Settings for S-Series Systems (continued)

Parameter	Description	Default Value
Timeout-period	When using NetFlow Version 9, the number of minutes NetFlow waits before retransmitting a template to the collector.	30 minutes

Procedure 33-1 provides a CLI example of a NetFlow setup. Steps 1 – 3 are required. Steps 4 – 6 are optional depending upon the needs of your configuration. All NetFlow commands can be configured in any command mode.

Procedure 33-1 Configuring NetFlow on S-Series Systems

Step	Task	Command(s)
1.	Enable NetFlow collection on the specified port.	**set netflow port** *port_string* **enable**
2.	Configure up to four NetFlow collector destination servers for this system.	**set netflow export-destination** *ip-address* [*udp-port*]
3.	Globally enable the NetFlow cache for this system.	**set netflow cache enable**
4.	Optionally, modify the active flow timer value for this system.	**set netflow export-interval** *interval*
5.	Optionally, change NetFlow record format between version 5 and version 9 for this system.	**set netflow export-version** *version*
6.	Optionally, enable NetFlow Version 9 MAC and VLAN export data.	**set netflow export-data enable** {**mac** | **vlan**}
7.	If using version 9, optionally modify the number of export packets sent that cause a template to be retransmitted by an individual S-Series module and the length of the timeout period, in minutes, after which a template is retransmitted by all modules in the system.	**set netflow template** {[**refresh-rate** *packets*] [**timeout** *minutes*]
8.	Verify any configuration changes made.	**show netflow config**

Terms and Definitions

Table 33-2 lists terms and definitions used in this NetFlow configuration discussion.

Table 33-2 NetFlow Configuration Terms and Definitions

Term	Definition
Active Flow Timer	A timer which specifies the maximum amount of time a flow may stay active. The ongoing flow continues to be tracked as a separate flow. It is the management application's responsibility to join these flows for analysis/reporting purposes.
Flow	A stream of IP packets that has not yet met an expiration criteria, in which the values of a set of key fields is the same for each packet in the stream.
Flow Record	A capture of information pertaining to a single flow within the NetFlow Cache based upon data type values supported by the NetFlow version format/template.
Inactive Flow Timer	A timer that determines how long a flow for which no packets are being received remains active.
NetFlow Cache	Contains the flow records for all currently active flows.

Table 33-2 NetFlow Configuration Terms and Definitions (continued)

Term	Definition
NetFlow Collector	An external location where a condensed and detailed history of flow information that entered each NetFlow-enabled switch or router is archived for use by the NetFlow management application.
NetFlow Export	A transport mechanism that periodically (based upon a timer or the number of flows accumulated in the cache) sends NetFlow data from the cache to a NetFlow collector for data analysis.
NetFlow Export Packet	A packet of flow records or version 9 templates (or both) that is periodically sent to the NetFlow collector based upon an export criteria.
NetFlow Management Application	An Extreme Networks SIEM or third-party software application(s) installed on the NetFlow collector, with client or browser access from a PC, capable of data reduction, monitoring, analysis, and/or troubleshooting specific to the purpose you are using NetFlow.
NetFlow Version	Primarily determines the data types supported and whether the format is fixed or in an extensible template.

NetFlow Version 5 Record Format

Table 33-3 provides a listing and description for the NetFlow Version 5 header fields. Table 33-4 provides a listing and description for NetFlow Version 5 data record fields. The contents of these data fields are used by the collector software application for flow analysis. Data fields are identified in the data record packet sent by the network switch to the collector. The data records contain the values specified by the format.

Table 33-3 NetFlow Version 5 Template Header and Data Field Support

NetFlow Version 5 Header	
Data Field	**Field Contains**
count	Number of flows exported in this packet (1-30).
sys_uptime	Current time in milliseconds since the export device booted.
unix_secs	Current count of seconds since 0000 UTC 1970.
unix_nsecs	Residual nanoseconds since 0000 UTC 1970.
flow_sequence	Sequence counter of total flows seen.
engine_type	Type of flow-switching engine.
engine_id	Slot number of the flow-switching engine.
sampling_interval	First two bits hold the sampling mode; remaining 14 bits hold value of sampling interval.
count	Number of flows exported in this packet (1-30).

Table 33-4 NetFlow Version 5 Data Record Field Format

NetFlow Version 5 Data Record Format	
Data Field	**Field Contains**
srcaddr	Source IP address of the device that transmitted the packet.
dstaddr	IP address of the destination of the packet.

Table 33-4 NetFlow Version 5 Data Record Field Format (continued)

NetFlow Version 5 Data Record Format	
Data Field	**Field Contains**
nexthop	IP address of the next hop router.
input	SNMP index of input interface.
output	SNMP index of output interface.
dPkts	Number of packets in the flow.
dOctets	Total number of Layer 3 bytes in the packets of the flow.
first	SysUptime at start of flow.
last	SysUptime at the time the last packet of the flow was received.
srcport	TCP/UDP source port number or equivalent.
dstport	TCP/UDP destination port number or equivalent.
pad1	Unused (zero) bytes.
tcp_flags	Cumulative OR of TCP flags.
prot	IP protocol type (for example, TCP = 6; UDP = 17).
tos	IP type of service (ToS).
src_as	Autonomous system number of the source, either origin or peer.
dst_as	Autonomous system number of the destination, either origin or peer.
src_mask	Source address prefix mask bits.
dst_mask	Destination address prefix mask bits.
pad2	Unused (zero) bytes.

NetFlow Version 9 Templates

The S-Series NetFlow Version 9 implementation supports 15 IPv4 (templates 256 through 271) and 15 IPv6 (templates 272 through 287) Version 9 templates. The templates are Extreme Networks defined supporting data record fields defined in the NetFlow standard. The contents of these data record fields are used by the collector software application for flow analysis. Ten base data record fields are included in all Version 9 templates. Up to an additional seven data record fields are included in the appropriate templates.

The S-Series platform implementation of the NetFlow Version 9 templates are detailed in the following tables:

- Table 33-5 on page 33-13 provides a listing and description of the supported NetFlow Version 9 header fields
- Table 33-6 on page 33-13 provides a listing and description of the supported NetFlow Version 9 base data record fields
- Table 33-7 on page 33-14 provides a listing of the supported additional template specific data record fields
- Table 33-8 on page 33-14 provides the template ID and a general description of each S-Series Version 9 template

Table 33-5 on page 33-13 details the NetFlow Version 9 template header fields supported by all Version 9 templates.

Table 33-5 NetFlow Version 9 Template Header Support

NetFlow Version 9 Header		
Data Field	**Description**	**Templates**
Format Version	NetFlow template Version 9	All Templates
Flow Record Count	The total number of records in the export packet, which is the sum of the options flow set records, template flowset records, and data flowset records.	All Templates
Sys Up Time	Time in milliseconds since this device was first booted.	All Templates
Unix Seconds	Time in seconds since 0000 UTC 1970, at which the export packet leaves the exporter.	All Templates
Flow Sequence Counter	Incremental sequence counter of all export packets sent from the exporter. This is an accumulative count that lets the collector know if any packets have been missed.	All Templates
Source ID	Engine Type (1 = Line Card). Engine ID (One based module slot number).	All Templates

Table 33-6 details the NetFlow Version 9 base data record fields supported by Version 9 templates. Base data record fields are supported by all IPv4 and IPv6 Version 9 templates. IPv4 specific data records are only supported by IPv4 templates. IPv6 specific data records are only supported by IPv6 templates.

Table 33-6 NetFlow Version 9 Template Data Record Field Support

NetFlow Version 9 Base Data Record Fields		
Data Field	**Description**	**Templates**
SIP	(Source) IPv4 or IPv6 address of the device that transmitted the packet.	256 - 271 IPv4 addresses 272 - 287 IPv6 addresses
DIP	(Destination) IPv4 or IPv6 address of the destination device.	256 - 271 IPv4 addresses 272 - 287 IPv6 addresses
Dest IfIndex	MIBII 32-bit ID of the interface on which the packet was transmitted.	All templates
Source IfIndex	MIBII 32-bit ID of the interface on which the packet was received.	All templates
Packet Count	The number of packets switched through this flow.	All templates
Byte Count	The number of bytes switched through this flow.	All templates
Start Time	sysUptime in milliseconds at which the first packet of this flow was switched.	All templates
Last Time	sysUptime in milliseconds at which the last packet of this flow was switched.	All templates
IP Protocol	IP protocol for this flow.	All templates
Source TOS	(Source) Type of service field value for this flow.	All templates

Table 33-7 details the additional NetFlow Version 9 data record fields specific to a given Version 9 template.

Table 33-7 NetFlow Version 9 Additional Template Specific Data Record Field Support

NetFlow Version 9 Additional Template Specific Data Record Fields		
Data Field	**Description**	**Templates**
Source MAC	Source MAC addresses for this flow.	**IPv4:** 257, 259, 261, 263, 265, 267, 269, 271 **IPv6:** 272, 274, 276, 278, 280, 282, 284, 286
Destination MAC	Destination MAC addresses for this flow.	**IPv4:** 257, 259, 261, 263, 265, 267, 269, 271 **IPv6:** 272, 274, 276, 278, 280, 282, 284, 286
Source VLAN	Source VLAN ID associated with the ingress interface for this flow.	**IPv4:** 258, 259, 262, 263, 266, 267, 270, 271 **IPv6:** 273, 274, 277, 278, 281, 282, 285, 286
Destination VLAN	Destination VLAN ID associated with the egress interface for this flow.	**IPv4:** 258, 259, 262, 263, 266, 267, 270, 271 **IPv6:** 273, 274, 277, 278, 281, 282, 285, 286
Layer 4 Source Port	TCP/UDP source port numbers (for example, FTP, Telnet, or equivalent).	**IPv4:** 260, 261, 262, 263, 268, 269, 270, 271 **IPv6:** 275, 276, 277, 278, 283, 284, 285, 286
Layer 4 Destination Port	TCP/UDP destination port numbers (for example, FTP, Telnet, or equivalent).	**IPv4:** 260, 261, 262, 263, 268, 269, 270, 271 **IPv6:** 275, 276, 277, 278, 283, 284, 285, 286
Next Hop Router	Specifies the BGP IPv4 or IPv6 next-hop address.	**IPv4:** 264, 265, 266, 267, 268, 269, 270, 271 **IPv6:** 279, 280, 281, 282, 283. 284, 285, 286

Table 33-8 provides a description of each IPv4 and IPv6 NetFlow Version 9 template per template ID.

Table 33-8 NetFlow Version 9 Templates

IPv4 Version 9 Templates	
Template ID	**Description**
256	Base switch template containing IPv4 base data record entries.
257	Switch and MAC ID template containing IPv4 base data record entries, along with source and destination MAC addresses.
258	Switch and VLAN ID template containing IPv4 base data record entries and source and destination VLAN IDs.

Table 33-8 NetFlow Version 9 Templates (continued)

259	Switch, MAC ID, and VLAN ID template containing IPv4 base data record entries, along with source and destination MAC addresses and source and destination VLAN IDs.
260	Switch and Layer 4 port template containing IPv4 base data record entries, along with source and destination Layer 4 ports.
261	Switch, Layer 4 port, and MAC ID template containing IPv4 base data record entries, along with source and destination layer 4 ports and source and destination MAC addresses.
262	Switch, Layer 4 port, and VLAN ID template containing IPv4 base data record entries, along with source and destination Layer 4 ports and source and destination VLAN IDs.
263	Switch, Layer 4 port , MAC ID, and VLAN ID template containing IPv4 base data record entries, along with source and destination Layer 4 port, source and destination MAC addresses and source and destination VLAN IDs.
264	Switch and IPv4 route ID template containing IPv4 base data record entries, along with the route next hop.
265	Switch, IPv4 route ID, and MAC ID template containing IPv4 base data record entries, along with the route next hop and source and destination MAC addresses.
266	Switch, IPv4 route ID, and VLAN ID template containing IPv4 base data record entries, along with the route next hop, and source and destination VLAN IDs.
267	Switch, IPv4 next hop, MAC ID, and VLAN ID template containing IPv4 base data record entries, along with the route next hop, source and destination MAC addresses, and source and destination VLAN IDs.
268	Switch, IPv4 route ID, and Layer 4 port template containing IPv4 base data record entries, along with the route next hop, and source and destination Layer 4 ports.
269	Switch, IPv4 route ID, Layer 4 port and MAC ID template containing IPv4 base data record entries, along with the route next hop, source and destination Layer 4 port, and source and destination MAC addresses.
270	Switch, IPv4 next hop, Layer 4 port and VLAN ID template containing IPv4 base data record entries, along with the route next hop, source and destination Layer 4 ports, and source and destination VLAN IDs.
271	Switch, IPv4 next hop, Layer 4 port, MAC ID, and VLAN ID template containing IPv4 base data record entries, along with the IPv4 next hop, source and destination Layer 4 ports, source and destination MAC addresses, and source and destination VLAN IDs.
IPv6 Version 9 Templates	
272	Base switch template containing IPv6 base data record entries.
273	Switch and MAC ID template containing IPv6 base data record entries, along with source and destination MAC addresses.
274	Switch and VLAN ID template containing IPv6 base data record entries and source and destination VLAN IDs.
275	Switch, MAC ID, and VLAN ID template containing IPv6 base data record entries, along with source and destination MAC addresses and source and destination VLAN IDs.
276	Switch and Layer 4 port template containing IPv6 base data record entries, along with source and destination Layer 4 ports.

Table 33-8 NetFlow Version 9 Templates (continued)

277	Switch, Layer 4 port, and MAC ID template containing IPv6 base data record entries, along with source and destination layer 4 ports and source and destination MAC addresses.
278	Switch, Layer 4 port, and VLAN ID template containing IPv6 base data record entries, along with source and destination Layer 4 ports and source and destination VLAN IDs.
279	Switch, Layer 4 port , MAC ID, and VLAN ID template containing IPv6 base data record entries, along with source and destination Layer 4 port, source and destination MAC addresses and source and destination VLAN IDs.
280	Switch and IPv6 route ID template containing IPv6 base data record entries, along with the route next hop.
281	Switch, IPv6 route ID, and MAC ID template containing IPv6 base data record entries, along with the route next hop and source and destination MAC addresses.
282	Switch, IPv6 route ID, and VLAN ID template containing IPv6 base data record entries, along with the route next hop, and source and destination VLAN IDs.
283	Switch, IPv6 next hop, MAC ID, and VLAN ID template containing IPv6 base data record entries, along with the route next hop, source and destination MAC addresses, and source and destination VLAN IDs.
284	Switch, IPv6 route ID, and Layer 4 port template containing IPv6 base data record entries, along with the route next hop, and source and destination Layer 4 ports.
285	Switch, IPv6 route ID, Layer 4 port and MAC ID template containing IPv6 base data record entries, along with the route next hop, source and destination Layer 4 port, and source and destination MAC addresses.
286	Switch, IPv6 next hop, Layer 4 port and VLAN ID template containing IPv6 base data record entries, along with the route next hop, source and destination Layer 4 ports, and source and destination VLAN IDs.
287	Switch, IPv6 next hop, Layer 4 port, MAC ID, and VLAN ID template containing IPv6 base data record entries, along with the IPv6 next hop, source and destination Layer 4 ports, source and destination MAC addresses, and source and destination VLAN IDs.

34

Connectivity Fault Management Configuration

This chapter provides information about configuring and monitoring the Connectivity Fault Management (CFM) protocol on S-Series devices.

How to Use Connectivity Fault Management in Your Network

The Connectivity Fault Management (CFM) process, as defined in the IEEE 802.1Q-2011 standard, provides network operators the means to monitor and troubleshoot services that may span multiple domain Ethernet networks. It provides a set of diagnostics and monitoring functions at the service provider level, allowing operators:

- To determine the relative health of an end-to-end network service and operational status
- To identify faults or mis-configurations within a network
- To take administrative action to correct those issues

CFM allows a customer to validate the end-to-end Ethernet service within a single domain network, a large network that is segmented into separate domains within the same organization, or a network that has contracted with a service provider. In the event of a degradation of that service, CFM provides diagnostic data that can locate the problem within a single organization or that can be forwarded to the service provider. Whether working with an outside entity or within a single organization, CFM can:

- Determine that the end-to-end service is properly configured
- Validate that all network nodes intended to be attached to the end-to-end service are reachable using that service

- Help pinpoint where a loss in connectivity may be located

The Extreme Networks S-Series CFM implementation supports the monitoring of a VLAN service, and the association of one or more VLANs with the primary service. For the remaining discussion in this chapter, any reference to a CFM monitored service is a reference to a CFM monitored VLAN service. A CFM service is monitored by the periodic sending of continuity check messages (CCM) across the monitored service. A CCM is a multicast message, confined to a single operator domain that provides a means to detect connectivity failures or configuration errors for that monitored service. These messages are unidirectional and do not solicit a response. Each end of a CFM monitored service transmits a periodic multicast CCM inward towards the other end of the service. The primary service can both send and receive CCMs. Services associated with the primary can only receive CCMs.

A CFM monitored service resides within three hierarchial layers of CFM configuration:

- Maintenance Domain (MD) – A logical container for all the equipment associated with the CFM monitored service and owned by a single network operator.
- Maintenance Association (MA) – A logical container for a specific CFM monitored service.
- Maintenance Point (MP) – A demarcation point on a port that implements the CFM functions within an MA. There are two types of MPs: Maintenance End-Points (MEP) and Maintenance Intermediate-Points (MIP)
  - MEPs – Ports, belonging to an MA, through which data enters and exits the portion of the network monitored by the CFM service.
  - MIPs – Auto-created MPs on ports that reside along the path between MEPs. The MIP supplements the function to the MEPs of the domain by passively snooping the CCMs that pass through them.

Multiple services can be associated with the primary service by configuring a VLAN table for the primary service.

The CFM loopback protocol provides for connectivity verification by sending loopback messages between the initiating device and a MEP or a MAC address. The CFM linktrace protocol provides for path verification and helps identify where in the path a connectivity problem is located. Linktrace messages can be sent between the initiating device and a MEP or a MAC address. CFM can notify the network operator by Syslog or SNMP traps when connectivity failures or configuration errors are detected.

The minimum generated alarm defect can be set for both Syslogs and traps. See "MEP Defect Definitions" on page 34-21 for reported defect details. Logging can be filtered by MD, MA, or MEP.

CFM configuration takes place within an hierarchy of configuration modes or contexts (see "CFM Configuration Modes" on page 34-8):

- The system global configuration mode
- The global system and CFM service specific default modes
- The three configuration layers in which a CFM service resides: MD, MA, and MEP
- The MA component configuration context which is a subset of the MA configuration

Connectivity Fault Management Overview

Maintenance Domain (MD)

A maintenance domain (MD) is the highest configuration context in which a CFM service resides. The MD is a collection of network devices, typically owned and operated by a single organization. Management of devices within a domain falls under the control of that single organization. Domains must be contiguous, such that all the devices belonging to a domain have uninterrupted network connectivity with each other. Domains may be nested or adjacent, but can not share network devices with other domains. Domains are intended to provide connectivity to systems outside of the domain. All devices associated with a monitored service must belong to the MD. The monitored service is configured within the MD. MDs are uniquely identified by an MD Name.

An MD is assigned a level relative to other MDs a monitored service passes through. The monitored service must belong to an MD with a higher level than all other MDs that the monitored service passes through. The CCM will not be allowed to enter an MD with a higher level than the MD that initiated the CCM.

Figure 34-1 displays how each administratively controlled segment of the network is configured within its own MD in a service provider context made up of:

- A customer equipment maintenance domain **mdCE1**
- A service provider maintenance domain **mdSP1**
- Two operator maintenance domains **mdOp1** and **mdOP2**

Figure 34-1 Maintenance Domain Overview

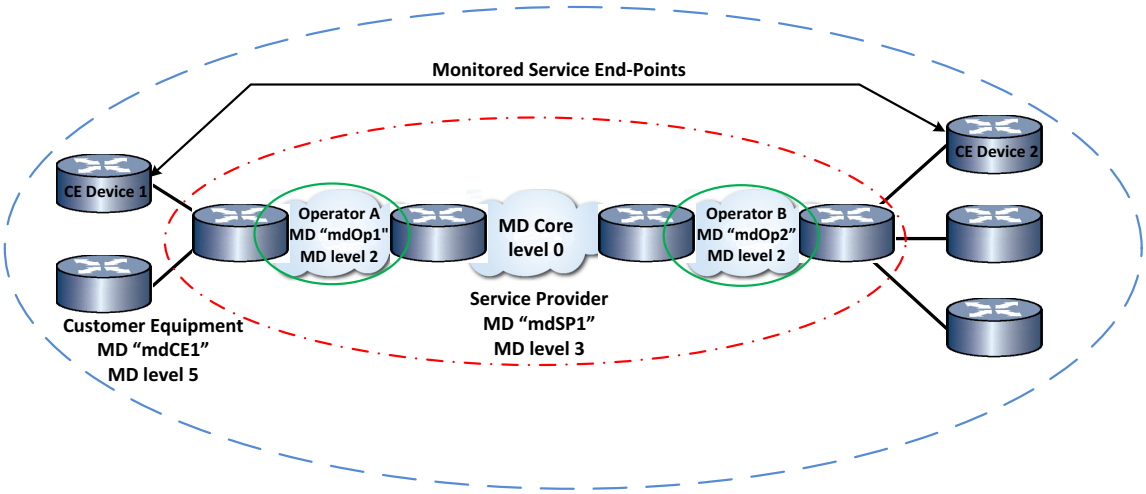


Figure 34-1 displays a typical MD domain configuration in a service provider context. The end-points of the monitored service (CE Device 1 and CE Device 2) belong to the customer equipment domain **mdCE1**. The monitored service is a VLAN these end-points belong to. The MD to which the monitored service devices belong must encapsulate any MDs the monitored service passes through from one end-point of the service to the other end-point. This encapsulation is

accomplished by the MD level. The customer equipment MD is assigned the highest MD level in our example, assuring that CCMs will be allowed to pass between the monitored service end-points. The encapsulated MDs can belong to such entities as service providers and network operators as shown here, or they can be segmented parts of your own network.

In our example, the monitored service belongs to MD **mdCE1** and passes through **mdSP1**, **mdOp1**, and **mdOp2** before reaching end-point CE Device 2 which belongs to MD **mdCE1**.

An MD constrains CFM traffic flows. CFM traffic flows of the MD owner and can transparently flow through any encapsulated MD with a lower MD level. mdCE1 CFM traffic flows pass through all the displayed MDs. mdSP1 traffic flows pass through mdOp1 and mdOp2, but are prevented from transiting into mdCE1. mdOP1 and mdOP2 CFM traffic is restricted to its own MD. This containment of traffic flows prevents unintended information flow between MDs.

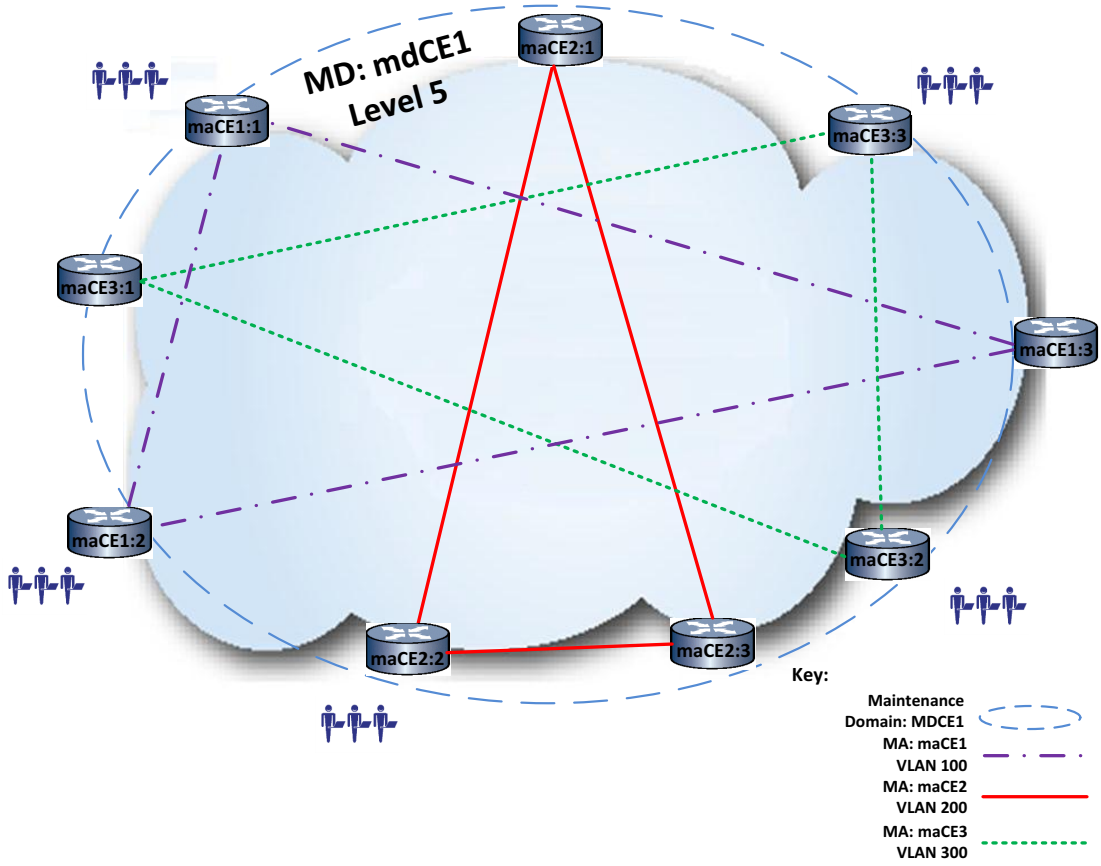
Maintenance Association (MA)

A maintenance association (MA) uniquely identifies a service within an MD. A service may be defined by an individual primary VLAN, and optionally by one or more VLANs associated with the primary VLAN, using a VLAN table configuration. There may be multiple MAs within a domain. Subsets of devices residing within the domain are collectively configured to form these associations. The devices belonging to a particular association will communicate among each other to implement the various features provided by CFM.

Figure 34-2 displays three maintenance associations configured for the customer equipment level 5 MD **mdCE1**. In this presentation, MD levels 0, 2, and 3 as shown in Figure 34-1 on page 34-3 are abstracted in the cloud. The presence of the non-customer equipment MDs in the cloud is transparent to the customer equipment MAs. Each network device on the edge of the cloud represents a CFM networking device configured for the appropriate MA that monitors the primary VLAN the device is on:

- VLAN 100 – MA: **maCE1**
- VLAN 200 – MA: **maCE2**
- VLAN 300 – MA: **maCE3**

Figure 34-2 Maintenance Association Overview



For each monitored VLAN, each network device depicted resides at the edge of the MD for its physical location; for example it may be on a separate university or enterprise campus from the other nodes of the monitored service. Data from these nodes may or may not have to transit an ISP or other operator domains outside of the customer equipment administrator's control, before reaching the other end of the monitored VLAN.

Within MA configuration you specify the monitored service (VLAN), create a list of all MEPs that belong to the MA and optionally change the interval between the sending of CCMs.

Maintenance Point (MP)

A maintenance point (MP) is a demarcation point on a port that implements the CFM functions within an MA. MPs serve as points of contact allowing communication with other devices within the MA. MPs also operate as filters to confine CFM Ethernet frames within the bounds of a domain by dropping frames that do not belong to the correct level. There are two types of MPs: maintenance end-points (MEPs) and maintenance intermediate points (MIPs).

Maintenance End-Point (MEP)

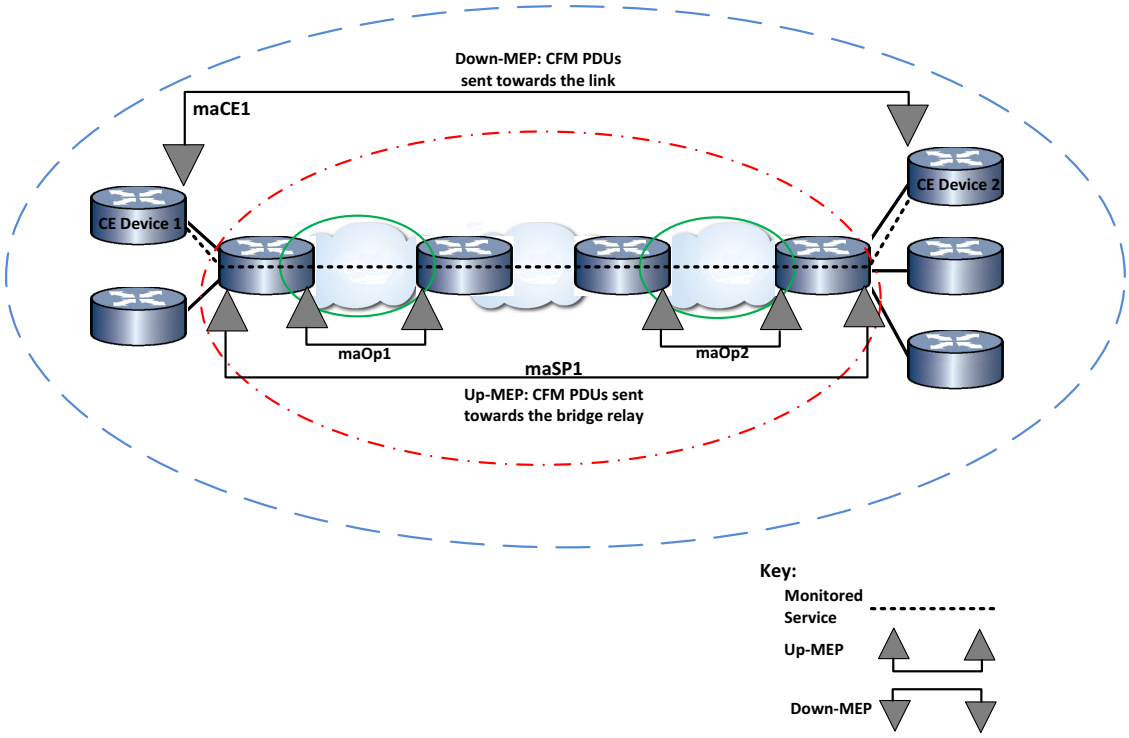
The Maintenance End-Point (MEP) serves as the logical boundary between devices that belong to different maintenance domains. A MEP resides at the edge of an MD. A MEP is associated with a single MA that monitors a primary VLAN service and any services associated with the primary service configured in an enabled VLAN table. The MEP must belong to the primary service associated with the MA or a service associated with the primary service. A MEP has a direction of either Up or Down. A Down-MEP sends CFM PDUs towards and receives CFM PDUs from the

link. In our overview example, The MEPs associated with the customer equipment monitored service configure their MEPs as Down-MEPs. An Up-MEP sends PDUs towards the bridge relay and receives PDUs from the bridge relay. Up-MEPs communicate through the bridge relay with all other ports attached to the protective service within that bridge. All non-customer equipment MEPs are configured as Up-MEPs for a customer equipment monitored service.

Figure 34-3 on page 34-6 displays MEP location and direction for our overview example. Customer equipment devices CE Device 1 and CE Device 2 belong to MA **maCE1** and each are configured with a Down-MEP facing the service provider MD configured as MA **maSP1**. The customer equipment administrator wants to monitor the service between CE Device 1 and CE Device 2. The customer equipment Down-MEPs send CFM PDUs towards the link and therefore towards the remote Down-MEP belonging to **maCE1** at the other end of the service. All Up-MEPs, belonging to other domains configured between the initiating and remote customer equipment down-MEPs, will transparently forward the CFM PDUs to the remote customer equipment MEP.

If a MEP fails to receive CFM PDUs in a timely fashion, CFM reports the failure.

Figure 34-3 Maintenance End-Point Overview



Within MEP configuration mode you must specify the port that the MEP is on and its direction. The MEP direction defaults to down.

Maintenance Intermediate Point (MIP)

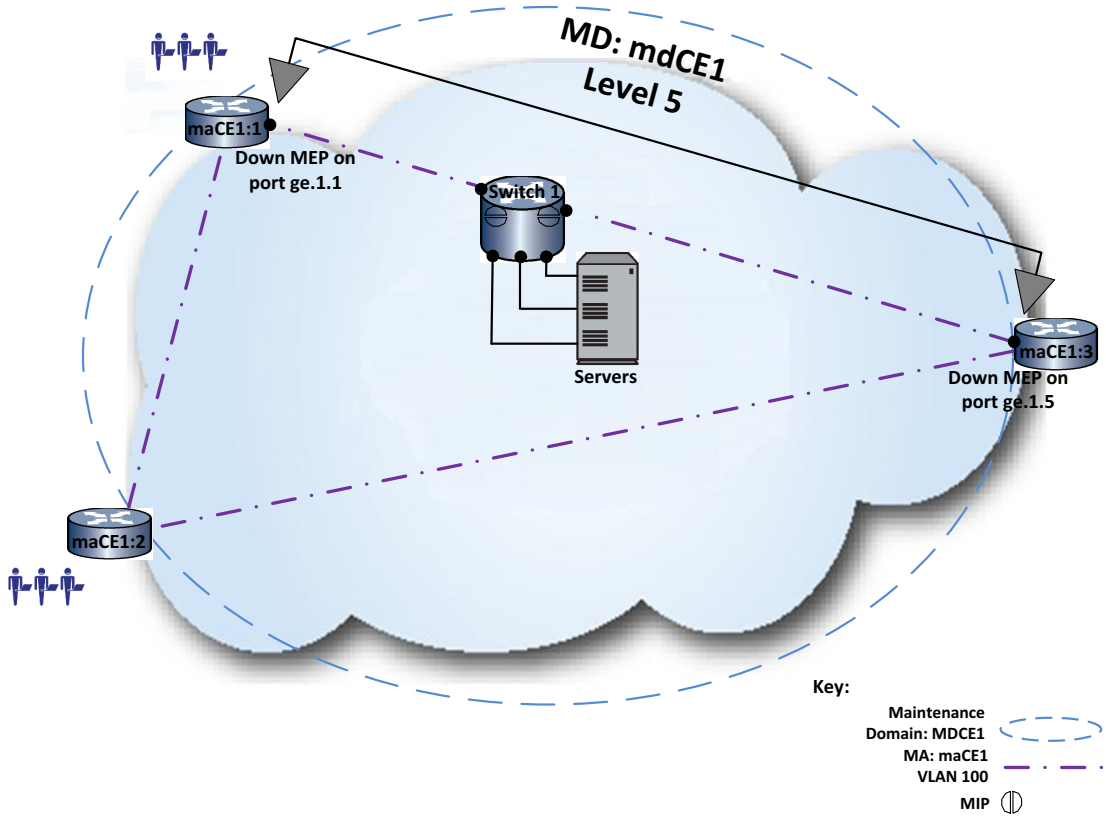
A Maintenance Intermediate Point (MIP) resides in the interior of an MD. MIPs are created on ports that reside along the path between MEPs. The MIP supplements the function to the MEPs of the domain. MIPs passively collect information by snooping the CCMs that pass through them. The information is collected in a database. These MIP databases act as highway “mile-markers” along the continuity check message path. MIPs may respond to loopback and linktrace requests received from MEPs in its MD. Use the loopback protocol to determine that a problem exists. Use the linktrace protocol to determine which MIP can identify the problem.

MIPs are comprised of two Maintenance Half Functions (MHFs), one with an “up” direction, and one with a “down” direction. The up direction half function points towards the bridge relay function. The down direction half function points towards the link. Use the **show cfm stack-table** command to access MIP direction port information. MIPs for a given service can be configured in any domain but must be configured with the same MD level as the MD of the MEPs sending the continuity check messages. MIPs can not be created on devices containing down-MEPs. MIPs can be created on devices containing Up-MEPs or no MEPs. MIPs can be created on down-MEPs using MD default configuration.

You do not administratively configure MIPs. MIPs are automatically created if MHF creation is enabled. If MHF creation is not enabled, MIPs are not created.

Figure 34-4 on page 34-7 presents a typical situation in which you would want to turn on MIPs. Switch 1 resides in the interior of the MD on a VLAN monitored by MA maCE1. Down-MEPs are configured on port ge.1.1 of maCE1:1 and port ge.1.5 of device maCE1:3. The MD default setting for MHF creation has been enabled on Switch 1. The ports on Switch 1 used by the service are being passively monitored by MIPs. If a problem is reported on one of the servers connected to Switch 1, from a CFM enabled device, use the loopback protocol specifying the MAC address of the Switch 1 port to verify connectivity with the server connected port. Use the linkstate protocol specifying the MAC address of the Switch 1 port to verify that a path exists between the initiating device and the Switch port. Because the server connected ports are on the edge of the network, you may want to put down MEPs on those ports.

Figure 34-4 Maintenance Intermediate-Point Overview



CFM Configuration Modes

Table 34-1 lists the seven modes in which CFM configuration takes place.

Table 34-1 CFM Configuration Modes

Configuration Mode	Description
Global System configuration	System level CFM configuration and access to CFM default and MD configuration modes takes place in global system configuration mode. See Table 34-5 on page 34-25 for a listing of CFM global system configuration commands.
Default Maintenance Domain configuration	Default maintenance domain is the CFM configuration mode with the lowest precedence. If no other configuration levels are administratively configured for a given global CFM parameter, the system default maintenance domain configured value for that parameter is used. Throughout this document system default maintenance domain configuration mode is referred to as system default MD. See Table 34-6 on page 34-26 for a listing of system default MD configuration commands.
Default maintenance domain VLAN	Default maintenance domain VLAN configuration mode has a higher precedence than system default MD configuration but is lower than all other CFM modes that contain the same parameters. See Table 34-6 on page 34-26 for a listing of Default maintenance domain VLAN configuration commands.
Maintenance Domain (MD) configuration	Provides access to the MD configuration commands and MA configuration mode. Through out this document maintenance domain configuration mode is referred to as MD configuration mode. MD configuration has a higher precedence than either of the default configuration modes. See Procedure 34-1 on page 34-26 for a listing of maintenance domain configuration commands.
Maintenance Association (MA) configuration	Provides access to the MA configuration commands and MA-Comp and MEP configuration modes. Through out this document maintenance association configuration mode is referred to as MA configuration mode. See Procedure 34-2 on page 34-27 for a listing of maintenance association configuration commands.
Maintenance Association Component (MA-Comp) configuration	Provides access to MA component configuration commands. MA component configuration is a subset of MA configuration. Through out this document maintenance association component configuration mode is referred to as MA-Comp configuration mode. MA-Comp configuration has a higher precedence than either of the default modes and the MD configuration mode. See Procedure 34-3 on page 34-27 for a listing of MA component configuration commands.
Maintenance End-Point (MEP) configuration	Provides access to MEP configuration commands. Through out this document maintenance end-point configuration mode is referred to as MEP configuration mode. See Procedure 34-4 on page 34-28 for a listing of maintenance end-point configuration commands.

Implementing Connectivity Fault Management

To Implement CFM on an Extreme Networks S-Series platform:

- Globally enable CFM on the device.
- Optionally modify system MD and MD VLAN defaults
- Optionally set the logging filter for Syslog messages

- Optionally configure VLAN tables that associate one or more CFM services with a primary service
- Configure the MDs for your system
- Configure an MA for each CFM monitored service
- Configure the MEP-list for each MA
- Configure the MEPs associated with each MA

Configuring CFM at the Global System Level

Within the global configuration command mode, you:

- Globally enable CFM on the device
- Access the default modes for configuring global MD default and MD default VLAN service parameters (see "CFM Configuration Modes" on page 34-8)
- Access the MD configuration mode (see "CFM Configuration Modes" on page 34-8)
- Configure Syslog message filtering

CFM must be enabled globally for CFM to be operational. CFM is disabled by default. Use the **cfm enable** command in global configuration mode to globally enable CFM on the device.

CFM Logging Filtering

By default, all CFM MD, MA, and MEP Syslog messages are sent. CFM logging filtering allows you to limit the sending of logged messages by a specified MD, MA or MEP. If an MA or MEP is not specified, Syslog messages are sent for all MAs and MEPs for the specified MD.

Use the **cfm logging filter** command to filter the sending of CFM Syslog messages to a specified MD, MA and MEP.

This example shows how to configure Syslog to display Syslog messages for all MEPs in the **myMA1** maintenance association of the **myMD1** maintenance domain:

```
S Chassis(rw-config)->cfm logging filter md string-name myMD1 ma string-name myMA1
S Chassis(rw-config)->
```

VLAN Table Configuration

By default there is a one-to-one relationship between a service ID and the CFM monitored service. If the service ID is 10 and the monitored CFM service is a VLAN, the monitored service is VLAN 10. In a one-to-one CFM service, the only monitored service is the primary service or selector. There are no other services associated with the primary service. The primary service can both receive and send CFM PDUs when the CFM device is a MEP. The primary service can receive CFM PDUs when the CFM device is a MIP. To fully monitor multiple services on the MEP would require that you create a separate service (MA) for each of the VLANs associated with a MEP adding considerable administrative overhead. When the primary goal is to fully monitor a single CFM service and, at the same time, monitor the reception of CFM PDUs on a group of associated services within a single MA, the VLAN table provides a significant reduction in administrative overhead.

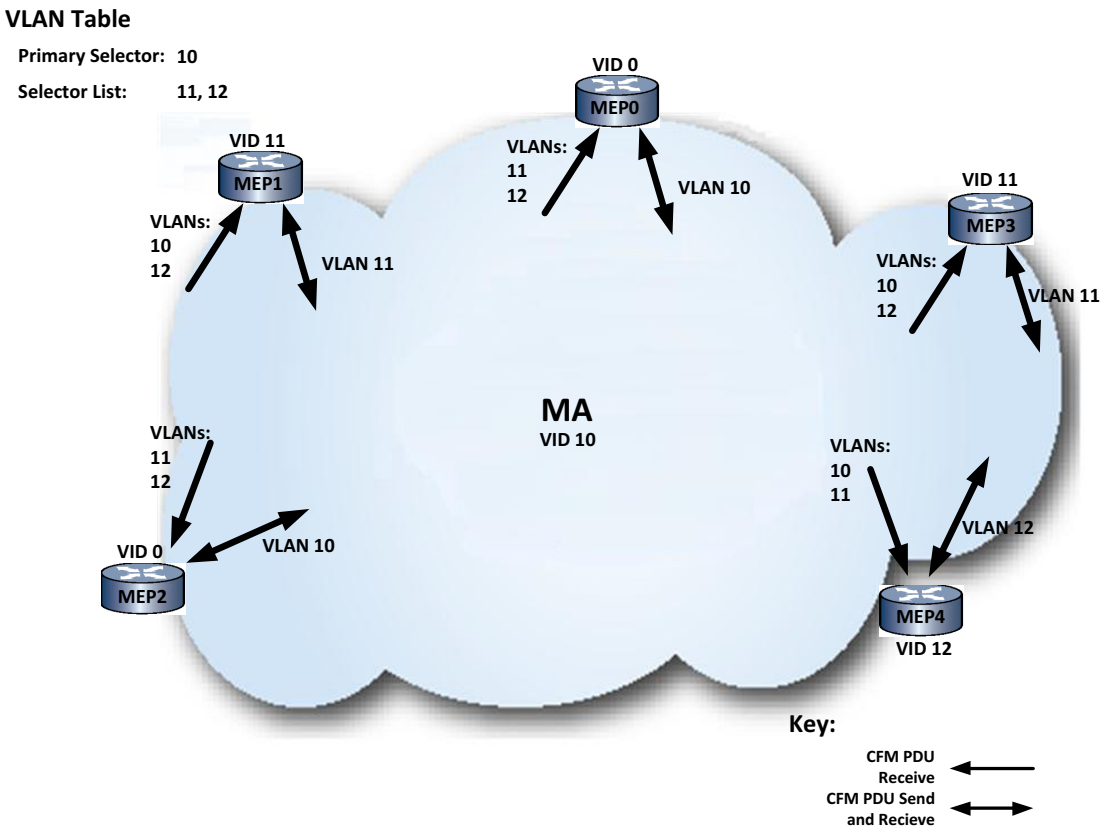
The CFM VLAN table allows you to associate one or more services or selectors with the primary service. These services associated with the primary are capable of receiving CFM PDUs on any service configured in the VLAN table to which the monitored service belongs. For example: if the

VLAN table for primary selector 10 has a configured selector list containing selectors 11 - 14, and the monitored service is a VLAN, MEPs with any primary VLAN of 10 through 14 will:

- Both receive and send CFM PDUs on the MEP's primary VLAN (the configured VID or the primary service configured for the MA if the MEP VID is set to **0**)
- Receive CFM PDUs on any other VLAN table member configured on the device

Figure 34-5 presents a VLAN table overview for an MA configured with a VLAN service and a VID of 10. The VLAN table is configured with a primary selector of 10 and associated selector list of 11 and 12. MEP0 and MEP2 are configured with the default VID 0 and inherit the MA VID setting 10. For these two MEPs, CFM PDUs are sent and received on VLAN 10 and received on VLANs 11 and 12. MEP1 and MEP3 are configured for VID 11. For these two MEPs, CFM PDUs are sent and received on VLAN 11 and received on VLANs 10 and 12. MEP4 is configured with VID 12. For MEP4, CFM PDUs are sent and received on VLAN 12 and received on VLANs 10 and 11.

Figure 34-5 VLAN Table Configuration Overview



The primary selector defines the ID of the service being modified. When configuring a CFM service, the configured VID for the new service is the primary selector. If the VID is configured for the default of **0**, the MA the device belongs to provides the primary selector. If the device primary selector is a member of the VLAN table, all other members of the table are associated with the device primary selector.

When configuring the VLAN table, the primary selector is specified. One or more service IDs are specified in the selector list. Specifying the **enable** command option activates the association between the IDs in the selector list and the primary service. Specifying the **disable** command

option disables the association of the IDs in the selector list with the primary service. The VLAN table remains configured, but is not active.

Maintenance points (MIP or MEP) associated with a CFM service will be able to receive CFM PDUs on any of the active IDs defined in the VLAN table selector list. MEPs may be configured with a Primary VID, which must belong to the enabled VLAN table, otherwise there is a one-to-one relationship between the VID and the service. With a primary VID defined, the MEP can transmit tagged PDUs using that primary service. If no primary VID is defined by the maintenance point, the VID, as inherited from the MA, will be used by the maintenance point to tag its transmitted PDUs.

Use the cfm vlan-table command in global configuration mode to configure a VLAN table.

Activating CFM Configuration

Changes made in the named MD, MA, MA component, and MEP configuration contexts do not take affect until the the CFM configuration is activated for that context. Use the **enable** command in the appropriate configuration context to activate the CFM configuration for that context.

If **enable** has already been entered in the current context, you must first enter **no enable** before making any further changes. If you attempt to make changes in a context that has already been enabled, you receive an error message like the following MD context error message:

```
Error: MD must be disabled ("no enable") before changes can be made.
```

Once changes are completed, enter **enable** again for the changes to take affect.

For a given configuration context to be operational, its parent context must be enabled:

- The MD context is the parent of MA and MA component contexts
- The MA context is the parent of the MA component context
- The MA and MA component contexts are the parents of the MEP context.

Configuring a Maintenance Domain (MD)

Refer to "Maintenance Domain (MD)" on page 34-3 for an MD overview discussion.

MD Configuration Modes

You access MD configuration from global configuration mode. The are two CFM MD default configuration modes and a named MD configuration mode.

Table 34-2 MD Configuration Modes

Mode	Description
Global MD and VLAN MD Service Default Configuration Mode	Global MD default configuration values are used when global default parameters have not been administratively modified in higher precedence configuration modes. Global MD default configuration mode is the lowest precedence configuration mode. VLAN MD service default configuration values are applied to the specified VLAN service. CFM monitors a network service. CFM currently supports VLAN monitoring. Any MD default parameters configured in MD VLAN service default mode take precedence over global MD default mode configuration for the specified VLAN service. In global MD and VLAN service default modes you can configure: ID permission – See "Setting SenderID TLV Permission" on page 34-13 MIP creation – See "Enabling Maintenance Intermediate-Points (MIP)" on page 34-14 MD level – See "Setting the MD Level" on page 34-14 Use the **cfm default-md default** command to enter global MD default configuration mode. Use the **cfm default-md vid** command, specifying the VLAN service to be monitored, to enter MD VLAN service default configuration mode.
Named MD Configuration Mode	Configuration entered in a named MD configuration mode takes precedence over global MD default configuration and MD VLAN service defaults. Use the **cfm md** command, using one of four MD naming conventions, specify an MD name to enter named MD configuration mode. See "MD Naming Conventions" on page 34-13 for MD naming options.

This example shows how to enter the system level default MD configuration command mode. Note that the command prompt changes to indicate that you have moved to Default Maintenance Domain mode for configuring default maintenance domain values.

```
S Chassis(rw-config)->cfm default-md default
S Chassis(su-config-cfm-default-md-def)->
```

This example enters default configuration mode for VLAN 20. Note that the command prompt changes to indicate that you have moved to Default Maintenance Domain mode for configuring VLAN 20.

```
S Chassis(rw-config)->cfm default-md vid 20
S Chassis(su-config-cfm-default-md.20)->
```

This example shows how to enter configuration command mode for the **myMD1** maintenance domain. This maintenance domain instance is assigned index 1.

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(rw-config-cfm-md.1)->
```

MD Naming Conventions

When accessing the named MD configuration mode, use one of four naming conventions to identify a CFM MD:

- **String-name** – A string of up to 43 printable characters. This format provides descriptive freedom in naming the association.
- **Dns-like-name** – A string of up to 43 printable characters that represents a standard domain name server convention. This option is for management purposes. A check is done to assure that you have entered a legally formatted DNS name.
- **Mac-int-name** – A MAC address format plus an integer index ID value. You must follow a supported MAC address format as described in the parameter table.
- **No-name** – A no name option that sets a NULL string as the MD name. It can only be used for a single MD on each device.

A non-configurable index value is associated with every MD and MA. This index value appears in the prompt to provide context to the prompt. For example, the prompt (rw-config-cfm-md.1) is for the MD index 1 configuration context. Use the **show cfm md** command to display the index value for each configured MD.

Setting SenderID TLV Permission

You configure the ID permission setting for the content sent in the SenderID TLV by the maintenance points. Given that CFM PDUs can pass through domains owned by organizations to which you may not want topology information exposed, use the ID permission settings to configure an appropriate permission level for the monitored service. The value CFM will use depends upon the context and precedence of the values configured. ID permission can be configured in CFM command modes: Default MD, Default MD VLAN, MD, MA-Comp configuration mode. The modes here are listed in precedence from lowest to highest. Configuring ID permission in MA-Comp configuration mode takes precedence over any other CFM configuration mode for the current MA context.

Enabling ID permission includes in PDUs sent by a maintenance point (MIP or MEP) informational TLVs that identify the bridge by chassis, remote management or both.

You can specify:

- **Chassis** – Includes the chassis' MAC address in the TLV
- **Manage** – Includes the method of remote management in the TLV
- **Both** – Includes both chassis and manage information in the TLV
- **None** – No SenderID TLV is sent
- **Defer** – In a VLAN default or MA component context, the SenderID TLV behavior defers to the next highest configuration level settings.

The **defer** option is only valid in the Default Maintenance Domain VLAN service context, the (named) Maintenance Domain context, or the Maintenance Association Component context. When **defer** is used in the:

- Default MD VLAN service context, the **id-permission** value defaults to the value set for the default MD
- MA component context, the **id-permission** value defaults to the value set for the (named) Maintenance Domain within which the MA component resides

Use the **id-permission** command in the appropriate mode to specify the informational content to include in the SenderID TLV. ID permission defaults to no SenderID TLVs being sent.

This example shows how to configure the maintenance points to defer to the MD setting for ID permission for the **myMA1** maintenance association component context:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->ma-comp
S Chassis(su-config-cfm-macomp)->id-permission defer
S Chassis(su-config-cfm-macomp)->enable
S Chassis(su-config-cfm-macomp)->
```

Enabling Maintenance Intermediate-Points (MIP)

If MIP creation is enabled, MIPs are auto-created on all ports assigned to the CFM service. See "Maintenance Intermediate Point (MIP)" on page 34-6 for a description of the role MIPs play in a CFM process.

MIP creation can be configured in CFM command modes: Default MD, Default MD VLAN, MD, MA-Comp configuration mode. The modes here are listed in precedence from lowest to highest. Configuring MIP creation in MA-Comp configuration mode takes precedence over any other CFM configuration mode for the current MA context.

You can specify:

- **Default** – MIPs are created for the MD or MA context
- **Explicit** – MIPs are created for the MD or MA context, only if a MEP exists at the next lower MD level
- **None** – MIPs are not created in the MD or MA context
- **Defer** – In a VLAN default or MA component context, the MIP creation defers to the next highest configuration level settings. The **defer** option is only valid in the Default Maintenance Domain VLAN service context, the (named) Maintenance Domain context, or the Maintenance Association Component context. When **defer** is used in the:
  - Default MD VLAN service context, MIP creation defaults to the value set for the default MD
  - MA component context, MIP creation defaults to the value set for the (named) Maintenance Domain within which the MA component resides

Use the **mhf-creation** command in the appropriate configuration mode to set whether the creation of maintenance intermediate-point half function (MHF) is allowed for the current context. MIPs are not enabled by default.

This example shows how to set the system default maintenance domain value to allow an MHF to be created only when a MEP in the next lowest MD exists:

```
S Chassis(rw-config)->cfm default-md default
S Chassis(su-config-cfm-default-md-def)->mhf-creation explicit
S Chassis(su-config-cfm-default-md-def)->
```

Setting the MD Level

The Ethernet CFM service network is partitioned into maintenance levels. Each maintenance level is defined by the reach and scope of the organization which administers the network equipment. Higher maintenance levels exist at the edge of the network. Network customers typically own these higher levels. Lower maintenance levels typically reside closer to the network core, and are

usually reserved for service providers or network operators. Maintenance levels are hierarchical in nature. Higher maintenance levels encapsulate lower maintenance levels.

CFM PDUs at a given level are distributed and processed among maintenance points within that specific domain. CFM PDUs from a domain that encapsulates a lower level domain, pass transparently through the enclosed lower level domain as they transit to the remote side of the domain they belong to.

Note: See Figure 34-1 on page 34-3 for a depiction of MD encapsulation. The customer equipment MD at level 5 encapsulates both the service provider (level 3) and network operator (level 2) MDs; the service provider MD at level 3 encapsulates the network operator MDs.

CFM PDUs from a lower level domain encapsulated by a higher level domain are restricted from exiting the domain. These CFM PDUs are filtered by maintenance points of the higher level encapsulating domain.

These rules for transport and containment ensure that domain management and administration is restricted to the responsible organization. It also ensures that information concerning either the enclosed or enclosing domain is restricted to that domain. This prevents the customer from learning details concerning the internal topology of the service provider and operators and service providers from learning details concerning the customer topology.

The maintenance level to be assigned to an MD is typically determined by the service provider in the service level agreement.

Use the **level** command in the default MD or MD configuration mode to specify the MD level.

This example shows how to configure the level to 5 for the **myMD1** maintenance domain:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->level 5
S Chassis(su-config-cfm-md.1)->enable
S Chassis(su-config-cfm-md.1)->
```

Changing the Maintenance Domain Name

The name of an MD can be changed from within the named MD context. Use the **name** command specifying a supported MD name type and name. See "MD Naming Conventions" on page 34-13 for details on naming an MD.

This example shows how to change the name of the **myMD1** maintenance domain to **yourMD1**. Note that the index number for this maintenance domain does not change.

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->name md string-name yourMD1
S Chassis(su-config-cfm-md.1)->enable
```

Configuring a Maintenance Association (MA)

Refer to "Maintenance Association (MA)" on page 34-4 for an MA overview discussion.

For information about...	Refer to page...
Accessing MA Configuration Mode	34-16
Enabling the Maintenance Association Configuration	34-16
Changing the Maintenance Association Name	34-16

Accessing MA Configuration Mode

You access MA configuration from the MD configuration mode specifying a name for the MA. One of three naming conventions can be used to identify a CFM MA:

- **String-name** – A string of up to 45 printable characters. This format provides descriptive freedom in naming the association.
- **Vid-name** – An integer value between 0 - 4094. This format restricts the association name to the VLAN range. For management purposes, show command output will label this format as a VLAN type. Use this format when the association is directly related to fault managing a VLAN.
- **Id-name** – An integer value between 0 - 65535. This format restricts the association name to an integer range. Use this format when a sequential naming scheme is being used to manage the associations.

Use the **ma** command in MD configuration command mode to enter MA configuration mode for the named MA.

This example shows how to enter configuration command mode for the **myMD1** maintenance domain and enter MA configuration mode for the maintenance association named **myMA1**:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(rw-config-cfm-md.1)->ma string-name myMA1
S Chassis(rw-config-cfm-ma.1)->
```

Enabling the Maintenance Association Configuration

Changes made in the named MA contexts do not take affect until the the CFM MA configuration is activated, using the **enable** command. See “Activating CFM Configuration” on page 34-11 for CFM configuration activation details.

This example shows how to activate **myMA1** for the **myMD1** maintenance domain:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->enable
S Chassis(su-config-cfm-ma.1)->
```

Changing the Maintenance Association Name

The name of an MA can be changed from within the named MA context. Use the **name** command specifying a supported MA naming convention and name. See the MA naming convention information in “Accessing MA Configuration Mode” on page 34-16 for details on naming an MA.

This example shows how to change the MA name for the **myMA1** maintenance association to **yourMA1**:

```
S Chassis(rw-config)->cfm md string-name myMD1
```

```
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->name string-name yourMA1
S Chassis(su-config-cfm-ma.1)->enable
S Chassis(su-config-cfm-ma.1)->
```

Setting the Continuity Check Message (CCM) Interval

The interval between CCMs can be set to 1 second, 10 seconds, 1 minute, or 10 minutes.

All maintenance end points (MEPs) in an association must be configured for the same CCM interval. The source MEP sends a continuity check message at the interval set by this command. Should the remote end-point not be configured for the same CCM interval as the source end-point, CFM logs a configuration error and potentially triggers a defect.

If the remote end-point does not receive the continuity check message within a period of 3.5 times the configured CCM interval, an error is logged. Intermediate-points (MIPs) on the path between the sending and receiving MEPs do not actively log errors.

Use the **ccm-interval** command in MA configuration mode to change the interval between the transmission of CCMs. The CCM interval defaults to 1 second.

This example sets the interval between CCM transmissions used by all MEPs in the **myMA1** maintenance association to **10** seconds:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->ccm-interval 10sec
S Chassis(su-config-cfm-ma.1)->enable
S Chassis(su-config-cfm-ma.1)->
```

Configuring the Maintenance Association MEP List

The MA must have knowledge of the local and remote MEP IDs for the local end-points to recognize the remote end-points. All MEPs in an association must be listed in the association MEP list and the MEP list must be enabled for MEPs to be operational. MEPs are specified as a list of MEP IDs each separated by a comma (“**,**”) or, if a range of IDs, by a dash (“**-**”).

Use the **mep-list** command, in MA configuration mode, to specify the MEPs that are or will be present in the MA. The MEP list is disabled by default. Use the **enable** option to enable the MEP list. Use the **disable** option to take a remote MEP out of defect consideration. You might want to disable a remote MEP during pre-provisioning a network, during the period in which all end-points have not yet been set up.

This example shows how to create and enable the MEP list for MEPs **30** through **35**, **1000**, and **1005** on the **myMA1** maintenance association:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->mep-list 30-35,1000,1005 enable
S Chassis(su-config-cfm-ma.1)->enable
S Chassis(su-config-cfm-ma.1)->
```

Configuring the Maintenance Association Components

MA component configuration makes up a subset of the MA configuration within its own configuration mode. You access MA component configuration from MA configuration mode using the **ma-comp** command.

Enter the MA component configuration mode to:

- Configure ID permission for the MA context (see "Setting SenderID TLV Permission" on page 34-13)
- Configure MHF creation for the MA context (see "Enabling Maintenance Intermediate-Points (MIP)" on page 34-14)
- Configure the VLAN the MA is associated with (see Setting the Maintenance Association VLAN Service)

Setting the Maintenance Association VLAN Service

The service CFM monitors gets assigned to the MA. CFM supports the monitoring of a single VLAN service. If you wish to protect multiple VLANs in your system, create an MA for each VLAN to be protected. When CFM monitors a VLAN service, the monitored VLAN must be specified in the MA component configuration mode using the **vid** command. The default VLAN value for an MA is **0** (no VLAN).

This example sets VLAN **1000** as the configured VLAN for the **myMA1** component configuration:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->ma-comp
S Chassis(su-config-cfm-macomp)->vid 1000
S Chassis(su-config-cfm-macomp)->
```

Enabling the Maintenance Association Component Configuration

Changes made in the MA component context do not take affect until the MA component configuration is activated, using the **enable** command. See "Activating CFM Configuration" on page 34-11 for CFM configuration activation details.

This example shows how to activate **myMA1** component configuration:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->ma-comp
S Chassis(su-config-cfm-macomp)->enable
S Chassis(su-config-cfm-macomp)->
```

Configuring a Maintenance End-Point (MEP)

Refer to "Maintenance Point (MP)" on page 34-5 for a Maintenance Point overview discussion that includes the MEP and MIP.

For information about...	Refer to page...
Accessing MEP Configuration Mode	34-19
Configuring the MEP Bridge Port	34-19

Accessing MEP Configuration Mode

You access MEP configuration mode from the MA configuration mode using the **mep** command. When accessing the MEP configuration mode, specify a MEP ID in the range **1** - **8191**.

This example enters configuration mode for the **myMA1** MEP **1000**:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->mep 1000
S Chassis(su-config-cfm-mep.1000)->enable
S Chassis(su-config-cfm-mep.1000)->
```

Configuring the MEP Bridge Port

The bridge port the MEP is on must be configured. Use the **port** command in MEP configuration mode to configure the MEP port.

This example shows how to set **tg.1.1** as the port the **myMA1** MEP 1000 is attached to:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->mep 1000
S Chassis(su-config-cfm-mep.1000)->port tg.1.1
S Chassis(su-config-cfm-mep.1000)->enable
```

Configuring the MEP VLAN

The MEP VLAN is the VLAN upon which the MEP transmits its PDUs. For the current CFM service implementation a MEP can only be associated with the single service VLAN, so there is no need to specify the MEP VLAN since it defaults to the service VLAN. If you do specify it, it must agree with the service VLAN. Use the **vid** command in MEP configuration mode to specify the MEP VLAN. VLANs can have a value from **0** - **4094**. The MEP VLAN defaults to **0** (no VLAN).

This example sets VLAN **1000** as the configured VLAN for the **myMA1** MEP **1000**:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->mep 1000
S Chassis(su-config-cfm-mep.1000)->vid 1000
S Chassis(su-config-cfm-mep.1000)->enable
```

```
S Chassis(su-config-cfm-mep.1000)->
```

Configuring MEP Direction

The direction a MEP faces is relative to the link and the bridge relay. A down-MEP sends CFM Ethernet frames towards and receives CFM Ethernet frames from the link. An up-MEP sends CFM Ethernet frames towards the bridge relay and receives CFM Ethernet frames from the bridge relay. The bridge relay sends and receives frames to the ports associated with the CFM service.

A down-MEP performs the following functions:

- Sends and receives CFM Ethernet frames at its MD level over the link connected to the port where the MEP is configured.
- Drops all CFM Ethernet frames at its MD level or lower that come from the bridge relay.
- Processes all CFM Ethernet frames at its MD level coming from the link.
- Drops all CFM Ethernet frames at a lower MD level coming from the link.
- Transparently forwards all CFM Ethernet frames at a higher MD level, regardless of whether they came from the link or the bridge relay.

An up-MEP performs the following functions:

- Transparently forwards all CFM Ethernet frames that have a higher MD level.
- Sends and receives CFM Ethernet frames with the same MD level through the bridge relay, but not over the link.
- Drops all CFM Ethernet frames at its MD level or lower that come from the link.
- Processes all CFM Ethernet frames at its MD level coming from the bridge relay.
- Drops all CFM Ethernet frames at a lower MD level coming from the bridge relay.

MEP direction defaults to down.

This example shows how to set the direction for the **myMA1** MEP **1000** to up:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->mep 1000
S Chassis(su-config-cfm-mep.1000)->direction up
S Chassis(su-config-cfm-mep.1000)->enable
S Chassis(su-config-cfm-mep.1000)->
```

Setting the Lowest Priority MEP Defect Alarm

You can set the lowest priority defect that will generate a fault alarm syslog message using the **alarm-defect-syslog** command in the MEP configuration mode.

You can set the lowest priority defect that will generate a fault alarm trap message using the **alarm-defect-trap** command in the MEP configuration mode.

The defects reported is based upon the specified level:

- **all-def** – Specifies that the DefRDICCM, DefMACstatus, DefRemoteCCM, DefErrorCCM, and DefXconCCM defects will generate a fault alarm syslog message. See Table 34-3 for defect descriptions.
- **mac-rem-err-xcom** – Specifies that the DefMACstatus, DefRemoteCCM, DefErrorCCM, and DefXconCCM defects will generate a fault alarm syslog message (default).

- **rem-err-xcon** – Specifies that the DefRemoteCCM, DefErrorCCM, and DefXconCCM defects will generate a fault alarm syslog message.
- **err-xcon** – Specifies that the DefErrorCCM, and DefXconCCM defects will generate a fault alarm syslog message.
- **xcon** – Specifies that the DefXconCCM defect will generate a fault alarm syslog message.
- **no-xcon** – Specifies that no fault alarm Syslog messages will be generated.

Table 34-3 MEP Defect Definitions

Defect	Description
DefRDICCM	One or more continuity check messages received by this MEP contained the RDI bit. This bit indicates that some other MEP in this MEP's MA is transmitting the RDI bit. This defect clears if continuity check messages from all remote MEPs do not have the RDI bit set. Devices set the RDI bit if they have received continuity check messages from a remote MEP that indicates the remote MEP has seen defect notifications from other MEPs in the MA.
DefMACstatus	The port status is not indicating "UP" for all remote MEPs on received continuity check messages, or the interface status for any remote MEP on received continuity check messages is not indicating "UP".
DefRemoteCCM	This MEP is not receiving continuity check messages from a MEP in its configured list.
DefErrorCCM	This MEP is receiving continuity check messages from a remote MEP that either uses an invalid MEP ID or uses a continuity check interval that does not match the receiving MEP.
DefXconCCM	This MEP is receiving continuity check messages from a remote MEP that either uses an MD level lower than the receiving MEP or uses a different MD or MA name than the receiving MEP.

You can set the minimum time a defect must be present before an alarm is generated using the **alarm-time** command. You can set the minimum amount of time a defect must be absent before and alarm is reset using the **reset-time** command. Alarm time and reset time values are set in centiseconds between **250** (2.5 seconds) and **1000** (10 seconds). The default value is **2.5** seconds.

This example sets the Syslog defect alarm setting for the **myMA1** MEP **1000** to generate a Syslog message if any supported defect is present for at least 3 seconds:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->mep 1000
S Chassis(su-config-cfm-mep.1000)->alarm-defect-syslog all-def
S Chassis(su-config-cfm-mep.1000)->alarm-time 300
S Chassis(su-config-cfm-mep.1000)->reset-time 300
S Chassis(su-config-cfm-mep.1000)->enable
```

Enabling MEP CCMs

By default a MEP does not send CCMs. Use the **cci-enabled** command in MEP configuration mode to enable the sending of CCMs for this MEP.

This example shows how to enable the sending of continuity check messages for MEP **1000** on **myMA1**:

```
S Chassis(rw-config)->cfm md string-name myMD1
```

```
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->mep 1000
S Chassis(su-config-cfm-mep.1000)->cci-enabled
S Chassis(su-config-cfm-mep.1000)->enable
S Chassis(su-config-cfm-mep.1000)->
```

Activating the MEP State Machine and the Remote MEP

The MEP state machine must be set to active for the MEP to be operational using the **active** command. If the MEP state machines are not active, the MEP will not operate.

If the state machine status is active and the MEP configuration is not enabled using the **enable** command, the MEP will have a Row Status of "Active" in the **show cfm md ma mep remote-mep** command, but will not perform any functions such as send and receive PDUs or generate defects.

The remote MEP(s) for this service must be enabled in the database for the CFM service to be operational between the two MEPs. Use the **remote-mep active** command to enable any remote MEPs associated with the local MEP. If you want to remove a remote MEP from defect consideration, for example, while provisioning the network, you can disable the remote MEP using the **no remote-mep active** command.

This example sets the administrative state for the **myMA1** MEP **1000** to active and enables remote MEP **2000** in the MEP database:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->mep 1000
S Chassis(su-config-cfm-mep.1000)->active
S Chassis(su-config-cfm-mep.1000)->remote-mep active 2000
S Chassis(su-config-cfm-mep.1000)->enable
S Chassis(su-config-cfm-mep.1000)->
```

Modifying the MEP CCM and Linktrace 802.1p Priority

You can modify the 802.1p priority used by connectivity check messages and messages sent by the linktrace protocol. 802.1p CCM and linktrace message priority is on a per MEP basis. Valid values are 0 - 7. The default value is 1. Use the **priority** command in MEP configuration mode to modify the 802.1p priority used by CCMs and linktrace messages.

The **priority** command does not affect loopback protocol message 802.1p priority. Loopback protocol message priority can be modified when entering the **loopback** command.

This example shows how to set set the continuity check message and linkstate message priority for MEP **1000** to **3**:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->mep 1000
S Chassis(su-config-cfm-mep.1000)->priority 3
S Chassis(su-config-cfm-mep.1000)->
```

Enabling the Maintenance End-point Configuration

Changes made in the MEP context do not take affect until the MEP configuration is activated, using the **enable** command. See "Activating CFM Configuration" on page 34-11 for CFM configuration activation details.

This example shows how to enable MEP **1000** on **myMA1**:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->mep 1000
S Chassis(su-config-cfm-mep.1000)->enable
S Chassis(su-config-cfm-mep.1000)->
```

CFM Loopback and Linktrace Protocols

There are two CFM diagnostic protocols available in MEP configuration mode: loopback and linktrace.

The CFM Loopback Protocol

The loopback protocol sends loopback messages (LBM) to either a specified maintenance end-point (MEP) or to an MP MAC address. The CFM loopback protocol is similar to IP ICMP ping. The CFM loopback protocol displays whether there is connectivity between the initiating device and the target device. If an operational path to the MEP or MIP exists, the remote MEP or MIP will respond. If no response is received by the source MEP, no operational path exists. If no response is received use the linktrace protocol ("The CFM Linktrace Protocol" on page 34-23) to help verify where in the path the problem occurred.

If you specify a MEP, the specified remote MEP must have already communicated with this MEP and there must be an entry in the MEP database for the remote MEP. The loopback will fail if the remote MEP is specified and the specified MEP has not yet communicated with this MEP.

The LBM 802.1p priority can be configured when initiating a loopback.

This example shows how to send 5 loopback messages to MAC address **01:3a:b2:af:65:de** from the **myMA1** MEP **1000**:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->mep 1000
S Chassis(su-config-cfm-mep.1000)->loopback mac 01:3a:b2:af:65:de messages 5
Sending 5 Ethernet CFM loopback messages to 01-3a-b2-af-65-de
...
Success rate is 100 percent (5/5)
S Chassis(su-config-cfm-mep.1000)->
```

The CFM Linktrace Protocol

The CFM linktrace protocol is used to help verify a path and identify where in a path a connectivity problem exists by indicating that an incompete path between the initiating device and the target device exists. The CFM linktrace protocol is similar to IP traceroute. Linktrace messages (LTM) are sent to either a specified MEP or to an MP MAC address. If you specify a MEP, the specified remote MEP must have already communicated with this MEP and there must

be an entry in the MEP database for the remote MEP. The linktrace will fail if the remote MEP is specified and the specified MEP has not yet communicated with this MEP.

You can specify a TTL which sets the maximum number of network hops the LTM will traverse before expiring.

You can specify whether each hop along the linktrace path uses the local filter database exclusively or is allowed to also use the local MIP CCM database to determine reachability to the target.

The LTM shares the same priority as the continuity check message and can not be separately configured. The priority for continuity check messages and LTMs is set using the **priority** command in MEP configuration mode.

This example shows how to send linktrace messages to MEP **2000** from the **myMA1** MEP **1000**:

```
S Chassis(rw-config)->cfm md string-name myMD1
S Chassis(su-config-cfm-md.1)->ma string-name myMA1
S Chassis(su-config-cfm-ma.1)->mep 1000
S Chassis(su-config-cfm-mep.1000)->linktrace mep 2000

Linktrace to 00-00-00-10-00-03, Transaction ID 29481

MD Name: abc
MA Name: abc
MEP ID : 1, Interface ge.4.18
=====================================================
Hop  TTL        Source MAC      Next hop MAC    Relay
---- ---- ----------------- ----------------- -------
   1   63 00-1f-45-9e-3e-d1 00-00-00-10-00-00  MIP-DB
   2   62 00-00-00-10-00-00 00-00-00-00-00-00     Hit

S Chassis(su-config-cfm-mep.1000)->
```

Configuring Connectivity Fault Management

Table 34-4 lists the S-Series device default CFM configuration settings.

Table 34-4 Default Connectivity Fault Management Configuration Settings

Parameter	Description	Default Value
CFM global state	The global state of CFM as configured in global configuration mode.	Disabled
CFM logging filter	The ability to limit the sending of CFM Syslog messages by MD, MA, and MEP.	All Syslog messages are sent
ID permission	The content sent in the SenderID TLV by maintenance points.	None
MD level	The maintenance domain level.	0
MHF creation	Sets whether the creation of MIP half functions is allowed.	None

Table 34-4 Default Connectivity Fault Management Configuration Settings (continued)

Parameter	Description	Default Value
CFM activation	The activation of CFM configuratin changes in an MD, MA, MA component, or MEP configuration context.	Not activated
CCM interval	The interval between the transmission of continuity check messages (CCM).	1 second
VLAN association	The VLAN the MA and MEP are associated with.	0
MEP state machine admin state	The administrative state of the MEP state machine.	Inactive
Minimum alarm defect reported	The minimum defect that will cause a Syslog or trap to be sent.	mac-rem-err-xcom
Alarm time	The minimum time a defect must be present before an alarm is generated.	2.5 seconds
CCI enabled	Determines whether continuity check messages (CCM) are enabled for generation and reception by a MEP.	Disabled
MEP direction	The direction a MEP faces which determines whether CFM PDUs are sent towards the link or bridge relay.	Down
CCM 802.1p priority	The 802.1p priority setting for CCMs and linktrace messages.	1
Remote MEP active	Determines the remote MEP state for the MEP being configured.	Disabled
Defect reset time	The time a MEP defect must be absent before an alarm is reset.	2.5 seconds

Table 34-5 lists CFM global configuration commands. Global configuration commands provide access to the CFM global default MD and monitored VLAN service configuration modes and the MD configuration mode (for a specific MD), the global enabling of CFM, and CFM logging filter configuration. These commands are accessed in global configuration mode.

Table 34-5 CFM Global Configuration

Task	Command
Access the CFM system default configuration mode.	**cfm default-md default**
Access the default configuration mode for a monitored VLAN service.	**cfm default-md vid** *vlan-id*
Globally enable CFM on the device. CFM is disabled by default	**cfm enable**
Filter the sending of CFM Syslog messages by maintenance domains (MDs), maintenance associations (MAs), and maintenance end points (MEPs). All CFM Syslog messages are sent by default.	**cfm logging filter md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} [**ma string-name** *name* | **vid-name** *vid-name* | **id-name** *id-name* | **index** *index*} [**mep** *mep-id*]
Enter Maintenance Domain (MD) Configuration mode for a specific named MD. If the maintenance domain does not exist, this command will create it.	**cfm md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name**}

Table 34-5 CFM Global Configuration (continued)

Task	Command
Optionally, configure a VLAN table to associate one or more CFM services with a primary CFM service	**cfm vlan-table primary-selector** *primary-selector* **selector-list** *selector-list* [**enable** | **disable**]

Table 34-6 on page 34-26 lists CFM system MD and monitored VLAN service default configuration commands. Enter the appropriate default MD configuration mode using commands listed in Table 34-5. In CFM system default configuration mode, entered default configuration is applied globally. In monitored VLAN service default configuration mode, entered default configuration is applied to the specified VLAN.

Table 34-6 CFM Global and Monitored VLAN Service Default Configuration

Task	Command
Optionally, configure the ID permission setting, within the appropriate default configuration mode, for the content sent in the SenderID TLV by the maintenance points. Defaults to **none**. The **defer** option is not supported in the global MD default command mode.	**id-permission** {**chassis** | **manage** | **chassis-manage** | **none** | **defer**}
Optionally, set the default maintenance domain level for the current default context. The MD level defaults to **0**.	**level** *level*
Optionally, set whether the creation of maintenance intermediate-point half function (MHF) is allowed for the current default context. Defaults to **none**.	**mhf-creation** {**default** | **explicit** | **none**}

Procedure 34-1 describes how to configure a MD. All commands listed in this procedure are entered in the MD configuration mode. See Table 34-5 on page 34-25 for the command to access the MD configuration mode.

Procedure 34-1 CFM Maintenance Domain (MD) Configuration

Step	Task	Command(s)
1.	Optionally, change the MD name for the current MD context.	**name** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name**}
2.	Optionally, configure the ID permission setting for the content sent in the SenderID TLV by the maintenance points. Defaults to **none**.	**id-permission** {**chassis** | **manage** | **chassis-manage** | **none** | **defer**}
3.	Optionally, set the maintenance domain level. The MD level defaults to **0**.	**level** *level*
4.	Optionally, set whether the creation of maintenance intermediate-point half function (MHF) is allowed. Defaults to **none**.	**mhf-creation** {**default** | **explicit** | **none** | **defer**}
5.	Activate the CFM configuration for the named MD context.	**enable**
6.	Optionally, enter Maintenance Association (MA) Configuration mode for the specified MA.	**ma** {**string-name** *name* | **vid-name** *vlan* | **id-name** *id*}

Procedure 34-2 describes how to configure an MA. All commands listed in this procedure are entered in the MA configuration mode. See Procedure 34-1 for the command to access the MA configuration mode.

Procedure 34-2 CFM Maintenance Association (MA) Configuration

Step	Task	Command(s)
1.	Specify the MEPs that are or will be present in the MA.	**mep-list** *mep-list* [**enable** | **disable**]
2.	Optionally, set the interval between continuity check messages (CCM)s. Defaults to **1** seconds.	**ccm-interval** {**1sec** | **10sec** | **1min** | **10min**}
3.	Optionally, change the Maintenance Association name for the current MA context.	**name** {**string-name** *name* | **vid-name** *vid-name* | **id-name** *id-name*}
4.	Optionally, enter MA component configuration mode for the MA.	**ma-comp**
5.	Activate the CFM configuration for the current MA context.	**enable**
6.	Optionally, enter Maintenance End-Point (MEP) Configuration mode for the specified end-point.	**mep** *mep-id*

Procedure 34-3 describes how to configure the MA components for the current MA context. All commands listed in this procedure are entered in the MA-Comp configuration mode. See Procedure 34-2 for the command to access the MA-Comp configuration mode.

Procedure 34-3 CFM Maintenance Association Component (MA-Comp) Configuration

Step	Task	Command(s)
1.	Specify the VLAN the maintenance association is associated with	**vid** *vlan-id*
2.	Optionally, configure the ID permission setting for the content sent in the SenderID TLV by the maintenance points for this MA context. Defaults to **none**.	**id-permission** {**chassis** | **manage** | **chassis-manage** | **none** | **defer**}
3.	Optionally, set whether the creation of maintenance intermediate-point half function (MHF) is allowed. Defaults to **none**.	**mhf-creation** {**default** | **explicit** | **none**}
4.	Activate the CFM configuration for the named MD context.	**enable**

Procedure 34-4 describes how to configure a MEP for the current MA context. All commands listed in this procedure are entered in the MEP configuration mode. See Procedure 34-2 on page 34-27 for the command to access the MEP configuration mode.

Procedure 34-4 CFM Maintenance Association End-Point (MEP) Configuration

Step	Task	Command(s)
1.	Optionally, set the lowest priority defect that will generate a fault alarm syslog message. Default is **mac-rem-err-xcon**.	**alarm-defect-syslog** {**all-def** | **mac-rem-err-xcon** | **rem-err-xcon** | **err-xcon** | **xcon** | **no-xcon**}
2.	Optionally, set the lowest priority defect that will generate a fault alarm trap message. Default is **mac-rem-err-xcon**.	**alarm-defect-trap** {**all-def** | **mac-rem-err-xcon** | **rem-err-xcon** | **err-xcon** | **xcon** | **no-xcon**}
3.	Optionally, set the minimum time a defect must be present before an alarm is generated. Default is **2.5** seconds.	**alarm-time** *time*
4.	Optionally, enable generation of continuity check messages. Default is disabled.	**cci-enabled**
5.	Optionally, configure whether the MEP faces the bridge relay (up) or the bridge port (down). Default is **down**	**direction** {**down** | **up**}
6.	Optionally, transmit CFM linktrace messages to the specified MEP or MAC address to help verify a path and identify where in a path a connectivity problem exists.	**tracelink** {**mep** *mep-id* | **mac** *mac-addr*} [**ttl** *time-to-live*] [**fdb-only**]
7.	Optionally, transmit CFM loopback messages to the specified MEP or MAC address to verify an operational path exists to the MEP.	**loopback** {**mep** *mep-id* | **mac** *mac-addr*} [**messages** *num-messages*] [**priority** *priority*] [**data** *data*]
8.	Configure the bridge port the MEP is attached to.	**port** *port*
9.	Optionally, configure the 802.1 priority for continuity check messages and linktrace message sent by this MEP. Default is **1**.	**priority** *priority*
10.	Enable a remote MEP in the database for the current MEP.	**remote-mep** *remote-mep-id* **active**
11.	Optionally, configure the time a MEP defect must be absent before an alarm is reset. Default is **2.5** seconds.	**reset-time** *time*
12.	Activate the administrative state of the MEP (maintenance end point) state machine. Defaults to inactive.	**active**
13.	Enable the CFM configuration for current MEP context. Defaults to disabled	**enable**

Table 34-7 describes how to manage CFM. CFM clear commands can be entered in any command mode.

Table 34-7 CFM Management Commands

Task	Command(s)
To clear the CFM bridge MIP CCM database.	**clear cfm bridge mip-ccm**

Table 34-7 CFM Management Commands (continued)

Task	Command(s)
To clear the check continuity message database for a specified MEP or all end-points for the specified context.	For a specified MD: **clear cfm ccm-database md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} [**mep** *mep-id*] For the specified or all endpoints in the specified MA: **clear cfm ccm-database md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} **ma** {**string-name** *name* | **vid-name** *vlan* | **id-name** *id* | **index** *index*} [**mep** *mep-id*]
To clear the MEP counters for a specified MEP or all end-points for the specified context.	For the specified or all end-points in the specified MD: **clear cfm counters md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} [**mep** *mep-id*] For the specified or all endpoints in the specified MA: **clear cfm counters md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} **ma** {**string-name** *name* | **vid-name** *vlan* | **id-name** *id* | **index** *index*} [**mep** *mep-id*] For the specified end-point: **clear cfm counters md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} **ma** {**string-name** *name* | **vid-name** *vlan* | **id-name** *id* | **index** *index*} **mep** *mep-id*

Table 34-8 describes how to display CFM configuration and statistics. CFM show commands can be entered in any command mode.

Table 34-8 CFM Show Commands

Task	Command(s)
To display CFM status and configuration for all CFM MDs, associations, and end-points.	**show cfm all**
To display maintenance intermediate point (MIP) continuity check message database entries for MHFs that do not belong to a specific MD and MA.	**show cfm bridge mip-ccm** [**vid** *vlan-id*]
To display the system level default MD or CFM service default values.	**show cfm default-md** [**default** | **vid** *vlan-id*]
To display MD information for all or the specified MD.	**show cfm md** [**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*]

Table 34-8 CFM Show Commands (continued)

Task	Command(s)
To display configuration and status for a specified or all MAs for the specified MD.	**show cfm md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} **ma** [**string-name** *name* | **vid-name** *vlan* | **id-name** *id* | **index** *index*]
To display the MA component configuration information.	**show cfm md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} **ma** {**string-name** *name* | **vid-name** *vlan* | **id-name** *id* | **index** *index*} **ma-comp**
To display the MEP configuration information.	**show cfm md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} **ma** {**string-name** *name* | **vid-name** *vlan* | **id-name** *id* | **index** *index*} **mep** [**mep-id** *mep-id*] [**-verbose**]
To display the error conditions in the MEP continuity check message database.	**show cfm md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} **ma** {**string-name** *name* | **vid-name** *vlan* | **id-name** *id* | **index** *index*} **mep** [**mep-id** *mep-id*] **ccm-errors** [**-verbose**]
To display linktrace database information for all or the specified end-point.	**show cfm md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} **ma** {**string-name** *name* | **vid-name** *vlan* | **id-name** *id* | **index** *index*} **mep** [**mep-id** *mep-id*] **linktrace** [**-verbose**]
to display the MA MEP list.	**show cfm md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} **ma** {**string-name** *name* | **vid-name** *vlan* | **id-name** *id* | **index** *index*} **mep-list**
to display the MEP's remote MEP configuration information.	**show cfm md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} **ma** {**string-name** *name* | **vid-name** *vlan* | **id-name** *id* | **index** *index*} **mep** [**mep-id** *mep-id*] **remote-mep** [**mep-id** *mep-id*] [**-verbose**]

Table 34-8 CFM Show Commands (continued)

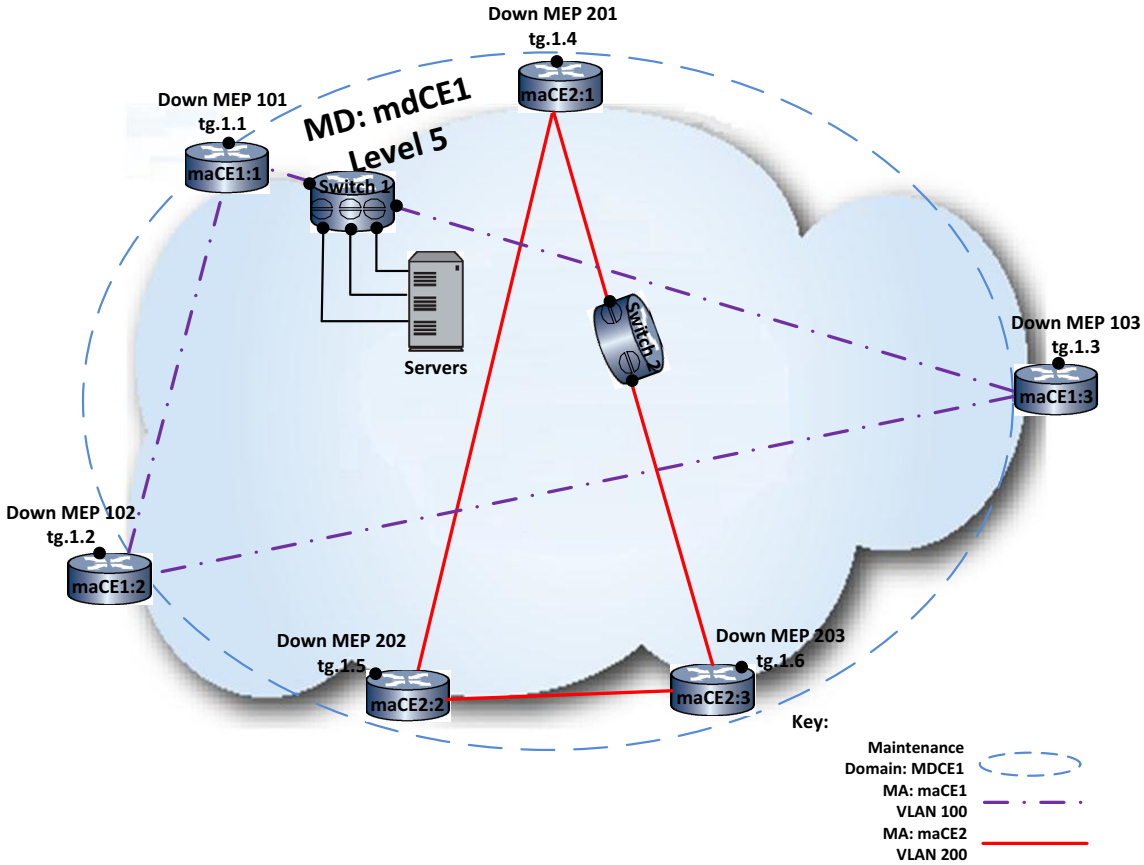
Task	Command(s)
to display maintenance intermediate point (MIP) continuity check message information by MD, MA, or MEP.	Display by MD: **show cfm md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} **mip-ccm** [**vid** *vlan-id*] Display by MA: **show cfm md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} **ma** {**string-name** *name* | **vid-name** *vlan* | **id-name** *id* | **index** *index*} **mip-ccm** [**vid** *vlan-id*] Display by MEP: **show cfm md** {**string-name** *name* | **dns-like-name** *dns-name* | **mac-int-name** *mac-name* | **no-name** | **index** *index*} **ma** {**string-name** *name* | **vid-name** *vlan* | **id-name** *id* | **index** *index*} **mep** [**mep-id** *mep-id*] **mip-ccm** [**vid** *vlan-id*]
To display the stack table which contains maintenance point information for each maintenance point for the device (MEP and MIP).	**show cfm stack-table**
To display the global CFM status for this device.	**show cfm status**

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

Single MD Configuration Example

A CFM monitored service can be configured within a single MD if you have administrative control of all the devices the service passes through. For our single MD CFM example we will monitor two VLANs. See Figure 34-6 for an overview of the CFM single MD configuration example.

Figure 34-6 Single MD Example Configuration Overview



The MD is named mdCE1 and uses the string-name naming convention. An MA is required for each monitored service. The MA monitoring VLAN 100 is named maCE1. The MA monitoring VLAN 200 is named maCE2. Each MA name uses the string-name naming convention. The naming format for each MEP enabled device is the name of the MA it belongs to with a ***:x*** where x is a numeric value. Each MA has a switch internal to the MD through which the monitored service passes that will be MIP enabled. Since there is only one MD to protect, and you are not concerned that CCMs will expose the network topology to any other operators, all MEPs in the example are down MEPs. Each MA has three MEPs configured.

Configuring Device maCE1:1

To configure maCE1:1:

- Enable CFM in global configuration mode
- Limit logging to MA **maCE1**
- Access MD mode for MD **mdCE1** using the string-name naming convention
- Set the MD level to **5**
- Enable the MD configuration
- Access MA mode for MA **maCE1** using the string-name naming convention
- Use the default CCM message interval of **1** second (not configured)
- Configure the MEP list for the two down MEPs: **101**, **102**, and **103**

- Enable the MA configuration
- Access the MA-Comp mode for **maCE1**
- Set VLAN **100** as the monitored service
- Enable MA-Comp configuration for **maCE1**
- Access MEP configuration for MEP **101**
- Set the MEP port to **tg.1.1**
- Set the MEP VLAN to **100**
- Set the MEP direction to **down** (optional because down is the direction default)
- Enable the sending of CCM messages for MEP **101**
- Set the remote MEP that CCM messages will be sent to **102** and **103**
- Activate the MEP state machines
- Enable the MEP configuration

Device maCE1:1 CLI Input

```
maCE1:1(rw-config)->cfm enable
maCE1:1(rw-config)->cfm logging filter md string-name mdCE1 ma string-name maCE1
maCE1:1(rw-config)->cfm md string-name mdCE1
maCE1:1(rw-config-cfm-md.1)->level 5
maCE1:1(rw-config-cfm-md.1)->enable
maCE1:1(rw-config-cfm-md.1)->ma string-name maCE1
maCE1:1(rw-config-cfm-ma.1)->mep-list 101-103 enable
maCE1:1(rw-config-cfm-ma.1)->enable
maCE1:1(rw-config-cfm-ma.1)->ma-comp
maCE1:1(rw-config-cfm-macomp)->vid 100
maCE1:1(rw-config-cfm-macomp)->enable
maCE1:1(rw-config-cfm-macomp)->exit
maCE1:1(rw-config-cfm-ma.1)->mep 101
maCE1:1(rw-config-cfm-mep.101)->port tg.1.1
maCE1:1(rw-config-cfm-mep.101)->vid 100
maCE1:1(rw-config-cfm-mep.101)->direction down
maCE1:1(rw-config-cfm-mep.101)->cci-enabled
maCE1:1(rw-config-cfm-mep.101)->remote-mep active 102,103
maCE1:1(rw-config-cfm-mep.101)->active
maCE1:1(rw-config-cfm-mep.101)->enable
maCE1:1(rw-config-cfm-mep.101)->exit
maCE1:1(rw-config-cfm-ma.1)->exit
maCE1:1(rw-config-cfm-ma.1)->exit
maCE1:1(rw-config)->
```

Configuring Device maCE1:2

To configure maCE1:2 use the information listed in “Configuring Device maCE1:1” on page 34-32 until you get to MEP configuration. At MEP configuration proceed here:

- Access MEP configuration for MEP **102**
- Set the MEP port to **tg.1.2**
- Set the MEP VLAN to **100**
- Set the MEP direction to **down** (optional because down is the direction default)
- Enable the sending of CCM messages for MEP **102**
- Set the remote MEP that CCM messages will be sent to **101** and **103**
- Activate the MEP state machines
- Enable the MEP configuration

Device maCE1:2 CLI Input

```
maCE1:2(rw-config)->cfm enable
maCE1:2(rw-config)->cfm logging filter md string-name mdCE1 ma string-name maCE1
maCE1:2(rw-config)->cfm md string-name mdCE1
maCE1:2(rw-config-cfm-md.1)->level 5
maCE1:2(rw-config-cfm-md.1)->enable
maCE1:2(rw-config-cfm-md.1)->ma string-name maCE1
maCE1:2(rw-config-cfm-ma.1)->mep-list 101-103 enable
maCE1:2(rw-config-cfm-ma.1)->enable
maCE1:2(rw-config-cfm-ma.1)->ma-comp
maCE1:2(rw-config-cfm-macomp)->vid 100
maCE1:2(rw-config-cfm-macomp)->enable
maCE1:2(rw-config-cfm-macomp)->exit
maCE1:2(rw-config-cfm-ma.1)->mep 101
maCE1:2(rw-config-cfm-mep.102)->port tg.1.2
maCE1:2(rw-config-cfm-mep.102)->vid 100
maCE1:2(rw-config-cfm-mep.102)->direction down
maCE1:2(rw-config-cfm-mep.102)->cci-enabled
maCE1:2(rw-config-cfm-mep.102)->remote-mep active 101,103
maCE1:2(rw-config-cfm-mep.102)->active
maCE1:2(rw-config-cfm-mep.102)->enable
maCE1:2(rw-config-cfm-mep.102)->exit
maCE1:2(rw-config-cfm-ma.1)->exit
maCE1:2(rw-config-cfm-ma.1)->exit
maCE1:2(rw-config)->
```

Configuring Device maCE1:3

To configure device maCE1:3 use the information listed in “Configuring Device maCE1:1” on page 34-32 until you get to MEP configuration. At MEP configuration proceed here:

- Access MEP configuration for MEP **103**
- Set the MEP port to **tg.1.3**
- Set the MEP VLAN to **100**
- Set the MEP direction to **down** (optional because down is the direction default)

- Enable the sending of CCM messages for MEP **103**
- Set the remote MEP that CCM messages will be sent to **101** and **102**
- Activate the MEP state machines
- Enable the MEP configuration

Device maCE1:3 CLI Input

```
maCE1:3(rw-config)->cfm enable
maCE1:3(rw-config)->cfm logging filter md string-name mdCE1 ma string-name maCE1
maCE1:3(rw-config)->cfm md string-name mdCE1
maCE1:3(rw-config-cfm-md.1)->level 5
maCE1:3(rw-config-cfm-md.1)->enable
maCE1:3(rw-config-cfm-md.1)->ma string-name maCE1
maCE1:3(rw-config-cfm-ma.1)->mep-list 101-103 enable
maCE1:3(rw-config-cfm-ma.1)->enable
maCE1:3(rw-config-cfm-ma.1)->ma-comp
maCE1:3(rw-config-cfm-macomp)->vid 100
maCE1:3(rw-config-cfm-macomp)->enable
maCE1:3(rw-config-cfm-macomp)->exit
maCE1:3(rw-config-cfm-ma.1)->mep 101
maCE1:3(rw-config-cfm-mep.103)->port tg.1.2
maCE1:3(rw-config-cfm-mep.103)->vid 100
maCE1:3(rw-config-cfm-mep.103)->direction down
maCE1:3(rw-config-cfm-mep.103)->cci-enabled
maCE1:3(rw-config-cfm-mep.103)->remote-mep active 101,103
maCE1:3(rw-config-cfm-mep.103)->active
maCE1:3(rw-config-cfm-mep.103)->enable
maCE1:3(rw-config-cfm-mep.103)->exit
maCE1:3(rw-config-cfm-ma.1)->exit
maCE1:3(rw-config-cfm-ma.1)->exit
maCE1:3(rw-config)->
```

Configuring Switch 1

To configure CE Device 1:

- Enable CFM in global configuration mode
- Limit logging to MA **maCE1**
- Configure the default MD VLAN to **100**
- Configure the default MD VLAN 100 MD level to **5**
- Set the default MD VLAN 100 MHF create setting to **default**

Switch 1 CLI Input

```
Switch 1(rw-config)->cfm enable
Switch 1(rw-config)->cfm logging filter md string-name mdCE1 ma string-name maCE1
Switch 1(rw-config)->cfm default-md vid 100
```

```
Switch 1(rw-config-def-md.100)->level 5
Switch 1(rw-config-def-md.100)->mhf-creation default
Switch 1(rw-config-def-md.100)->exit
Switch 1(rw-config)->
```

Configuring Device maCE2:1

To configure maCE2:1:

- Enable CFM in global configuration mode
- Limit logging to MA **maCE2**
- Access MD mode for MD **mdCE1** using the string-name naming convention
- Set the MD level to **5**
- Enable the MD configuration
- Access MA mode for MA **maCE2** using the string-name naming convention
- Use the default CCM message interval of **1** second (not configured)
- Configure the MEP list for the two down MEPs: **201**, **202**, and **203**
- Enable the MA configuration
- Access the MA-Comp mode for **maCE2**
- Set VLAN **200** as the monitored service
- Enable MA-Comp configuration for **maCE2**
- Access MEP configuration for MEP **201**
- Set the MEP port to **tg.1.4**
- Set the MEP VLAN to **200**
- Set the MEP direction to **down** (optional because down is the direction default)
- Enable the sending of CCM messages for MEP **201**
- Set the remote MEP that CCM messages will be sent to **202** and **203**
- Activate the MEP state machines
- Enable the MEP configuration

Device maCE2:1 CLI Input

```
maCE2:1(rw-config)->cfm enable
maCE2:1(rw-config)->cfm logging filter md string-name mdCE1 ma string-name maCE2
maCE2:1(rw-config)->cfm md string-name mdCE1
maCE2:1(rw-config-cfm-md.1)->level 5
maCE2:1(rw-config-cfm-md.1)->enable
maCE2:1(rw-config-cfm-md.1)->ma string-name maCE2
maCE2:1(rw-config-cfm-ma.2)->mep-list 201-203 enable
maCE2:1(rw-config-cfm-ma.2)->enable
maCE2:1(rw-config-cfm-ma.2)->ma-comp
maCE2:1(rw-config-cfm-macomp)->vid 200
maCE2:1(rw-config-cfm-macomp)->enable
```

```
maCE2:1(rw-config-cfm-macomp)->exit
maCE2:1(rw-config-cfm-ma.2)->mep 201
maCE2:1(rw-config-cfm-mep.201)->port tg.1.4
maCE2:1(rw-config-cfm-mep.201)->vid 200
maCE2:1(rw-config-cfm-mep.201)->direction down
maCE2:1(rw-config-cfm-mep.201)->cci-enabled
maCE2:1(rw-config-cfm-mep.201)->remote-mep active 202,203
maCE2:1(rw-config-cfm-mep.201)->active
maCE2:1(rw-config-cfm-mep.201)->enable
maCE2:1(rw-config-cfm-mep.201)->exit
maCE2:1(rw-config-cfm-ma.2)->exit
maCE2:1(rw-config-cfm-ma.2)->exit
maCE2:1(rw-config)->
```

Configuring Device maCE2:2

To configure maCE2:2 use the information listed in "Configuring Device maCE1:1" on page 34-32 until you get to MEP configuration. At MEP configuration proceed here:

- Access MEP configuration for MEP **202**
- Set the MEP port to **tg.1.5**
- Set the MEP VLAN to **200**
- Set the MEP direction to **down** (optional because down is the direction default)
- Enable the sending of CCM messages for MEP **202**
- Set the remote MEP that CCM messages will be sent to **201** and **203**
- Activate the MEP state machines
- Enable the MEP configuration

Device maCE2:2 CLI Input

```
maCE2:2(rw-config)->cfm enable
maCE2:2(rw-config)->cfm logging filter md string-name mdCE1 ma string-name maCE1
maCE2:2(rw-config)->cfm md string-name mdCE1
maCE2:2(rw-config-cfm-md.1)->level 5
maCE2:2(rw-config-cfm-md.1)->enable
maCE2:2(rw-config-cfm-md.1)->ma string-name maCE1
maCE2:2(rw-config-cfm-ma.2)->mep-list 101-103 enable
maCE2:2(rw-config-cfm-ma.2)->enable
maCE2:2(rw-config-cfm-ma.2)->ma-comp
maCE2:2(rw-config-cfm-macomp)->vid 100
maCE2:2(rw-config-cfm-macomp)->enable
maCE2:2(rw-config-cfm-macomp)->exit
maCE2:2(rw-config-cfm-ma.2)->mep 101
maCE2:2(rw-config-cfm-mep.202)->port tg.1.2
maCE2:2(rw-config-cfm-mep.202)->vid 100
maCE2:2(rw-config-cfm-mep.202)->direction down
```

```
maCE2:2(rw-config-cfm-mep.202)->cci-enabled
maCE2:2(rw-config-cfm-mep.202)->remote-mep active 101,103
maCE2:2(rw-config-cfm-mep.202)->active
maCE2:2(rw-config-cfm-mep.202)->enable
maCE2:2(rw-config-cfm-mep.202)->exit
maCE2:2(rw-config-cfm-ma.2)->exit
maCE2:2(rw-config-cfm-ma.2)->exit
maCE2:2(rw-config)->
```

Configuring Device maCE2:3

To configure device maCE1:3 use the information listed in "Configuring Device maCE1:1" on page 34-32 until you get to MEP configuration. At MEP configuration proceed here:

- Access MEP configuration for MEP **203**
- Set the MEP port to **tg.1.6**
- Set the MEP VLAN to **200**
- Set the MEP direction to **down** (optional because down is the direction default)
- Enable the sending of CCM messages for MEP **203**
- Set the remote MEP that CCM messages will be sent to **201** and **202**
- Activate the MEP state machines
- Enable the MEP configuration

Device maCE2:3 CLI Input

```
maCE2:3(rw-config)->cfm enable
maCE2:3(rw-config)->cfm logging filter md string-name mdCE1 ma string-name maCE2
maCE2:3(rw-config)->cfm md string-name mdCE1
maCE2:3(rw-config-cfm-md.1)->level 5
maCE2:3(rw-config-cfm-md.1)->enable
maCE2:3(rw-config-cfm-md.1)->ma string-name maCE2
maCE2:3(rw-config-cfm-ma.1)->mep-list 201-203 enable
maCE2:3(rw-config-cfm-ma.1)->enable
maCE2:3(rw-config-cfm-ma.1)->ma-comp
maCE2:3(rw-config-cfm-macomp)->vid 200
maCE2:3(rw-config-cfm-macomp)->enable
maCE2:3(rw-config-cfm-macomp)->exit
maCE2:3(rw-config-cfm-ma.1)->mep 201
maCE2:3(rw-config-cfm-mep.102)->port tg.1.6
maCE21:3(rw-config-cfm-mep.102)->vid 200
maCE2:3(rw-config-cfm-mep.102)->direction down
maCE2:3(rw-config-cfm-mep.102)->cci-enabled
maCE2:3(rw-config-cfm-mep.102)->remote-mep active 201,203
maCE2:3(rw-config-cfm-mep.102)->active
maCE2:3(rw-config-cfm-mep.102)->enable
maCE2:3(rw-config-cfm-mep.102)->exit
```

```
maCE2:3(rw-config-cfm-ma.1)->exit
maCE2:3(rw-config-cfm-ma.1)->exit
maCE2:3(rw-config)->
```

Configuring Switch 2

To configure CE Device 2:

- Enable CFM in global configuration mode
- Limit logging to MA **maCE2**
- Configure the default MD VLAN to **100**
- Configure the default MD VLAN 100 MD level to **5**
- Set the default MD VLAN 100 MHF create setting to **default**

Switch 1 CLI Input

```
Switch 2(rw-config)->cfm enable
Switch 2(rw-config)->cfm logging filter md string-name mdCE2 ma string-name maCE1
Switch 2(rw-config)->cfm default-md vid 100
Switch 2(rw-config-def-md.100)->level 5
Switch 2(rw-config-def-md.100)->mhf-creation default
Switch 2(rw-config-def-md.100)->exit
Switch 2(rw-config)->
```

Multiple MD Configuration Example

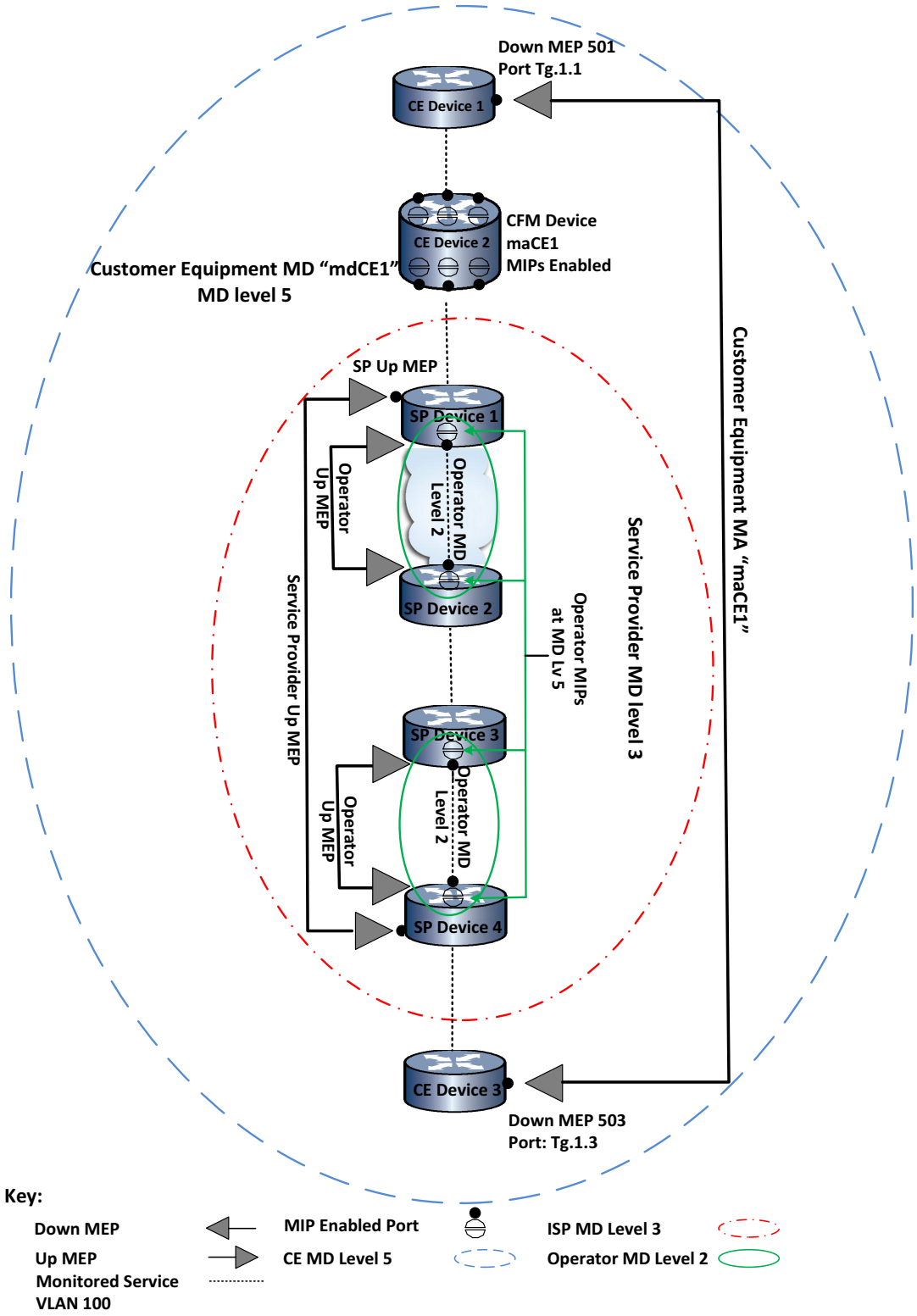
A CFM configuration requires multiple MDs when one or more network segments of a monitored service are either not directly under the control of a single administrator or the administrator chooses to segment the network for administrative purposes. Our multiple MD configuration example is a basic customer equipment VLAN monitored service that passes through the service provider MD and two network operator MDs. See the "Single MD Configuration Example" on page 34-31 if you are using CFM in a single administrative control network environment.

The example will provide the customer equipment configuration for the two monitored service end points (CE Device 1 and CE device 3) and a MIP enabled device (CE Device 2) that will snoop CCMs that traverse the service between the two end points. CE Device 2 configuration is representative of all devices within the customer equipment MD (level 5) that reside between the two end points with the exception that enabling MIPs is optional.

See Figure 34-7 on page 34-40 for a presentation of the multiple MD configuration. The configuration of the service provider and network operator MDs is transparent to the customer equipment MD and is discussed only to the extent that it relates to understanding the customer equipment configuration.

In our example, CFM monitors VLAN 100. The customer equipment resides within MD level 5. The service provider equipment resides within MD 3, and the two network operators' equipment resides within two MD level 2 domains. The down MEPs 501 and 503 are the two customer service end points for the service. MIPs are enabled on customer equipment CE Device 2 to snoop CCMs that traverse the device. The customer equipment administrator has requested that the network operators enable MIPs on their Up MEPs configured for MD level 5 providing snooping of CCMs from MEPs 501 and 503 that transit the service. This configuration is performed by the network operator administrators. MIPs are not enabled on service provider ports to prevent exposing the customer service network topology within the service provider network.

Figure 34-7 Multiple MD Example Configuration Overview



Configuring CE Device 1

To configure CE Device 1:

- Enable CFM in global configuration mode
- Limit logging to MA **maCE1**
- Access MD mode for MD **mdCE1** using the string-name naming convention
- Set the MD level to **5**
- Enable the MD configuration
- Access MA mode for MA **maCE1** using the string-name naming convention
- Set the interval between sending CCM messages to **10** seconds
- Configure the MEP list for the two down MEPs: **501** and **503**
- Enable the MA configuration
- Access the MA-Comp mode for **maCE1**
- Set VLAN 100 as the monitored service
- Enable MA-Comp configuration for **maCE1**
- Access MEP configuration for MEP **501**
- Set the MEP port to **tg.1.1**
- Set the MEP VLAN to **100**
- Set the MEP direction to **down** (optional because down is the direction default)
- Enable the sending of CCM messages for MEP **501**
- Set the remote MEP that CCM messages will be sent to **503**
- Activate the MEP state machines
- Enable the MEP configuration

CE Device 1 CLI Input

```
CE Device 1(rw-config)->cfm enable
CE Device 1(rw-config)->cfm logging filter md string-name mdCE1 ma string-name
maCE1
CE Device 1(rw-config)->cfm md string-name mdCE1
CE Device 1(rw-config-cfm-md.1)->level 5
CE Device 1(rw-config-cfm-md.1)->enable
CE Device 1(rw-config-cfm-md.1)->ma string-name maCE1
CE Device 1(rw-config-cfm-ma.1)->ccm-interval 10sec
CE Device 1(rw-config-cfm-ma.1)->mep-list 501,503 enable
CE Device 1(rw-config-cfm-ma.1)->enable
CE Device 1(rw-config-cfm-ma.1)->ma-comp
CE Device 1(rw-config-cfm-macomp)->vid 100
CE Device 1(rw-config-cfm-macomp)->enable
CE Device 1(rw-config-cfm-macomp)->exit
CE Device 1(rw-config-cfm-ma.1)->mep 501
CE Device 1(rw-config-cfm-mep.501)->port tg.1.1
CE Device 1(rw-config-cfm-mep.501)->vid 100
CE Device 1(rw-config-cfm-mep.501)->direction down
CE Device 1(rw-config-cfm-mep.501)->cci-enabled
```

```
CE Device 1(rw-config-cfm-mep.501)->remote-mep active 503
CE Device 1(rw-config-cfm-mep.501)->active
CE Device 1(rw-config-cfm-mep.501)->enable
CE Device 1(rw-config-cfm-mep.501)->exit
CE Device 1(rw-config-cfm-ma.1)->exit
CE Device 1(rw-config-cfm-ma.1)->exit
CE Device 1(rw-config)->
```

Configuring CE Device 2

To configure CE Device 2:

- Enable CFM in global configuration mode
- Limit logging to MA **maCE1**
- Access MD mode for MD **mdCE1** using the string-name naming convention
- Set the MD level to **5**
- Enable the MD configuration
- Access MA mode for MA **maCE1** using the string-name naming convention
- Enable the MA configuration
- Access the MA-Comp mode for **maCE1**
- Set VLAN 100 as the monitored service
- Enable MIP creation for all ports on the device
- Enable MA-Comp configuration for **maCE1**

CE Device 2 CLI Input

```
CE Device 2(rw-config)->cfm enable
CE Device 2(rw-config)->cfm logging filter md string-name mdCE1 ma string-name
maCE1
CE Device 2(rw-config)->cfm md string-name mdCE1
CE Device 2(rw-config-cfm-md.1)->level 5
CE Device 2(rw-config-cfm-md.1)->enable
CE Device 2(rw-config-cfm-md.1)->ma string-name maCE1
CE Device 2(rw-config-cfm-ma.1)->enable
CE Device 2(rw-config-cfm-ma.1)->mep-list 501,503 enable
CE Device 2(rw-config-cfm-ma.1)->ma-comp
CE Device 2(rw-config-cfm-macomp)->vid 100
CE Device 2(rw-config-cfm-macomp)->mhf-creation default
CE Device 2(rw-config-cfm-macomp)->enable
CE Device 2(rw-config-cfm-macomp)->exit
CE Device 2(rw-config-cfm-ma.1)->exit
CE Device 2(rw-config-cfm-md.1)->exit
CE Device 2(rw-config)->
```

Configuring CE Device 3

To configure CE Device 3:

- Enable CFM in global configuration mode
- Limit logging to MA **maCE1**
- Access MD mode for MD **mdCE1** using the string-name naming convention
- Set the MD level to **5**
- Enable the MD configuration
- Access MA mode for MA **maCE1** using the string-name naming convention
- Set the interval between sending CCM messages to **10** seconds
- Configure the MEP list for the two down MEPs: **501** and **503**
- Enable the MA configuration
- Access the MA-Comp mode for **maCE1**
- Set VLAN 100 as the monitored service
- Enable MA-Comp configuration for **maCE1**
- Access MEP configuration for MEP **503**
- Set the MEP port to **tg.1.3**
- Set the MEP VLAN to **100**
- Set the MEP direction to **down** (optional because down is the direction default)
- Enable the sending of CCM messages for MEP **503**
- Set the remote MEP that CCM messages will be sent to **501**
- Activate the MEP state machines
- Enable the MEP configuration

CE Device 3 CLI Input

```
CE Device 3(rw-config)->cfm enable
CE Device 3(rw-config)->cfm logging filter md string-name mdCE1 ma string-name
maCE1
CE Device 3(rw-config)->cfm md string-name mdCE1
CE Device 3(rw-config-cfm-md.1)->level 5
CE Device 3(rw-config-cfm-md.1)->enable
CE Device 3(rw-config-cfm-md.1)->ma string-name maCE1
CE Device 3(rw-config-cfm-ma.1)->ccm-interval 10sec
CE Device 3(rw-config-cfm-ma.1)->mep-list 501,503 enable
CE Device 3(rw-config-cfm-ma.1)->enable
CE Device 3(rw-config-cfm-ma.1)->ma-comp
CE Device 3(rw-config-cfm-macomp)->vid 100
CE Device 3(rw-config-cfm-macomp)->enable
CE Device 3(rw-config-cfm-macomp)->exit
CE Device 3(rw-config-cfm-ma.1)->mep 503
CE Device 3(rw-config-cfm-mep.501)->port tg.1.3
```

```
CE Device 3(rw-config-cfm-mep.501)->vid 100
CE Device 3(rw-config-cfm-mep.501)->direction down
CE Device 3(rw-config-cfm-mep.501)->cci-enabled
CE Device 3(rw-config-cfm-mep.501)->active
CE Device 3(rw-config-cfm-mep.501)->remote-mep active 501
CE Device 3(rw-config-cfm-mep.501)->enable
CE Device 3(rw-config-cfm-ma.1)->exit
CE Device 3(rw-config-cfm-ma.1)->exit
CE Device 3(rw-config)->
```

Terms and Definitions

Table 34-9 lists terms and definitions used in this CFM configuration discussion.

Table 34-9 Connectivity Fault Management (CFM) Terms and Definitions

Term	Definition
CFM Linktrace Protocol	A diagnostic protocol used to verify the path between the initiating device and a target device, and to help determine where in the path a problem might exist.
CFM Loopback Protocol	A diagnostic protocol used to determine connectivity between the initiating device and the target device.
CFM monitored service	The network entity monitored by CFM (currently a single VLAN per MA).
Connectivity Check Messages (CCM)	A CCM is a unidirectional multicast message, confined to a single operator domain that provides a means to detect connectivity failures in an MA. These messages are unidirectional and do not solicit a response. Each MEP transmits a periodic multicast CCM inward towards the other MEPs belonging to the operator.
Connectivity Fault Management (CFM)	Provides network operators the means to monitor and troubleshoot services that may span multiple domain Ethernet networks based upon IEEE 802.1Q-2011.
ID Permission	Specifies whether chassis or remote management information or both are sent in the SenderID TLV.
Maintenance Association (MA)	A logical container for a specific CFM monitored service.
Maintenance Domain (MD)	A logical container for all the equipment associated with the CFM monitored service and owned by a single network operator.
Maintenance End-Point (MEP)	A port, belonging to an MA, through which data enters and exits the portion of the network monitored by the CFM service.
Maintenance Intermediate-Point (MIP)	An auto-created MP on a port that resides along the path between MEPs, supplements the MEP functionality by passively snooping the CCMs that pass through it, and is able to respond to a loopback or linktrace message.
Maintenance Point (MP)	A demarcation point on a port that implements the CFM functions within a MA.
MD level	A CFM MD value that determines the reach and scope of the organization controlling the MD, with higher levels able to encapsulate lower levels, allowing CCMs to pass transparently through lower level MDs.
MEP direction	Specifies whether a MEP faces the link or the bridge relay thereby controlling the transmission direction of CFM PDUs by the MEP.
MEP list	A list containing all the MEPs in an MA.

Table 34-9 Connectivity Fault Management (CFM) Terms and Definitions (continued)

Term	Definition
MHF creation	Specifies whether MIPs are created on the ports for the configured device.
VLAN table	The association of one or more CFM services with a primary CFM service.

35

Virtual Routing and Forwarding (VRF) Configuration

This document provides the following information about configuring Virtual Routing and Forwarding (VRF) on the Extreme Networks S-Series platforms.

Using VRF in Your Network

Virtual Routing and Forwarding (VRF) provides a method of partitioning your network into different routed domains. A VRF is a segregated domain for the routed forwarding of packets. VRFs are used to divide a router into multiple standalone forwarding domains that may contain unique IP networks, routes, and other configuration that would otherwise conflict if they were all deployed on the same router. VRFs can exchange routes between one another. An Interface may be configured to one and only one VRF. An interface configured to a particular VRF is considered a member of that VRF. One or more VRF(s) can be used as a gateway (or access point) to a larger Internet. VRFs with overlapping IP networks that communicate to a larger internet can coexist, using the Network Address Translation (NAT) feature NAT-inside-VRF.

Implementing VRF

To configure a VRF:

- Create the VRF in any command mode, optionally specifying an SNMPv3 context name.
- Enter the VRF router mode, followed by entering configuration mode for the created VRF.
- For each VRF with a subnet reachable by a different VRF instance, configure static routes to perform next hop lookup in the VRF instance.
- For IP address policy, in which the next hop interface is a member of a different VRF, when configuring a policy route map, set the next hop behavior to perform the route lookup on the next hop VRF.

- When multiple VRFs contain overlapping IP networks that communicate to a larger internet, use the NAT-inside-VRF feature to differentiate between the VRFs containing the overlapping IP networks.
- When a single VRF provides Server Load Balancing (SLB) services for multiple VRFs, configure the virtual server to provide SLB services to all VRFs in this router.
- When changing the destination address for the forwarding of local UDP broadcasts to an address located on a different VRF, specify the destination VRF in the helper address configuration. Also, set DHCP relay information to force the client to include VPN option 82 in packets sent to the DHCP server.

VRF Overview

S-Series devices have a single default router named “global”. The global router:

- Exists when you first boot the device
- Manages the VRFs for this physical router
- Can neither be created nor deleted
- Can manage up to 128 VRF instances depending upon your system

Use the **show limits application vrf** command to determine the number of VRF instances your system supports.

Each optional VRF instance you create functions as its own routing domain. All routing features and protocols that are supported on the global router are also supported in a VRF instance. VRF instance router protocol configuration (for example, configuring PIM, OSPF, and IGMP) is identical to the global router protocol configuration. Protocol configurations in different VRFs do not conflict with each other because they are completely separate instances of the protocol.

You create a VRF router, in any command mode, using the **set router vrf create** command. The command requires that you specify a name of up to 31 printable characters, except for the space character. Extreme Networks recommends that you provide the VRF with a meaningful name such as “Marketing” or “Internet-Access”.

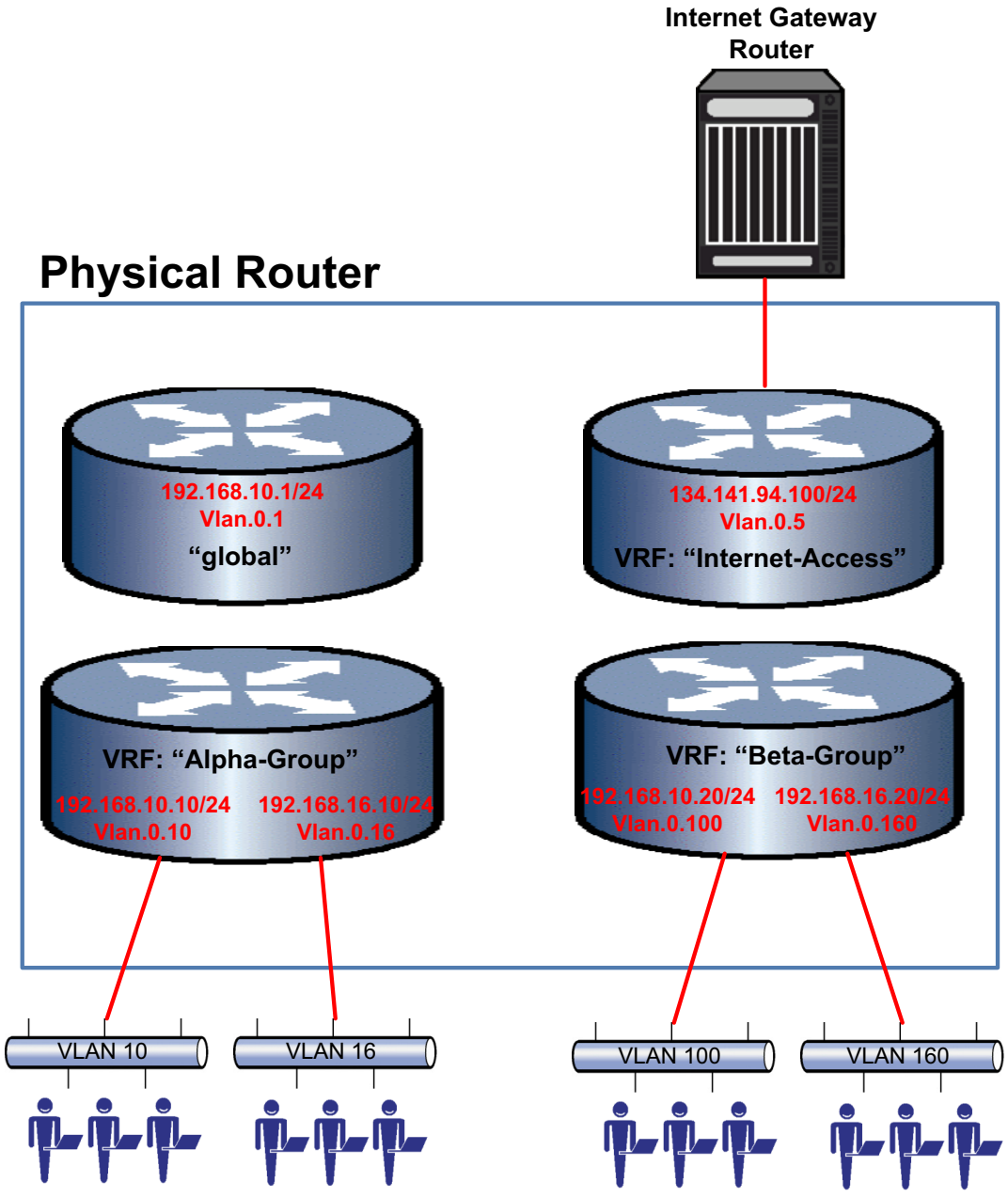
You can optionally specify an SNMPv3 context of up to 28 characters. If not specified, the SNMPv3 context defaults to the name of the VRF instance. If the VRF instance name exceeds 28 characters, the SNMPv3 context must be specified when creating the VRF. Refer to the **set router vrf create** command for information on creating a VRF instance.

The behavior when clearing the global router is different versus clearing a VRF instance. When you clear the global router, a blank configuration file is written to persistent memory. The global router is not deleted. Unlike the global router, all VRFs can be both created and deleted. When you clear a VRF, the VRF is deleted along with all of its configuration.

Use the **clear router vrf** command to clear the global router configuration or to delete a VRF instance from the system.

Figure 35-1 on page 35-3 presents a router that has been segmented into three VRF routers: two VRF routers with user group access named Alpha-Group and Beta-Group, and a VRF for internet access named Internet-Access.

Figure 35-1 VRF Overview



VRFs, Interfaces, and IP Addresses

By default, interfaces do not belong to any VRF instance until they are assigned. An interface may belong to only one VRF at a time. When you first create a VRF, the next available loopback interface is assigned as the default interface for the VRF router. Once bound to a VRF router, interfaces are configured in that VRF router context. You must first remove the bound VRF

interface from its current VRF instance before moving the interface to a different VRF instance. To remove an interface from a VRF instance, along with all its configuration, use the command **no interface** *interface-name*.

In VRF configuration mode, the **interface** *interface-name* command automatically binds the named interface to the current VRF and enters interface configuration mode. If the interface has already been bound to a different VRF, an error message is displayed.

IP addresses assigned in different VRFs are completely separate, thus overlapping or identical IP addressing is permitted across different VRFs. For example, VRF "Corporate" may have IP address range 10.1.100.1/16 associated with interface ge.1.1 while the "Marketing" VRF has IP address range 10.1.100.1/16 associated with interface ge.1.2. As a packet ingresses the router, the interface it ingresses on will determine which VRF router will receive it.

The routing tables for each VRF router will handle routes within the physical router for overlapping IP addresses. If an overlapping IP address requires communication with the outside internet through a shared-access-VRF, you must configure the IP address for NAT-inside-VRF on the shared-access-VRF so that it will know how to communicate with the correct VRF. See "VRFs With Overlapping IP Networks" on page 35-5 for NAT-inside-VRF details.

VRF and Static Route Next Hop Lookup

When a subnet is reachable from a VRF different from the ingress VRF, a static route can be configured specifying that the egress VRF instance performs the next hop lookup.

Use the **ip route** {*prefix mask* | *prefix/prefix-length*} **vrf** *egress-vrf* command to configure an egress VRF to perform the static route next hop lookup.

Note: The default VRF router is referred to as the **global** router. Named VRF routers within a device configured using the **set router vrf create** command are referred to as non-global VRF routers. Static routes are supported between both the **global** router and any non-global VRF router and between any two non-global VRF routers.

Refer to Figure 35-1 on page 35-3 for the following discussion. Only VRF Internet-Access contains next hop information for destination addresses reachable by the internet gateway router. If a packet ingresses on VLAN 10 for IP address 192.168.10.5, with a destination address of 66.249.81.104 that is only reachable by the internet gateway router, a lookup on the VRF Alpha-Group route table will fail. By configuring a static route on VRF Alpha-Group pointing to VRF Internet-Access as the egress VRF, the Internet-Access VRF will be used for the next hop lookup destination address 66.249.81.104.

Note: Using the **vrf** *vrf-name* parameter is more dynamic than configuring a standard static route, in that it determines the next hop based upon a route table lookup. A standard static route specifies a single next hop. Should that next hop be unavailable, the subnet is no longer reachable. A standard static route can be configured to reach the next hop that is a member of a different VRF using the syntax: **ip route** *destination-prefix/length next-hop-address* **interface** *next-hop-interface*. Because the **vrf** *vrf-name* parameter provides greater flexibility in determining the next hop, it is recommended that you use the **vrf** *vrf-name* parameter.

This example shows how to specify on VRF Alpha-Group that the next hop lookup to destination prefix 66.249.81.0/24, for packets ingressing on VRF Alpha-Group, is performed on VRF Internet-Access:

```
S Chassis(rw-*ha-Group-config)->ip route 66.249.81.0/24 vrf Internet-Access
```

This example shows how to specify on VRF Alpha-Group that the next hop lookup to destination address **2001:11ac:fd34::/48**, for packets ingressing on VRF Alpha-Group, is performed on VRF Internet-Access:

```
S Chassis(rw-*ha-Group-config)->ipv6 route 2001:11ac:fd34::/48 vrf
Internet-Access
```

VRF and Set Policy Next Hop Lookup

VRF segmented systems support overlapping IP addresses because the interface each IP address belongs to are members of a particular VRF. When configuring a policy route map on a VRF, in which the next hop for an IP address match belongs to a different VRF, the next hop VRF that will perform the route lookup must be specified.

Use the **set vrf** *vrf-name* command to configure the VRF that will perform the next hop lookup for the IP address match.

Only one set VRF clause is allowed, and only one VRF can be specified. All subsequent set clauses are ignored if a valid set VRF clause is detected. A set VRF clause is valid when the specified VRF name exists. If the VRF exists, the packet is forwarded to the VRF, even if there are no interfaces or any other configuration present.

If the VRF specified in the set clause does not exist, then any other existing set clause will be processed, and the frame is forwarded by the VRF it came in on.

This example shows how to set VRF **vr2** to determine the next hop, for policy route map 101, based upon its route table lookup:

```
S Chassis(rw-vr1-config)->route-map policy 101 permit 20
S Chassis(rw-vr1-config-route-map-pbr)->match ip address 1
S Chassis(rw-vr1-config-route-map-pbr)->set vrf vr2
```

VRFs With Overlapping IP Networks

A shared-access-VRF is a VRF that provides the access to the outside internet to one or more VRFs in the system that do not have direct access to the internet. Multiple VRFs that contain overlapping IP networks do not provide any means of determining which of the overlapping VRFs the packet is intended for, when packets ingress a shared-access-VRF .

In Figure 35-2 on page 35-6, Packet A ingresses the VRF segmented router on VRF Alpha-Group using VLAN 10. Even though overlapping 192.168.10.10/24 IP networks exist on both the VRF Alpha-Group and VRF Beta-Group, the VLAN Packet A ingresses on determines the VRF that will route the packet.

Packet B ingresses the system on the shared-access-VRF Internet-Access. Packet B ultimately needs to be routed to 192.168.10.15 on VRF Alpha-Group, which is a member of subnet 192.168.10.10/24 on VLAN 10. Subnet 192.168.10.10/24 on VRF Alpha-Group VLAN 10 overlaps with subnet 192.168.10.10/24 on VRF Beta-Group VLAN 100.

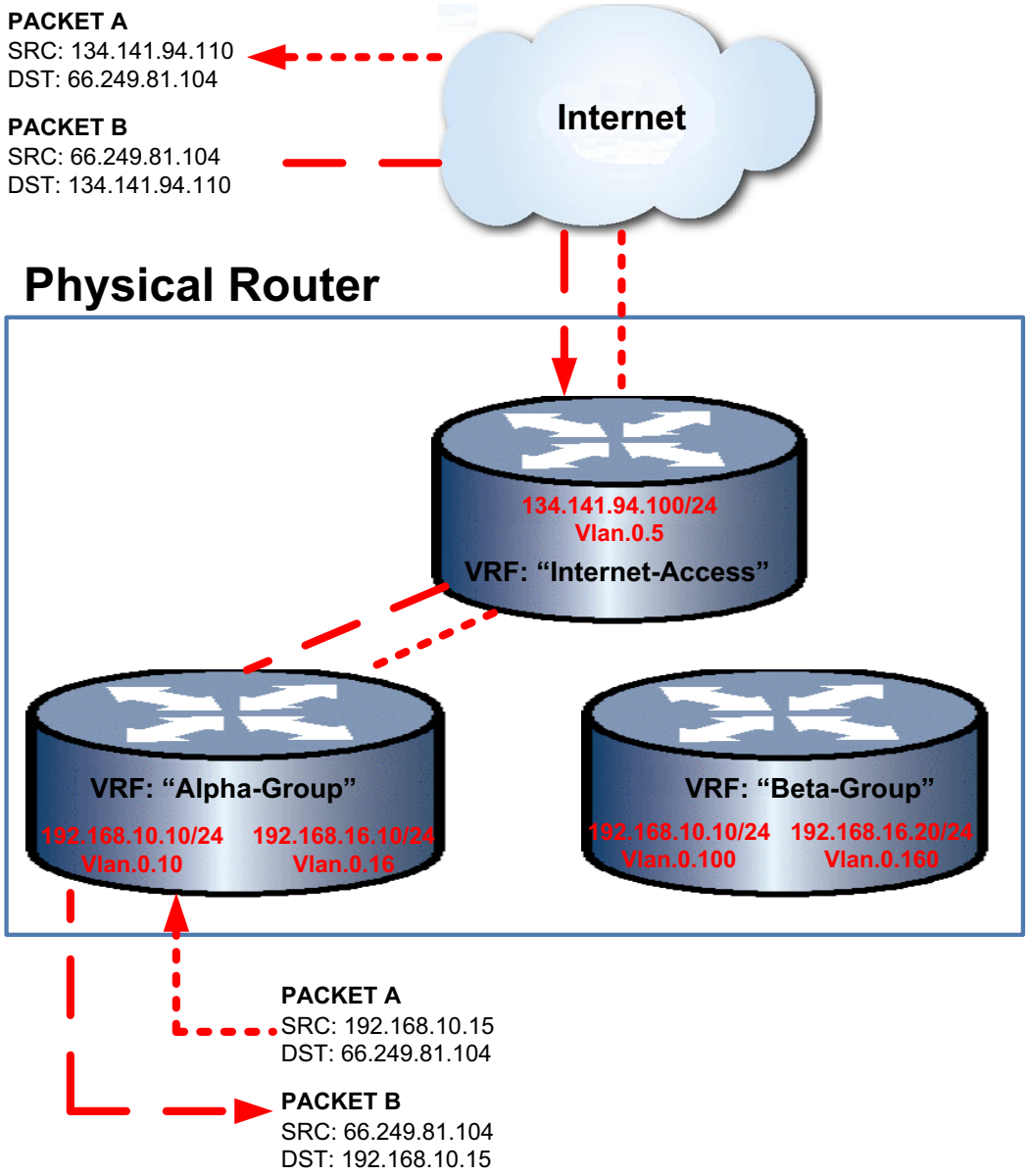
Given the configuration in Figure 35-2, there is a conflict between VRFs Alpha-Group and Beta-Group for any packet sourced outside of the system that needs to be routed to the correct VRF through the shared-access-VRF Internet-Access.

There would be no problem if VRF Alpha-Group or Beta-Group were:

- Completely isolated networks that never needed to access other networks
- Configured with another non-overlapping interface that provided access to VRF Internet-Access

Because VRF Internet-Access is used as the shared access resource out of the router for both VRF Alpha-Group and Beta-Group, a means of masking the conflicting networks is required. These conflicting networks can be masked using the NAT-inside-VRF feature. NAT-inside-VRF is a means of letting the outside NAT configuration know which VRF the inside NAT configuration belongs to. NAT-inside-VRF can be configured for both static or dynamic inside NAT.

Figure 35-2 NAT-Inside-VRF Configuration for Overlapping IP Networks



Static NAT-Inside-VRF Configuration

To configure static NAT-inside-VRF for this discussion:

1. On VRF Alpha-Group, configure interface VLAN 10, IP address 192.168.10.1/24 for IP NAT inside using the **ip nat inside** command in interface configuration mode. This assures that any packet with a source IP address of 192.168.10.1/24 will be considered for network address translation on this system.

2. On VRF Internet-Access, configure interface VLAN 5, IP address 134.141.94.100/24 for IP NAT outside using the **ip nat outside** command in interface configuration mode. This assures that any packet egressing the system on IP subnet 134.141.94.100/24 will be considered for network address translation.
3. On VRF Internet-Access, configure the NAT static rule specifying 192.168.10.15 (VLAN 10) as the inside source address and 134.141.94.1 (VLAN 5) as the outside source address, and VRF Alpha-Group as the inside VRF. This assures that any packet that has been considered for network address translation, with an IP source address of 192.168.10.15 on an interface configured for NAT inside, and belongs to VRF Alpha-Group will be NATed. The IP source address will be changed to 134.141.94.110.

Packet A is received on VLAN 10, IP address 192.168.10.15. The VRF Alpha-Group routing table determines that 134.141.94.110 on VLAN 5 is the next hop for this route. Because the receive interface is configured for inside NAT and the destination interface is configured for outside NAT, the NAT process considers Packet A for network address translation.

The static rule **ip nat inside source static 192.168.10.15 134.141.94.110 inside-vrf Alpha-Group** results in the source address for Packet A being changed from 192.168.10.15 to 134.141.94.110 and is routed to the next hop router out interface VLAN 5.

When Packet B from IP source address 66.249.81.104 is received on IP interface 134.141.94.100, because the receiving interface is configured as NAT outside, the interface is checked against NAT global addresses, and the IP destination for packet B is changed to its original source IP address: 192.168.10.15.

```
S Chassis(su)->router Alpha-Group
S Chassis(su-*ha-Group)->configure
S Chassis(su-*ha-Group-config)->interface vlan 10
S Chassis(su-*ha-Group-config-intf-vlan.0.10)->ip address 192.168.10.1/24
S Chassis(su-*ha-Group-config-intf-vlan.0.10)->ip nat inside
S Chassis(su-*ha-Group-config-intf-vlan.0.10)->exit
S Chassis(su-*ha-Group-config)->exit
S Chassis(su-*ha-Group)->exit
S Chassis(su)->router Internet-Access
S Chassis(su-*t-Access)->configure
S Chassis(su-*t-Access-config)->interface vlan 5
S Chassis(su-*t-Access-config-intf-vlan.0.5)->ip address 134.141.94.100/24
S Chassis(su-*t-Access-config-intf-vlan.0.5)->ip nat outside
S Chassis(su-*t-Access-config-intf-vlan.0.5)->exit
S Chassis(su-*t-Access-config)->ip nat inside source static 192.168.10.15
134.141.94.110 inside-vrf Alpha-Group
```

Dynamic NAT-Inside-VRF Configuration

To configure dynamic NAT-inside-VRF for this discussion:

1. On VRF Alpha-Group, configure interface VLAN 10, IP address 192.168.10.1/24 for IP NAT inside using the **ip nat inside** command in interface configuration mode. This assures that any packet from the IP subnet 192.168.10.1/24 will be considered for network address translation on this system.
2. On VRF Internet-Access, configure interface VLAN 5, IP address 134.141.94.100/24 for IP NAT outside using the **ip nat outside** command in interface configuration mode. This assures that any packet egressing the system on any member of IP subnet 134.141.94.100/24 will be considered for network address translation.

3. On VRF Internet-Access, configure a standard access-list named **dynamic-nat** with a permit host 192.168.10.15 entry.
4. On VRF Internet-Access, configure an IP NAT pool named **internet-out** containing outside address range 134.141.94.121 to 134.141.94.129.
5. On VRF Internet-Access, configure an IP NAT inside source list with the inside access-list **dynamic-nat** and outside address pool **internet-out**, specifying Alpha-Group as the inside VRF.

Packet A is received on VLAN 10, IP address 192.168.10.15. The VRF Alpha-Group routing table determines that 134.141.94.104 on VLAN 5 is the next hop for this route. Because the receive interface is configured for inside NAT and the destination interface is configured for outside NAT, the NAT process considers Packet A for network address translation.

The inside source list, configured in Step 5 above, assures that any packet being considered for network address translation, with an IP source address matching a **dynamic-nat** access-list permit clause, received on an interface configured for NAT inside, and belonging to VRF **Alpha-Group,** will be NATed. In this case, the IP source address will will be changed to a dynamically selected address from NAT pool **internet-out**.

When Packet B from IP source address 66.249.81.104 is received on IP interface 134.141.94.100, because the receiving interface is configured as NAT outside, the interface is checked against NAT global addresses, and the IP destination for packet B is changed to its original source IP address: 192.168.10.15.

```
S Chassis(su)->router Alpha-Group
S Chassis(su-*ha-Group)->configure
S Chassis(su-*ha-Group-config)->interface vlan 10
S Chassis(su-*ha-Group-config-intf-vlan.0.10)->ip address 192.168.10.1/24
S Chassis(su-*ha-Group-config-intf-vlan.0.10)->ip nat inside
S Chassis(su-*ha-Group-config-intf-vlan.0.10)->exit
S Chassis(su-*ha-Group-config)->exit
S Chassis(su-*ha-Group)->exit
S Chassis(su)->router Internet-Access
S Chassis(su-*t-Access)->configure
S Chassis(su-*t-Access-config)->interface vlan 5
S Chassis(su-*t-Access-config-intf-vlan.0.5)->ip address 134.141.94.100/24
S Chassis(su-*t-Access-config-intf-vlan.0.5)->ip nat outside
S Chassis(su-*t-Access-config-intf-vlan.0.5)->exit
S Chassis(su-*t-Access-config)->ip access-list standard dynamic-nat
S Chassis(su-*t-Access-cfg-std-acl-dyna*-nat)->permit host 192.168.10.15
S Chassis(su-*t-Access-cfg-std-acl-dyna*-nat)->exit
S Chassis(su-*t-Access-config)->ip nat pool internet-out 134.141.94.121
134.141.94.129
S Chassis(su-*t-Access-config)->ip nat inside source list dynamic-nat pool
internet-out inside-vrf Alpha-Group
```

Server Load Balancing (SLB) Services Between VRFs

SLB is the process by which a service is provided by a proxy device for a set of real servers (the actual server devices) that implement the service. The proxy device load balances the service by distributing the service between itself and the real servers. LSNAT provides SLB services on the

S-Series platforms. An SLB configuration consists of a virtual server, acting as the proxy device, and a server-farm made up of one or more real servers.

The virtual server configuration specifies:

- A Virtual IP address (VIP)
- Either a UDP or TCP port number to listen for client requests on
- A server-farm from which a real server is selected to handle a client request

The server-farm configuration specifies:

- A list of real servers
- A load balancing method

The virtual server selects a real server to handle a client request for a service.

SLB services can be configured on a single VRF and shared with multiple non-SLB configured VRFs, by specifying the **all-vrfs** parameter when configuring the virtual server.

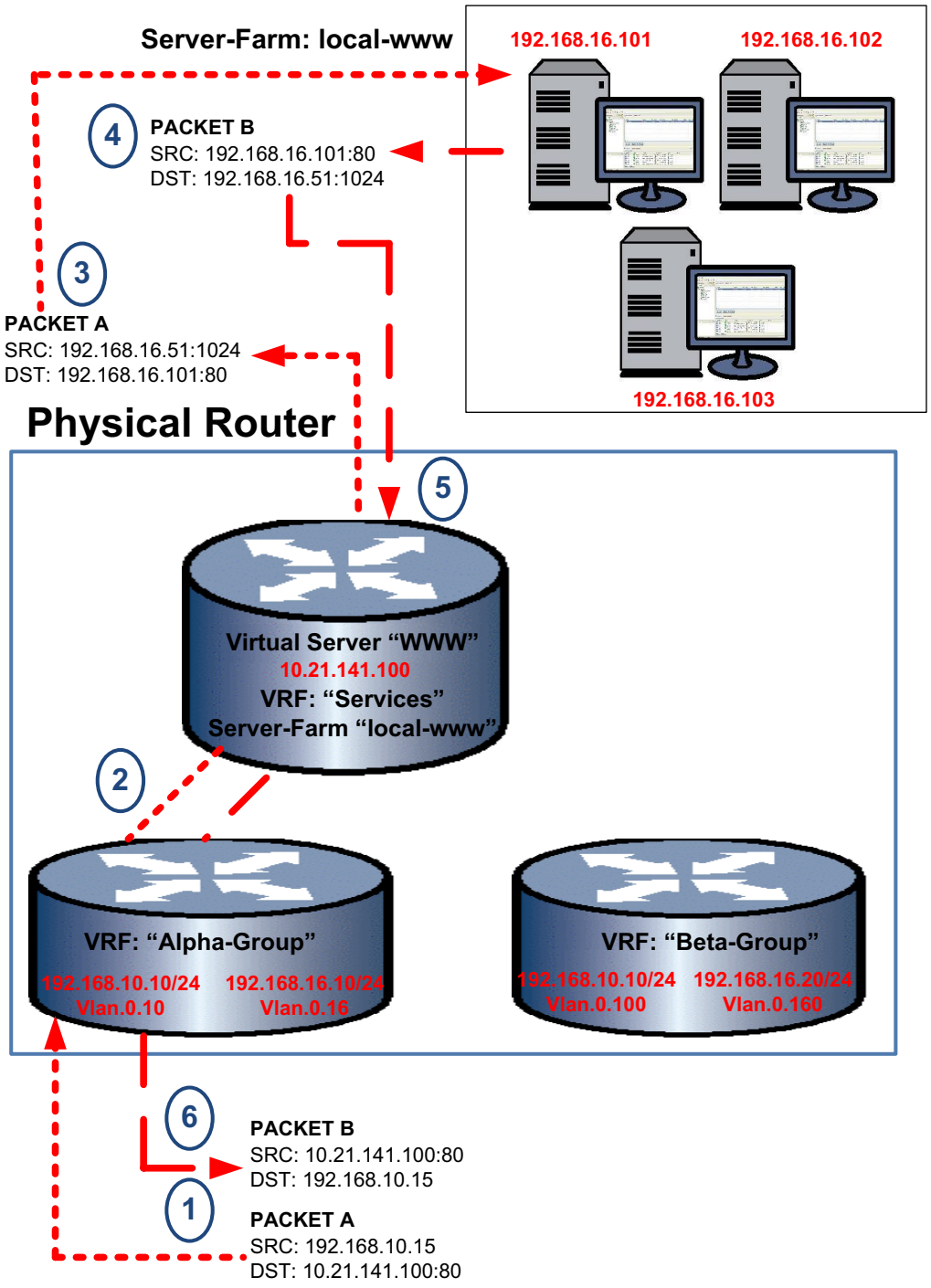
Figure 35-3 on page 35-11 presents an example of an SLB all-VRFs configuration. The packet processing and flow for this example is as follows:

1. Packet A ingresses the router on VLAN 10, IP address 192.168.10.15 of VRF Alpha-Group. Packet A's destination is the virtual server 10.21.141.100 which is configured for all-VRF on VRF Services.
2. VRFs Alpha-Group and Beta-Group contain overlapping IP networks. See "VRFs With Overlapping IP Networks" on page 35-5 for a full explanation of how overlapping IP networks are handled in a VRF environment. VRF Services is configured with the "local-net" source NAT pool with an address range 192.168.16.51 through 192.168.16.55. VRF Services performs Network Address Translation (NAT) on Packet A. An SLB binding is created, selecting the new addresses from the "local-net" pool. The SLB binding stores both sets of addresses that make up the network address translation.
3. Packet A is forwarded to the selected real server by VRF Services.
4. The real server responds with Packet B. The source address for Packet B is the real server. The destination address for Packet B is the NATed address on VRF Services.
5. On VRF Services, Packet B's source address is changed to the pre-NATed virtual server address 10.21.141.100 and the destination address is changed to the pre-NATed VRF Alpha-Group address 192.168.10.15.
6. Packet B is forwarded to VRF Alpha-Group.

```
S Chassis(su)->router Services
S Chassis(su-Services)->configure
S Chassis(su-Services-config)->ip nat pool local-net 192.168.16.51 192.168.16.55
S Chassis(su-Services-config)->ip slb serverfarm local-www
S Chassis(su-Services-config-slb-sfarm)->real 192.168.16.101
S Chassis(su-Services-config-slb-real)->inservice
S Chassis(su-Services-config-slb-real)->exit
S Chassis(su-Services-config-slb-sfarm)->real 192.168.16.102
S Chassis(su-Services-config-slb-real)->inservice
S Chassis(su-Services-config-slb-real)->exit
S Chassis(su-Services-config-slb-sfarm)->real 192.168.16.103
S Chassis(su-Services-config-slb-real)->inservice
S Chassis(su-Services-config-slb-real)->exit
```

```
S Chassis(su-Services-config-slb-sfarm)->exit
S Chassis(su-Services-config)->ip slb vserver WWW
S Chassis(su-Services-config-slb-vserver)->virtual 10.21.141.100 tcp www all-vrfs
S Chassis(su-Services-config-slb-vserver)->serverfarm local-www
S Chassis(su-Services-config-slb-vserver)->source nat pool local-net
S Chassis(su-Services-config-slb-vserver)->inservice
S Chassis(su-Services-config-slb-vserver)->exit
S Chassis(su-Services-config)->
```

Figure 35-3 Sharing SLB Services With Multiple VRFs



Forwarding Local UDP Broadcasts To A Different VRF

When enabling DHCP/BOOTP relay and forwarding local UDP broadcasts to a new destination address that is located on a different VRF or the global router, the destination VRF or the global router must be specified in the **ip helper-address** command. The **vrf** *vrf-name* and **global** parameters have been added to the the **ip helper-address** command.

When forwarding the local UDP broadcasts from a VRF to a destination address on the global router or a different VRF, the DHCP relay agent must include information about itself in order for the DHCP server to determine which pool of client addresses to pull the lease from. Including Option 82 in the DHCP relay information provides the required DHCP relay information.

Use the **ip dhcp relay information option vpn** command to include DHCP relay agent information in the packet sent to the DHCP server by the client.

The following example:

- Enables IP forwarding for the UPD protocol on VRF Alpha-Group
- Enables DHCP/BOOTP relay on VLAN 10 of VRF Alpha-Group and sets the new destination address to 134.141.95.105 on VRF Internet-Access
- Configures the inclusion of DHCP relay agent information in the packet sent to the DHCP server by the client

```
S Chassis(su)->router Alpha-Group
S Chassis(su-*ha-Group)->configure
S Chassis(su-*ha-Group-config)->ip forward-protocol udp
S Chassis(su-*ha-Group-config)->interface vlan.0.10
S Chassis(su-*ha-Group-config-intf-vlan.0.10)->ip helper-address 134.141.95.105
vrf Internet-Access
S Chassis(su-*ha-Group-config-intf-vlan.0.10)->exit
S Chassis(su-*ha-Group-config)->ip dhcp relay information option vpn
S Chassis(su-*ha-Group-config)->
```

Configuring VRF

This section provides details for the configuration of VRF on the S-Series products.

Table 35-1 lists VRF parameter default values.

Table 35-1 Default VRF Parameters

Parameter	Description	Default Value
SNMPv3 context Name	The name that SNMPv3 will associate with this VRF.	VRF Name
router context	The VRF router command mode context if no router is specified	global

Procedure 35-1 describes how to configure VRF.

Procedure 35-1 VRF Configuration

Step	Task	Command(s)
1.	Create the VRF, in any configuration mode, optionally specifying an SNMPv3 context name.	**set router vrf create** *vrf-name* [**context** *context-name*]
2.	Enter router mode for the VRF to be configured.	**router** [*name*]
3.	Enter configuration mode for this VRF router instance.	**configure**

Procedure 35-1 VRF Configuration (continued)

Step	Task	Command(s)
4.	Optionally, configure static routes to perform next hop lookup on the egress VRF for any route that the egress interface is on a different VRF instance. A layer 3 tunnel interface is currently only supported on the Global VRF.	**ip route** {*prefix mask* | *prefix*/*prefix-length*} {*ip-address* [**recursive**] | **interface** *interface-name* | **vlan** *vlan-id* | **vrf** *egress-vrf* | **blackhole** | **reject**} [*distance*] [**tag** *tag-id*] or **ipv6 route** *prefix/length* {*ipv6-address* [**recursive**] | [**interface** *interface-name*] | **vlan** *vlan-id* | **vrf** *egress-vrf* | **blackhole** | **reject**} [*distance*] [**tag** *tag-id*]
5.	Optionally, when creating a policy route map, with a match IP address policy in which the interface belongs to a different VRF, configure the next hop VRF to perform the route lookup using its routing table.	**set vrf** *vrf-name*
6.	Optionally, when multiple VRFs contain overlapping IP networks that communicate to the outside internet, use the NAT-inside-VRF feature to differentiate the VRFs containing the overlapping IP networks.	**ip nat inside source static** *local-ip global-ip* [**inside-vrf** *vrf-name*] or **ip nat inside source static** {**tcp** | **udp**} *local-ip local-port global-ip global-port* **inside-vrf** *vrf-name*
7.	Optionally, when a VRF provides LSNAT SLB services to one or more non-SLB configured VRFs, configure the virtual server or a range of virtual servers of the SLB configured VRF with the all-VRFs feature	**virtual** *ip-address* {**tcp** | **udp**} *port* [**service** *service-name*] [**all-vrfs**] **virtual-range** *start-address end-address* {**tcp** | **udp**} *port* [**service** *service-name*] [**all-vrfs**]
8.	Optionally, in interface configuration mode, when forwarding local UDP broadcasts to a new destination address, on a different VRF, specify the destination VRF using the **vrf** parameter. In addition, in VRF configuration mode, specify that option 82 information be included in packets sent to the DHCP server by the client.	**ip helper-address** *destination-address* [**global**] [**vrf** *vrf-name*] **ip dhcp relay information option vpn**

Terms and Definitions

Table 35-2 lists terms and definitions used in this VRF configuration discussion.

Table 35-2 VRF Configuration Terms and Definitions

Term	Definition
all-VRFs	An LSNAT feature which allows the SLB virtual server on a VRF to provide SLB services to all other VRFs on the router.
egress VRF	Within a static route context, specifies the egress VRF for next hop lookup when different from a route's ingress VRF.
global router	The default router for the physical router. Also responsible for managing VRFs configured on the physical router.
NAT-inside-VRF	A NAT feature that identifies the appropriate VRF context to use within a static or dynamic inside source NAT configuration.

Table 35-2 VRF Configuration Terms and Definitions (continued)

Term	Definition
shared-access-VRF	A VRF that provides access to the outside internet to one or more other VRFs in the system.
SNMPv3 context	Specifies the SNMPv3 context name to be used by SNMP for a given VRF instance.
Virtual Routing and Forwarding (VRF)	A method of partitioning a global router network into different routed domains.
VRF instance	A segregated routing domain for the routed forwarding of packets managed by the global router.

36

IP Routing Configuration

This document describes IPv4 and IPv6 routing configuration on Extreme Networks S-Series devices.

The Router

The current firmware implementation supports a single default Virtual Routing and Forwarding (VRF) router named **global** and up to 128 VRF instances, depending upon your system. See Chapter 35, **Virtual Routing and Forwarding (VRF) Configuration** for VRF feature and configuration details.

There are two ways of accessing the **global** VRF router configuration:

- Directly from global configuration mode, accessed by entering the **configure** command from the system command mode
- First entering router command mode from system command mode using the **router** command, specifying **global** as the name of the router, followed by entering the **configure** command to gain access to the router configuration mode

To enter a non-global VRF router instance, use the **router** command, specifying the name of the VRF instance to configure, followed by entering the **configure** command to gain access to the router configuration mode for that VRF instance.

Once in either router configuration or global configuration command mode, the same set of router configuration commands are available to you.

Use the **clear router vrf** command to clear the routing configuration for the **global** router or the specified VRF router instance on the device. This is a very powerful command that should only be

used if you intend to completely clear all router and interface configuration for the specified VRF router. Unless attached via a direct console connection, loss of management connectivity to the VRF router should be expected after using the **clear router vrf** command.

Entering Router Configuration

To enter the **global** VRF router configuration context from the system command mode, and verify the current router context, enter:

```
S Chassis(rw)->configure
S Chassis(rw--config)->show router
Router Services are currently running on module 1.
VRF Context    : global
RD             : not set
S Chassis(rw-router-config)->
```

To enter the **global** VRF router configuration context from the router configuration command mode, and verify the current router context, enter:

```
S Chassis(rw)->router global
S Chassis(rw-router)->configure
S Chassis(rw-router-config)->show router
Router Services are currently running on module 1.
VRF Context    : global
RD             : not set
S Chassis(rw-router-config)->
```

Table 36-1 describes how to enter router configuration mode.

Table 36-1 Entering Router Configuration Mode

Task	Command(s)
In system configuration mode, enter router command mode for the specified router. Supported routers: **global** or a named VRF created using the **set router vrf create** command. To enter router configuration command mode for the **global** or named VRF router, use the **configure** command in router command mode. The **global** router can also be configured in global configuration command mode.	**router** [*name*]

Display Router Configuration

Use the **show router** command in any command mode to display router settings for the current VRF context.

Use the **show limits** command in any command mode to display application limits associated with the current VRF context. Use the **show limits vrf** command to display the limits for a named VRF. Use the **show limits application** command to display the limits for a specified application in the current VRF context. Use the show limits resource-ipv6netmask command to display the IPv6 netmask setting. The following example displays a sample output of the **show limits** command:

```
S Chassis(su)->show limits
```

```
Chassis limits:
Application                          Limit    In use   Entry size  Total Memory
-------------------------------- --------- --------- ------------ ------------
access-lists                          1000         0         6.2K           6M
  access-list-entries                 5000         0         160B       781.6K
  access-list-entries-per-list        5000         -            -           -
  applied-access-lists                4096         0         152B       152.1K
    applied-ipv4-in                   1024         0            -           -
    applied-ipv4-out                  1024         0            -           -
    applied-ipv6-in                   1024         0            -           -
    applied-ipv6-out                  1024         0            -           -
appsvc-ftp-alg-entries                8000         0          40B       312.5K
appsvc-global-bindings               65536         0         104B         6.5M
.
.
.
Total Memory                             -         -            -       529.7M


S Chassis(su)->
```

Use the **show running-config** command to display non-default router configuration for either all or a specified option. When specifying **all**, both default and non-default configuration displays. Additional options are available for the display of a subset of the running configuration by feature or protocol. Enter the **show running-config ?** command for a listing of the additional options. The following example displays a sample output of the **show running-config** command:

```
S Chassis(su)->show running-config
# **** Global Router Configuration ****
configure terminal
!
 interface vlan.0.1
  ip address 100.10.10.10 255.0.0.0 primary
  ip dvmrp
  no shutdown
  exit
 interface vlan.0.56
.
.
.
S Chassis(su)->
```

The Routing Interface

Routing interfaces are configured by entering the **interface** command from the configuration command mode, specifying the interface ID and whether the interface is a VLAN or a loopback interface. If the interface has not previously been created, this command creates a new routing interface.

A VLAN routing interface can be configured before its VLAN is created in system configuration mode, but VLANs must be created from the system CLI before they will be operational within IP routing. See "Configuring VLANs" on page 24-9 for VLAN configuration details.

Each VLAN or loopback interface must be configured for routing separately using the **interface** command. To end configuration on one interface before configuring another, type **exit** at the command prompt. Enabling the interface for IP routing is required using the **no shutdown** command before exiting the interface mode.

IPv4 forwarding is enabled by default on a routing interface. Use the **no ip forwarding** command within interface configuration command mode to disable IPv4 forwarding on a routing interface.

IPv6 forwarding is disabled by default on a routing interface.

Note: IPv4 and IPv6 forwarding are both enabled by default on loopback interfaces. Without forwarding, a loopback interface is unreachable. This configuration setting cannot be modified.

IP Routing Addresses

IPv4 Interface Address

A single primary network IPv4 address is configurable on an interface. Up to 100 secondary network IPv4 addresses are configurable. The first network IP address assigned to an interface is the primary whether explicitly configured as primary or not. To configure a secondary network IP address on an interface, the address must be explicitly configured as secondary, otherwise you will be queried as to whether you want to overwrite the current primary.

In the following example the IP address is set to **99.0.0.1/24**. This setting is followed by an attempt to configure **99.0.0.2/16** as a secondary address, while failing to specify the **secondary** keyword. When queried as to whether the primary IP address should be changed, **n** is entered. The **secondary** keyword is added on the next line. The **show running-config** command output confirms the configuration:

```
S Chassis(rw-config-intf-vlan.0.99)->ip address 99.0.0.1/24
S Chassis(rw-config-intf-vlan.0.99)->ip address 99.0.0.2/16
Do you want to replace primary IP address (y/n) [n]?n
S Chassis(rw-config-intf-vlan.0.99)->ip address 99.0.0.2/16 secondary
S Chassis(rw-config-intf-vlan.0.99)->show running-config interface vlan.0.99
# **** VRF default (default) ****
configure terminal
!
 interface vlan.0.99
  ip address 99.0.0.1 255.255.255.0 primary
  ip address 99.0.0.2 255.255.0.0 secondary
```

```
  exit
!
exit
S Chassis(rw-config-intf-vlan.0.99)->
```

The **ip address** command in interface configuration command mode is used to assign IP networks as primary or secondary to a routing interface.

See "IPv6 Interface Address" on page 36-5 for IPv6 address configuration information.

The **no ip address** command removes the specified IPv4 address configuration for this interface.

IPv4 Router Interface Configuration Example

The following example:

- Creates the interface for VLAN 1
- Configures a primary IP address of 10.21.130.59 255.255.128.0

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 1
S Chassis(rw-config-intf-vlan.0.1)->ip address 10.21.130.59 255.255.128.0
S Chassis(rw-config-intf-vlan.0.1)->no shutdown
S Chassis(rw-config-intf-vlan.0.1)->exit
S Chassis(rw-config)->
```

See the current firmware release notes for the number of routing interfaces supported on an S-Series routing module. Each interface can be configured for the RIP, BGP and/or OSPF routing protocols.

A primary IP address must be configured on each routing interface. Secondary IP addresses can optionally be configured. See the current firmware release notes for the number of secondary addresses supported on an interface and module. Use the **ip address** command in interface configuration command mode to assign an IP address and optional secondary IP addresses to an interface, specifying whether the assigned address is primary or secondary.

S-Series routing interfaces support Equal Cost Multipath (ECM). ECM is a routing technique for routing packets along multiple paths of equal cost. Two algorithms are available for ECM routing:

- **Hash threshold** — Path selection is based upon a firmware generated hash. This is the default algorithm
- **Round robin** — Path selection is based upon a simple round robin algorithm

Use the **ip ecm-forwarding-algo** command to set the ECM forwarding algorithm for this S-Series device. ECM forwarding uses the hash threshold algorithm by default.

IPv6 Interface Address

One or more unicast IPv6 addresses and a single link local address can be configured for an interface using the **ipv6 address** command in interface configuration mode.

Link local addresses are network addresses which are intended only for communications within one segment of a local network (a link) or a point-to-point connection. They allow addressing hosts without using a globally-routable address prefix. Routers will not forward packets with link-local addresses. A link local address must begin with **fe80:**.

An interface can be configured to have its IPv6 address auto acquired using the **autoconfig** option.

A single link local address is supported per interface. If IPv6 autoconfiguration is enabled, the link local address is autoconfigured. When manually configuring a link local address, if a link local address already exists on the interface, a warning displays asking you if you wish to change it.

EUI-64 is an IPv6 address automatic interface addressing capability. By implementing the IEEE's 64-bit Extended Unique Identifier (EUI-64) format, a host can automatically assign itself a unique 64-bit IPv6 interface identifier without the need for manual configuration or DHCP. This is accomplished on Ethernet interfaces by referencing the already unique 48-bit MAC address and reformatting that value to match the EUI-64 specification as specified in RFC 2373. When configuring an EUI-64 address, the specified prefix must have a length of 64.

A general prefix allows an assigned name to represent a network prefix from which longer IPv6 addresses can be configured. The sub-bits added to the general prefix can both extend the network prefix by adding to the specified prefix length, as well as complete the IPv6 address.

Use the **ipv6 general-prefix** command to configure a general prefix. See "IPv6 General Prefix" on page 36-6 for general prefix details.

Use the show ipv6 interface command to display IPv6 addresses assigned by the **ipv6 address** command.

See "IPv4 Interface Address" on page 36-4 for IPv4 address configuration information.

The **no ipv6 address** command removes the specified IPv6 address configuration for this interface.

IPv6 General Prefix

The general prefix is an ease of use feature that allows an assigned name to represent a network prefix from which longer IPv6 addresses can be configured. Network renumbering is simplified by redefining the general prefix, thereby changing the portion of addresses to which the general prefix is assigned.

When using a general prefix to configure an IPv6 address, you can extend the network prefix by adding to the length specified in the **ipv6 address** command.

Deleting a general prefix does not delete the underlying addresses defined by the general prefix. Any IPv6 addresses based upon the general prefix remain. Use the **no ipv6 address** command to remove the IPv6 address.

The S-Series supports the configuration of up to 64 general prefixes on a system.

The following example creates a general prefix named "Doc-Prefix" with a prefix value of **2001:11ac:fd34::/48** and assigns the IPv6 address **2001:11ac:fd34:50:0:0:abcd:33** to VLAN 51. The general prefix **Doc-Prefix** is followed by **::50:0:0:abcd:33/64**. The subnet length is changed to **/64** adding **:50** to the general prefix to create a network prefix of **2001:11ac:fd34:50/64** for this IPv6 address:

```
S Chassis(su)->configure
S Chassis(su-config)->ipv6 general-prefix Doc-Prefix 2001:11ac:fd34::/48
S Chassis(su-config)->show ipv6 general-prefix
  ipv6 general-prefix Doc-Prefix 2001:11ac:fd34::/48
S Chassis(su-config)->interface vlan 51
S Chassis(su-config-intf-vlan.0.51)->ipv6 address Doc-Prefix ::50:0:0:abcd:33/64
S Chassis(su-config-intf-vlan.0.51)->show ipv6 interface vlan.0.51

 vlan.0.51 is Operationally down, Administratively down
  IPv6 is enabled link-local address is fe80::211:88ff:fe7c:32c1%vlan.0.51
```

```
  Global unicast address(es):
    2001:11ac:fd34:50:0:0:abcd:33, subnet is 2001:11ac:fd34:50::/64
...
S Chassis(su-config-intf-vlan.0.51)->
```

IPv6 Router Interface Configuration Examples

This example sets the IPv6 address for interface VLAN 50 to ba10:1100:aa11:c171:0:0:1111:00/48:

```
S Chassis(su-config)->interface vlan 50
S Chassis(su-config-intf-vlan.0.50)->ipv6 address
ba10:1100:aa11:c171:0:0:1111:00/48
S Chassis(su-config-intf-vlan.0.50)->
```

This example sets the IPv6 link local address for interface VLAN 50 to fe80:1234:5678::300:

```
S Chassis(su-config)->interface vlan 50
SChassis(su-config-intf-vlan.0.50)->ipv6 address fe80:1234:5678::300 link-local
Do you want to replace IPv6 link-local address (y/n) [n]?y
S Chassis(su-config-intf-vlan.0.50)->
```

This example sets an IPv6 EUI-64 address for interface VLAN 50 based upon the prefix 2001:febd:1234:0/64, and displays the EUI-64 address in the interface output:

```
S Chassis(su-config)->interface vlan 50
S Chassis(su-config-intf-vlan.0.50)->ipv6 address 2001:febd:1234:0/64 eui-64
S Chassis(su-config-intf-vlan.0.50)->show ipv6 interface vlan.0.50
 vlan.0.50 is Operationally down, Administratively down
  IPv6 is enabled link-local address is fe80::2e0:63ff:fe6b:1d26%vlan.0.50
  Global unicast address(es):
    2001:febd:1234::2e0:63ff:fe6b:1d26, subnet is 2001:febd:1234::/64 [EUI]
...
S Chassis(su-config-intf-vlan.0.50)->
```

Secondary and Private VLAN

The secondary VLAN configuration on an IP Interface provides the ability to associate multiple L2 VLANs with one L3 IP interface. The secondary VLAN feature provides for the configuration of private VLANs by configuring ports on the secondary VLAN as private members. A secondary VLAN port can be configured as a private VLAN member by restricting the ports they can egress to using the **set vlan egress** command. Members of the private VLAN are connected hosts that share the IP interface of the primary VLAN, while at the same time are restricted from directly communicating with each other. Hosts on the primary VLAN, also referred to as the community VLAN, can communicate directly with hosts on both the primary and private VLANs.

When configuring members of the private VLAN, set both the secondary and primary VLAN constraint to shared, using the same constraint set ID. This setting assures that both the primary and secondary VLAN use the same FID. VLAN constraint is set using the **set vlan constraint** command.

Set ports on the primary VLAN as members of the egress list for all ports on both the primary and secondary VLANs. Set private member ports on the secondary VLAN as members of the egress list for all members of the primary VLAN. Use the **set vlan egress** command to set ports as members of a VLAN's egress list.

The secondary VLAN is not configured as an independent routing interface; it is configured within the primary VLAN. Only set an IP address for the primary VLAN interface. Do not set an IP address for the secondary VLAN.

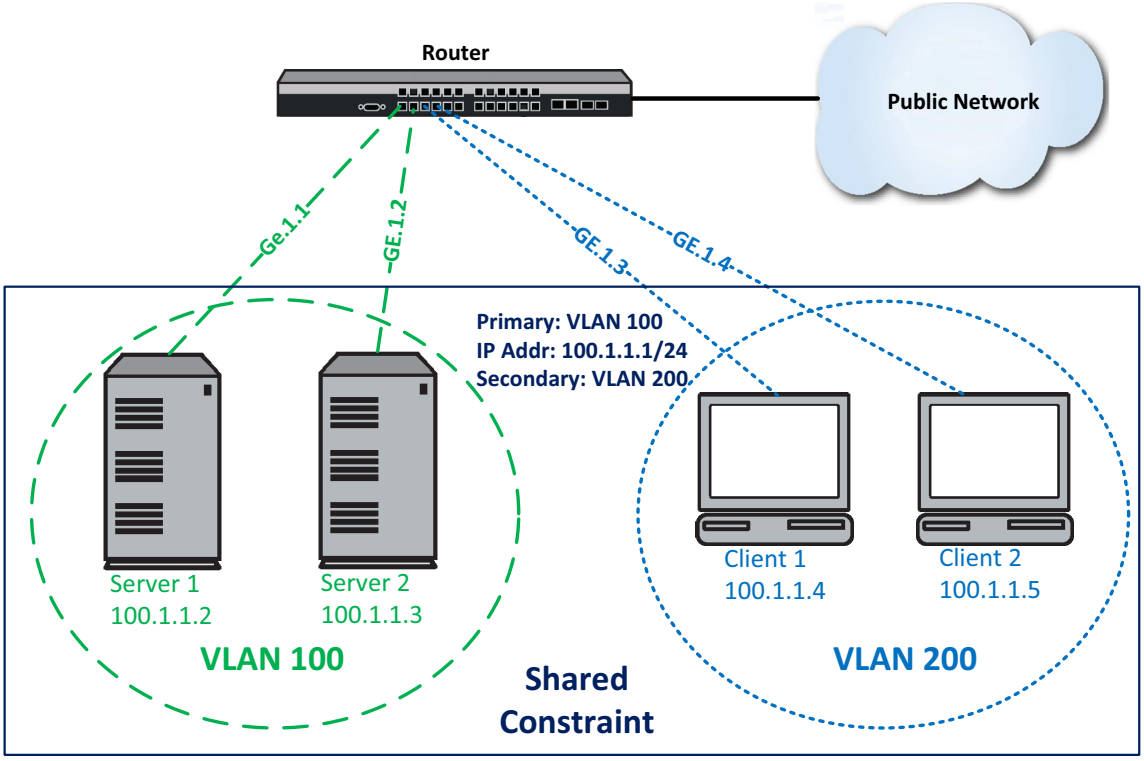
Refer to Chapter 24, **VLAN Configuration** for VLAN configuration details.

This feature could be used by an internet service provider network where clients should not be directly communicating with other clients on the same network unless permitted to do so. These restricted clients would be assigned to the secondary VLAN.

Private VLAN Configuration Example

Figure 36-1 displays a private VLAN configuration example. VLAN 100 and VLAN 200 are VLANs configured on the 100.1.1.1/24 network. In this example VLAN 100 is the primary VLAN with members Server 1 and Server 2. VLAN 200 is the secondary VLAN with members Client 1 and Client 2. Primary VLAN members are configured on ports ge.1.1-2 and are members of the egress list for all ports on both VLAN 100 and VLAN 200. Private VLAN members are configured on secondary VLAN ports ge.1.3-4 with an egress only on VLAN 100 ports. Both the primary VLAN and the secondary VLAN are configured with the same constraint set ID of 100 which means they share the same filtering database (FID 100). The routing interface is VLAN 100. The secondary VLAN is configured within the routing interface VLAN 100 configuration mode.

Figure 36-1 Secondary VLAN Configuration



To configure this example:

- Create the static primary (VLAN **100**) and secondary (VLAN **200**) VLANs
- Assign ports **ge.1.1-2** to the primary VLAN
- Assign ports **ge.1.3-4** to the secondary VLAN
- Configure VLAN 200 as a private VLAN by:

  - Setting egress for VLAN 100 for all ports
  - Setting egress for VLAN 200 only on primary VLAN ports ge.1.1-2
- Set the VLAN constraint to shared for each VLAN with a constraint set ID of 100
- Configure the primary interface with a primary IP address of **100.1.1.1/24** and a secondary VLAN of **200**

```
S Chassis(rw)->set vlan name 100 PrimaryVlan
S Chassis(rw)->set vlan name 200 SecondaryVlan
S Chassis(rw)->set port vlan ge.1.1-2 100
S Chassis(rw)->set port vlan ge.1.3-4 200
S Chassis(rw)->set vlan egress 100 ge.1.1-4 untagged
S Chassis(rw)->set vlan egress 200 ge.1.1-2 untagged
S Chassis(rw)->set vlan constraint 100 100 shared
S Chassis(rw)->set vlan constraint 200 100 shared
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 100
S Chassis(rw-config-intf-vlan.0.100)->ip address 100.1.1.1/24 primary
S Chassis(rw-config-intf-vlan.0.100)->secondary-vlan 200
```

Non-Forwarding IP Management Interfaces

Multiple IP interface configuration provides the ability to assign a unique IP address to each non-routing interface on the switch. The ability to set a unique IP address on each VLAN configured on the switch means that host management can be accessed from any VLAN configured with its own IP address.

The ability to assign an IP subnet to an interface that is separate from the subnet that is passing data through the switch, allows the network administrator to create an out-of-band management subnet designed to only pass network management data.

Note: All interfaces can be configured as either a routing interface or a non-forwarding IP interface. It is recommended that you only use the non-routing multiple IP interface feature on a non-routing switch: a switch that does not have any routing capability turned on and is not directly connected to a router.

A non-routing host management IP interface can now be configured:

- In interface configuration command mode using the **interface** command
- In any command mode using an enhanced **set ip address** command

When configuring the non-routing host management IP interface in interface configuration command mode you must explicitly set the interface as a non-forwarding interface using the **no ip forwarding** command for IPv4 forwarding. IPv6 forwarding is disabled by default. On an IPv4 interface, you must disable IP Proxy ARP using the **no ip proxy-arp** command.

When configuring a non-routing host management IPv4 and IPv6 interfaces in any command mode, use the **set ip address** command. The IP address is assigned to the specified interface. The **set ip address** command automatically configures the specified interface to disable both IP forwarding and IP Proxy ARP for IPv4. IPv6 forwarding is disabled by default and IPv6 proxy is not supported. This command can only be used in a non-routing host management IP interface context.

The **set ip address** command only allows for the specifying of a primary IPv4 address or an IPv6 address. If you wish to configure a non-forwarding IP interface with secondary IP addresses, use

the **interface** command in configuration command mode to configure the interface. IPv6 addressing makes no distinction between primary and secondary addresses and treats IPv6 addresses equally.

When clearing an IPv4 or IPv6 address, the IP address to be cleared is explicitly stated. This command can be used on a primary IPv4 address or any IPv6 address. Use the **no ip address** command in interface configuration command mode to clear a secondary IP address.

Use the **clear ip interface** command to clear the IP interface the IP address is assigned to.

The following example clears the IP interface VLAN **5**:

```
S Chassis(rw)->clear ip interface vlan.0.5
```

Non-Forwarding IPv4 Management Interface Examples

The following multiple IP interface example configures VLANs 1 and 5 as non-routing host management IP interfaces in interface configuration command mode. Both interfaces are configured with IP forwarding and IP Proxy ARP disabled as follows:

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan.0.1
S Chassis(rw-config-intf-vlan.0.1)->ip address 125.50.10.1/16
S Chassis(rw-config-intf-vlan.0.1)->no ip forwarding
S Chassis(rw-config-intf-vlan.0.1)->no ip proxy-arp
S Chassis(rw-config-intf-vlan.0.1)->no shutdown
S Chassis(rw-config-intf-vlan.0.1)->exit
S Chassis(rw-config)->interface vlan.0.5
S Chassis(rw-config-intf-vlan.0.5)->ip address 125.100.10.1/16
S Chassis(rw-config-intf-vlan.0.5)->no ip forwarding
S Chassis(rw-config-intf-vlan.0.5)->no ip proxy-arp
S Chassis(rw-config-intf-vlan.0.5)->no shutdown
S Chassis(rw-config-intf-vlan.0.5)->exit
S Chassis(rw-config)->
```

The above example is replicated below using the **set ip address** command in system command mode:

```
S Chassis(rw)->set ip address 125.50.10.1 mask 255.255.0.0 interface vlan.0.1
S Chassis(rw)->set ip address 125.100.10.1 mask 255.255.0.0 interface vlan.0.5
S Chassis(rw)->
```

Non-Forwarding IPv6 Management Interface Examples

The following multiple IPv6 interface example configures VLANs 1 and 5 as non-routing host management IP interfaces in interface configuration command mode. IPv6 forwarding is disabled by default and IPv6 does not support proxy configuration:

```
S Chassis(rw-config)->interface vlan 1
S Chassis(rw-config-intf-vlan.0.1)->ipv6 address
ba10:1100:aa11:c171:0:0:1111:1/48
S Chassis(rw-config-intf-vlan.0.1)->no shutdown
S Chassis(rw-config-intf-vlan.0.1)->exit
S Chassis(rw-config)->interface vlan.0.5
```

```
S Chassis(rw-config-intf-vlan.0.5)->ipv6 address
ba10:1100:aa11:c171:0:0:111:5/48
S Chassis(rw-config-intf-vlan.0.5)->no shutdown
S Chassis(rw-config-intf-vlan.0.5)->exit
S Chassis(rw-config)->
```

Backward Compatibility Note

Firmware prior to release 7.x supported the configuration of a single non-routing host management interface using the following system level method:

- **set port vlan host.0.1** command to configure the port
- **set vlan egress** *vid* **host.0.1 untagged** to configure the VLAN
- **set ip address** command to assign the IP address to the host interface specified in the **set vlan egress** command

In release 7.x, this method is still supported for the configuration of a single non-routing host management interface.

Note: When using the legacy method of configuring a single non-routing host management interface, the **set ip address** command **interface** parameter is optional, though recommended. You must explicitly specify the interface when configuring multiple IP interfaces.

Setting a Default Host Management IP Interface

Setting the default host management interface is not supported in interface configuration command mode accessed using the **interface** command. In release 7.0, the **set ip interface** command can be entered in any command mode and provides for the optional setting of the interface as the default host management interface. The **set ip interface** command also allows for the initial configuration of a non-routing IP interface that you can assign an IP address to using the **set ip address** command.

Show Interface Examples

Use the **show interface** command to display information about one or more VLAN or loopback interfaces configured on the router.

```
S Chassis(rw-config)->show interface vlan.0.1
 vlan.0.1 is Administratively up, Operationally up
   IP Address 10.21.130.59 Mask 255.255.128.0
   MAC-Address is: 0011.880c.9f78
   The name of this device is vlan.0.1
   MTU is 1500 bytes
   The bandwidth is 10000 Mb/s
   Encapsulation ARPA, Loopback not set
   ARP type: ARPA,   ARP Timeout: 3600 seconds
   Policy Routing disabled
```

Use the **show ip interface** command to display information for interfaces configured for IP.

```
S Chassis(rw-config)->show ip interface vlan.0.1

 vlan.0.1 is Operationally up, Administratively up
   IP Address 10.21.130.59 Mask 255.255.128.0
```

```
   IP forwarding enabled
   Frame Type ARPA
   MAC-Address 00.11.88.0c.9f.78
   Incoming IPv4 Access list is
   Outgoing IPv4 Access list is
   Directed-broadcast is disabled
   MTU is 1500 bytes
   ARP Timeout is 3600 seconds
   ARP Retransmit Time is 1 seconds
   ARP Stale-Entry-Timeout is 1200 seconds
   Proxy ARP is disabled
   Gratuitous ARP updating is set to update on ARP replies and ARP requests
   Gratuitous ARP learning is not set
   ICMP Re-Directs are enabled
   ICMP Echo Replies are always sent
   ICMP Mask Replies are always sent
   NAT INSIDE:  Not Set
   NAT OUTSIDE: Not Set
   TWCB Redirect Outbound WebCache: Not Set
   Policy routing disabled
S Chassis(rw-config)->
```

This example shows how to display IPv6 configuration information for VLAN 51:

```
S Chassis(rw)->show ipv6 interface vlan.0.51

 vlan.0.51 is Operationally down, Administratively down
  IPv6 is enabled link-local address is fe80::21f:45ff:fe5b:f5cf%vlan.0.51
  Global unicast address(es):
    2001:11ac:fd34:50::abcd:33, subnet is 2001:11ac:fd34:50::/64
  Joined group address(es):
    (None)
  IPV6 forwarding disabled
  IPV6 address auto-configuration is enabled
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  Sending of ICMP Destination Unreachable Messages is enabled
  Sending of ICMP Redirect Messages is enabled
  Sending of ICMP Echo-Reply Messages is enabled
  ND DAD is enabled, number of DAD attempts: 1
S Chassis(rw)->
```

Procedure 36-1 describes how to configure the routing interface.

Procedure 36-1 Configuring the Routing Interface

Step	Task	Command(s)
1.	Enter router interface configuration command mode for the specified interface, from either global configuration or router configuration command mode.	**interface** {**vlan** *vlan-id* | **loopback** *loopback-id* | *interface-name*}
2.	Set the primary, and optionally the secondary or management, IPv4 address for this interface, in interface configuration command mode.	**ip address** {*ip-address* | *ip-address*/*prefixLength*} *ip-mask* [**primary** | **secondary** | **management**]
3.	Optionally, configure an IPv6 general prefix in global configuration mode to be assigned to an IPv6 address.	**ipv6 general-prefix** *name prefix/length*
4.	Set the IPv6 address for this interface in interface configuration command mode.	**ipv6 address** {*link-local-address* **link-local** | *ipv6-address/length* | *ipv6-prefix*/*length* **eui-64** | **autoconfig** | *general-prefix sub-bits/length*}
5.	Optionally disable IPv4 forwarding on this interface.	**no ip forwarding**
6.	Optionally, set the Equal Cost Multipath (ECM) forwarding algorithm for forwarding IP packets on routing interfaces, from global configuration command mode.	**ip ecm-forwarding-algo** [**hash-thold** | **round-robin**]
7.	Optionally, configure a secondary VLAN on the VLAN interface.	**secondary-vlan** *vlan-id*
8.	Enable this interface along with any changes made, in interface configuration command mode.	**no shutdown**

IP Static Routes

An IP static route can be configured as a traffic forwarding route or as a non-forwarding management route for IPv4.

Traffic forwarding static routes are configured in global configuration mode using the **ip route** command for IPv4 routes. See Traffic Forwarding IP Static Routes for a traffic forwarding static route discussion.

Non-forwarding management routes can be configured using either the **ip route** or **ipv6 route** commands in configuration command mode or the **set ip route** in any command mode. See "Traffic Non-Forwarding IP Static Routes" on page 36-16 for a non-forwarding static route discussion.

Traffic Forwarding IP Static Routes

Traffic forwarding IP static routes are configured by specifying the destination IPv4 prefix and mask or prefix/length for the route and one of the following:

- The next hop router IP address, optionally specifying the next hop interface ID or that the next hop interface is determined by route lookup
- The next hop interface name

- The next hop VLAN ID
- The egress VRF router as the next hop (IPv4 routes only)
- Packets destined for this route's subnet are silently dropped
- Packets destined for this route's subnet are dropped, and an ICMP network unreachable message is sent to the packet source

An administrative distance can be optionally configured that is used for route selection preference. The lower the numeric distance value, the greater the preference for the route. An OSPF tag-ID can be specified.

Routes are managed by the RTM (Route Table Manager) and are contained in the RIB (Route Information Base). The RIB contains up to 8 equal cost routes from any route source to each network and installs these routes in the FIB (Forwarding Information Base). The routes in the FIB are distributed to every module for use by the router's ingress module as frames are received.

A probe can be configured on a static route. When configuring a probe for the static route using the **probe** option, the probe session is created on the nexthop address. When the probe session goes down, the static route is disabled. When the probe session comes up, the static route is enabled.

Traffic forwarding IP static routes are configured using the **ip route** command for IPv4 routes or the **ipv6 route** command for IPv6 routes, in global configuration mode.

Traffic Forwarding IPv4 Static Route Examples

The following series of static route input examples are based upon the following route configuration:

```
# **** VRF default (default) ****
configure terminal
!
# Static routes configured on routed interfaces
 ip route 33.1.1.0/24 133.1.1.2 interface vlan.0.333 1
 ip route 33.1.2.0/24 144.1.1.2 interface vlan.0.444 1
 ip route 192.168.1.0/24 blackhole 1
 ip route 192.168.1.0/30 reject 1
 ip route 192.168.1.4/30 100.1.1.3 interface vlan.0.100 1
!
# Static routes configured on non-routed interfaces
 ip route 10.0.0.0/8 10.21.128.1 interface vlan.0.4000 1
 ip route 134.141.0.0/16 10.21.128.1 interface vlan.0.4000 1
!
exit
!
```

The following example enters a static route with no next-hop interface specified. The route prefix and length is **33.1.1.0/24** and the next-hop is **133.1.1.2**.

```
S Chassis(rw-config)->ip route 33.1.1.0/24 133.1.1.2
```

This is a legacy format. You are not prevented from entering the route in this format, but the behavior has changed as follows:

- A search of all configured subnets for a subnet containing the next-hop **133.1.1.2** is performed. That search will determine that this next-hop is on interface **vlan.0.333** as indicated in the configuration above. The configured route will be as if you had entered the command:

  ```
  S Chassis(rw-config)->ip route 33.1.1.0/24 133.1.1.2 interface vlan.0.333
  ```

- Should an interface not be found for this next-hop, the route will be configured as if you specified the route as a recursive route as follows:

  ```
  S Chassis(rw-config)->ip route 33.1.1.0/24 133.1.1.2 recursive
  ```

The following example enters a static route for prefix and length **33.1.2.0/24** with a next-hop of **144.1.1.2**, but this time specifying the interface, **vlan.0.444**, that the next-hop is on:

```
S Chassis(rw-config)->ip route 33.1.2.0/24 144.1.1.2 interface vlan.0.444
```

The following example configures a blackhole route for prefix and length **192.168.1.0/24**. Packets destined for blackhole routes are silently dropped. An ICMP network unreachable message is not sent to the packet source.

```
S Chassis(rw-config)->ip route 192.168.1.0/24 blackhole
```

The following example configures a reject route that overlaps the 192.168.1.0/24 blackhole route for prefix and length 192.168.1.0/30. In this case, packets destined for this next-hop are also dropped, but an ICMP network unreachable message is sent to the packet source:

```
S Chassis(rw-config)->ip route 192.168.1.0/30 reject
```

The following example configures an overlapping route allowing frames to 192.168.1.5 and 192.168.1.6 to be forwarded to next-hop **100.1.1.3** on interface **vlan.0.100**:

```
S Chassis(rw-config)->ip route 192.168.1.4/30 100.1.1.3 interface vlan.0.100
```

Use the **show ip route** command to display IP routes for this device. Route display can be narrowed by specifying route type: **connected**, **ospf**, **rip**, or **static**. The **show ip route** command output for this series of inputs is:

```
S Chassis(rw-config)->show ip route
Host IP Route Table for VRF default
Codes: C-connected, D-dynamic, H-host, S-static
       *-no forwarding interface

S*     10.0.0.0/8              10.21.128.1         vlan.0.4000
C*     10.21.128.0/17          10.21.130.151       vlan.0.4000
H      10.21.130.151           10.21.130.151       lo.0.1
S      33.1.1.0/24             133.1.1.2           vlan.0.333
S      33.1.2.0/24             144.1.1.2           vlan.0.444
C      100.1.1.0/24            100.1.1.2           vlan.0.100
H      100.1.1.2               100.1.1.2           lo.0.1
C      101.1.1.0/24            101.1.1.2           vlan.0.100
H      101.1.1.2               101.1.1.2           lo.0.1
H      127.0.0.1               127.0.0.1           lo.0.1
C      133.1.1.0/24            133.1.1.1           vlan.0.333
C      133.1.1.0/24            direct              vlan.0.333
H      133.1.1.1               133.1.1.1           lo.0.1
S*     134.141.0.0/16          10.21.128.1         vlan.0.4000
```

```
S      192.168.1.4/30          100.1.1.3          vlan.0.100

Number of routes = 15
```

Use the **show ip protocols** command to display information about IP protocols running on this device.

Traffic Non-Forwarding IP Static Routes

Non-forwarding IP static routes are management routes.

There are two methods for configuring a non-forwarding management route. The recommended method is to first set the routing interface as a non-forwarding interface using the IPv4 **no ip forwarding** command in interface configuration mode (IPv6 forwarding is disabled by default). In global configuration mode, configure the static route using the **ip route** command for an IPv4 route or **ipv6 route** command for and IPv6 route. Because the **ip route** and **ipv6 route** commands are configured in router configuration command mode, the configuration is capable of automatically determining the correct VLAN if not specified.

The second method is using the legacy command **set ip route** in system configuration mode specifying an IPv4 or IPv6 destination address.

For static routes that will be used to route transit frames, use the **ip route** command as described in section "Traffic Forwarding IP Static Routes" on page 36-13.

Traffic Non-Forwarding IP Static Route Examples

Non-forwarding interfaces are configured for IPv4 traffic using the **no ip forwarding** command and for IPv6 traffic using the **no ipv6 forwarding** command, in interface configuration mode. IPv6 forwarding is disabled by default on the interface. The following example enters static routes specifying the non-forwarding interface **vlan.0.4000** as the next-hop interface:

```
S Chassis(rw-config)->interface vlan.0.4000
S Chassis(rw-config-intf-vlan.0.4000)->no ip forwarding
S Chassis(rw-config-intf-vlan.0.4000)->exit
S Chassis(rw-config)->ip route 10.0.0.0/8 10.21.128.1 interface vlan.0.4000
S Chassis(rw-config)->ip route 125.20.0.0/16 125.20.10.1 interface vlan.0.4000
S Chassis(rw-config)->ipv6 route 2001:11ac:fd34::/48 2001:11ac:fd34:3333::4
interface vlan.0.4000
```

The following example uses the legacy method of configuring a non-forwarding static route from the system command mode with a destination of 192.122.173.42 and a gateway of 192.122.168.38:

```
S Chassis(rw)->set ip route 192.122.173.42 192.122.168.38
```

The following example uses the legacy method of configuring a non-forwarding static route from the system command mode with a destination of 192.122.173.50 and a next-hop interface of VLAN 50:

```
S Chassis(rw)->set ip route 192.122.173.50 vlan.0.50
```

Procedure 36-2 describes how to configure a non-forwarding IP traffic route.

Procedure 36-2 Configuring Non-forward IP Static Routes

Step	Task	Command(s)
1.	In interface configuration mode, set the routing interface for this static route to not forward IP traffic.	**no ip forwarding**

Procedure 36-2 Configuring Non-forward IP Static Routes (continued)

Step	Task	Command(s)
2.	In global configuration mode, configure the static route.	**ip route** {*prefix mask* | *prefix*/*prefix-length*}{*ip-address* [**recursive**] | **interface** *interface-name*| **vlan** *vlan-id*} [*distance*] [**tag** *tag-id*] [**blackhole**] [**reject**]
3.	Optionally, in global configuration command mode, configure IPv6 static routes. IPv6 forwarding is disabled by default.	**ipv6 route** *prefix*/*length* {*ipv6-address* [**recursive** | **interface** *interface-name*] | **interface** *interface-name* | **vlan** *vlan-id* | **blackhole** | **reject**} [*distance*] [**tag** *tag-id*]
4.	Alternatively, in system configuration mode, configure the non-forwarding static route. This method supports legacy configurations. It is recommended that you use the method described in steps 1 - 3.	**set ip route** {*destination* | **default**} {*gateway* | *interface*} [*mask*]

IPv6 Neighbor Discovery

The Neighbor Discovery (ND) protocol for IPv6 is defined in RFC4861. The neighbor discovery protocol uses ICMPv6 messages to determine the link-layer addresses of nodes residing on the same local link, to locate neighboring routers, to learn certain link and address configuration information, and to track the reachability of neighbors.

You can configure the IPv6 prefixes to include in IPv6 Neighbor Discovery (ND) router advertisements for the interface.

This example sets the IPv6 prefix ba10:1100:aa11/48 to be included in the ND router advertisements for VLAN 50:

```
S Chassis(su-config)->interface vlan 50
S Chassis(su-config-intf-vlan.0.50)->ipv6 nd prefix ba10:1100:aa11/48
S Chassis(su-config-intf-vlan.0.50)->
```

Address Configuration Flag

You can set the ND managed address configuration flag in router advertisements. When the managed address configuration flag is set, attached hosts use stateful autoconfiguration to obtain addresses. The managed address configuration flag feature is disabled by default.

This example enables the use of stateful autoconfiguration by attached hosts to obtain addresses on VLAN 50:

```
S Chassis(su-config)->interface vlan 50
S Chassis(su-config-intf-vlan.0.50)->ipv6 nd managed-config-flag
S Chassis(su-config-intf-vlan.0.50)->
```

Reachable Time

You can set the number of milli-seconds the router is considered to be reachable on this IPv6 interface between 0 and 3600000 (1 hour). A neighbor is determined to be reachable if positive confirmation has been received within the reachable interval that the forward path to the neighbor

was functioning properly. If no confirmation is received within the reachable interval, it is assumed that the neighbor is unreachable.

This example sets the router reachability interval to **120000** ms (120 seconds) for VLAN 50:

```
S Chassis(su-config)->interface vlan 50
S Chassis(su-config-intf-vlan.0.50)->ipv6 nd reachable-time 120000
S Chassis(su-config-intf-vlan.0.50)->
```

Other Configuration Flag

You can set the other configuration flag in router advertisements. When the other config flag is set, attached hosts use stateful autoconfiguration to obtain non-address information. If the managed address configuration flag (see "Address Configuration Flag" on page 36-17) is set, the attached host uses stateful autoconfiguration to obtain non-address information regardless of the other config flag setting.

This example enables the use of stateful autoconfiguration by attached hosts to obtain non-address information on VLAN 50:

```
S Chassis(su-config)->interface vlan 50
S Chassis(su-config-intf-vlan.0.50)->ipv6 nd other-config-flag
S Chassis(su-config-intf-vlan.0.50)->
```

Neighbor Solicitation Interval

You can set the interval between neighbor solicitation messages in milli-seconds between 1000 and 4294967295.

This example sets the interval between neighbor solicitation messages at 1.5 seconds for VLAN 50:

```
S Chassis(su-config)->interface vlan 50
S Chassis(su-config-intf-vlan.0.50)->ipv6 nd ns-interval 1500
S Chassis(su-config-intf-vlan.0.50)->
```

Router Advertisement Interval

You can set the maximum and minimum router advertisement interval for the IPv6 interface to an interval between 4 and 1800 seconds. If minimum interval is not specified, the minimum router advertisement interval is set to **.33** times the current maximum router advertisement interval.

This example sets the maximum router advertisement interval to 650 seconds and the minimum router advertisement value to .33 times 650 (214) for VLAN 50:

```
S Chassis(su-config)->interface vlan 50
S Chassis(su-config-intf-vlan.0.50)->ipv6 nd ra interval 650
S Chassis(su-config-intf-vlan.0.50)->
```

Router Lifetime Value

You can set the router lifetime value in seconds for router advertisements on the IPv6 interface to 0 or from the configured maximum router advertisement interval to 9000 seconds. The router lifetime value specifies the usefulness of the router as a default router on this IPv6 interface. Configuring the lifetime to **0** specifies that the router should not be considered a default router for

this interface. If the lifetime is set to a nonzero value, it can not be less than the configured maximum router advertisement interval.

This example sets the router lifetime value to 2200 seconds for VLAN 50:

```
S Chassis(su-config)->interface vlan 50
S Chassis(su-config-intf-vlan.0.50)->ipv6 nd ra lifetime 2200
S Chassis(su-config-intf-vlan.0.50)->
```

Router Advertisement Maximum Transmission Unit

You can set the Maximum Transmission Unit (MTU) value in bytes for router advertisements on the IPv6 interface to a value between 1280 - 4294967295 bytes.

This example sets the router advertisement MTU value to **12000** bytes for VLAN 50:

```
S Chassis(su-config)->interface vlan 50
S Chassis(su-config-intf-vlan.0.50)->ipv6 nd ra mtu 12000
S Chassis(su-config-intf-vlan.0.50)->
```

Router Advertisement Hoplimit Suppression

You can suppress IPv6 router advertisement transmissions on an interface. The router advertisement hoplimit suppress feature suppresses IPv6 router advertisement transmissions on an interface by setting the router advertisement hoplimit to 0.

This example enables router advertisement hoplimit suppression for VLAN 50:

```
S Chassis(su-config)->interface vlan 50
S Chassis(su-config-intf-vlan.0.50)->ipv6 ra hoplimit suppress
S Chassis(su-config-intf-vlan.0.50)->
```

Router Advertisement Suppression

You can configure ND to stop sending router advertisements on the IPv6 interface. By default, router advertisements are sent on the IPv6 interface. The **ipv6 nd ra suppress** command stops the sending of router advertisements on the IPv6 interface.

This example suppresses the sending of router advertisements on VLAN 50:

```
S Chassis(su-config)->interface vlan 50
S Chassis(su-config-intf-vlan.0.50)->ipv6 nd ra suppress
S Chassis(su-config-intf-vlan.0.50)->
```

Duplicate Address Detection

IPv6 Duplicate Address Detection (DAD) is described in RFC 4862. DAD uses neighbor solicitation and neighbor Advertisement messages to verify the uniqueness of an address. DAD must be performed on unicast addresses prior to assigning them to an interface. An address remains in a tentative state while DAD is being performed. If a tentative address is found to be a duplicate, an error message is returned and the address is not assigned to the interface.

IPv6 Address Autoconfiguration

IPv6 address autoconfiguration determines whether an interface IPv6 address will be auto-configured by acquiring the address from an attached router or must be manually configured. IPv6 address autoconfiguration is enabled for the interface by specifying the **autoconfig** option when entering the **ipv6 address** command in interface configuration mode.

The following example configures VLAN 3050 to acquire its IPv6 address from an attached router:

```
S Chassis(su-config)->interface vlan 3050
S Chassis(su-config-intf-vlan.0.3050)->ipv6 address autoconfig
S Chassis(su-config-intf-vlan.0.3050)->
```

Binding an IPv6 Address to a MAC Hardware Address

Much like IPv4 addresses are bound to MAC hardware addresses in the ARP table, IPv6 addresses are bound to MAC hardware addresses in the neighbor discovery cache.

Use the **ipv6 neighbor** command in global configuration mode to statically bind an IPv6 address to a MAC hardware address.

The following example configures a static neighbor cache entry for IPv6 address **2001:11ac:fd34:3333:0:0:0:3** on a hardware device with a MAC address of **1111.1111.1111** on interface VLAN 51:

```
S Chassis(su)->configure
S Chassis(su-config)->ipv6 neighbor 2001:11ac:fd34:3333:0:0:0:3 1111.1111.1111 interface vlan.0.2501
S Chassis(su-config)->show ipv6 neighbor
FLAGS:    I = Incomplete     R = Reachable
          S = Stale          D = Delay
          P = Probe          L = Local
          F = Fixed (Static) H = Host Owned

IPv6 Address                          Hardware Address  Flg Age        Updated Expire Interface   Port
------------------------------------- ----------------- --- ---------- ------- ------ ----------- --------
2001:11ac:fd34:3333:0:0:0:3           11-11-11-11-11-11  FR         1m       -      - vlan.0.2501 host.0.1
2501:0:0:0:0:0:0:4                    00-1f-45-5b-f5-cf  LR       1h01m      -      - vlan.0.2501 host.0.1
fe80:0:0:0:21f:45ff:fe5b:f5cf         00-1f-45-5b-f5-cf  LR       1h01m      -      - vlan.0.2501 host.0.1
2502:0:0:0:0:0:0:4                    00-1f-45-5b-f5-cf  LR      18h26m      -      - vlan.0.2502 host.0.1
fe80:0:0:0:21f:45ff:fe5b:f5cf         00-1f-45-5b-f5-cf  LR      18h26m      -      - vlan.0.2502 host.0.1
fe80:0:0:0:21f:45ff:fe5b:f5cf         00-1f-45-5b-f5-cf  LR      18h26m      -      - vlan.0.2503 host.0.1
fe80:0:0:0:21f:45ff:fe5b:f5cf         00-1f-45-5b-f5-cf  LR      18h26m      -      - vlan.0.2504 host.0.1
3014:0:0:0:0:0:0:4                    00-1f-45-5b-f5-cf  LR      18h26m      -      - vlan.0.3014 host.0.1
fe80:0:0:0:21f:45ff:fe5b:f5cf         00-1f-45-5b-f5-cf  LR      18h26m      -      - vlan.0.3014 host.0.1
------------------------------------- ----------------- --- ---------- ------- ------ ----------- --------
Neighbor Entries Found: 9
S Chassis(su-config)->
```

IPv4 and IPv6 ICMP Configuration

The Internet Control Message Protocol (ICMP) is a protocol in the Internet Protocol Suite. It is chiefly used by the operating systems of networked computers to send error messages when a requested service is not available or a host or router could not be reached. ICMPv6 is the IPv6 version of ICMP. The router can be configured to:

- Enable sending ICMP destination unreachable messages on an interface using the **ip icmp unreachable** or **ipv6 icmp unreachable** command
- Enable sending of ICMP redirect messages on an interface using the **ip icmp redirects** or **ipv6 icmp redirects** command
- Enable sending ICMP echo-reply messages on an interface using the **ip icmp echo-reply** or **ipv6 icmp echo-reply** command.

Configuring IPv6 Neighbor Discovery

Procedure 36-3 describes how to configure a static IPv6 neighbor discovery cache entry.

Procedure 36-3 Configuring an IPv6 Static Neighbor Discovery Cache Entry

Task	Command(s)
In configuration command mode, optionally create a static binding between an IPv6 address to a MAC hardware address.	**ipv6 neighbor** *ipv6-address hardware-address* **interface** *interface*
In any command mode, verify the IPv6 neighbor discovery cache configuration.	**show ipv6 neighbors** [*ipv6-address*] [**group**] [**interface** *interface*] [**verbose**] [**statistics**]
In interface configuration command mode, set the managed address configuration flag in router advertisements.	**ipv6 nd managed-config-flag**
In interface configuration command mode, set the interval between neighbor solicitation messages.	**ipv6 nd ns-interval** *interval*
In interface configuration command mode, set the other configuration flag in router advertisements.	**ipv6 nd other-config-flag**
In interface configuration command mode, configure the IPv6 prefixes to include in IPv6 Neighbor Discovery (ND) router advertisements for the interface.	**ipv6 nd prefix** *ipv6-prefix/length*
In interface configuration command mode, configure neighbor discovery to insert a zero (0) in the Cur Hop Limit field of the router advertisements sent by this interface.	**ipv6 nd ra hoplimit suppress**
In interface configuration command mode, set the maximum and minimum router advertisement interval for the IPv6 interface.	**ipv6 nd ra interval** {*maxinterval* | **msec** *maxinterval*} [*mininterval*]
In interface configuration command mode, set the router lifetime value in seconds for router advertisements on the IPv6 interface.	**ipv6 nd ra lifetime** *value*
Optionally, in interface configuration mode, configure the router to respond to all ARP and Neighbor Discovery requests.	**arp-nd-proxy-all**
In interface configuration command mode, set the Maximum Transmission Unit (MTU) value in bytes for router advertisements on the IPv6 interface.	**ipv6 nd ra mtu** *mtu*
In interface configuration command mode, stop sending router advertisements on the IPv6 interface.	**ipv6 nd ra suppress**
In interface configuration command mode, set the number of milli-seconds the router is considered to be reachable on this IPv6 interface.	**ipv6 nd reachable-time** *interval*
In interface configuration mode, Optionally modify the number of neighbor discovery neighbor solicitation messages to send during duplicate address detection on unicast IPv6 addresses on the interface.	**ipv6 nd dad attempts** *num*

The ARP Table

Address Resolution Protocol (ARP) is the method for finding a MAC hardware address when only the IP address is known. The S-Series firmware allows you to configure Address Resolution Protocol (ARP) table entries and parameters. ARP is used to associate IP addresses with MAC addresses. Once determined, the IP address and MAC association is stored in an ARP cache for rapid retrieval. An IP datagram is then encapsulated into a link-layer frame and sent over the network. A retransmit time period can be set to determine how often ARP requests are transmitted.

ARP table entries can be temporary or permanent. A temporary ARP entry has a timeout interval associated with it. The ARP entry expires at the end of the timeout interval. Expired ARP entries are referred to as stale entries. A stale entry timeout value determines how long the stale entry remains in the ARP table before it is removed.

Use the **arp** command in configuration command mode to configure a permanent static ARP entry.

Use the **set arp** command to configure a permanent ARP entry with the option of setting the entry to temporary.

Use the **show arp** command to display ARP table entries.

Use the **clear arp** command to clear specific static or all temporary ARP entries from the ARP table.

Gratuitous ARP

Gratuitous ARP provides an ARP announcement packet containing valid sender hardware and protocol addresses for the host that sent it. Such a request is not intended to solicit a reply, but merely updates the ARP caches of other hosts that receive the packet. Gratuitous ARP is usually formatted as an ARP request, but it may also be formatted as an ARP reply. ARP announcements are typically sent out during startup. This helps to resolve problems which would otherwise occur if, for example, an IP-address-to-MAC-address mapping changed because a network card was replaced. In this example, gratuitous ARP solves the problem of remote hosts that still have the old mapping in their ARP caches. The S-Series provides for setting the behavior when an ARP announcement is received for an already existing ARP table entry or for a non-existing ARP table entry, referred to as ARP learning.

IP gratuitous ARP is enabled by default for the modification of pre-existing ARP table entries and is disabled by default for the learning of new table entries.

Use the **ip gratuitous-arp** command in interface configuration command mode to:

- Configure the device to ignore gratuitous ARP announcements received for existing ARP table entries
- Configure the device to change the ARP table only if the gratuitous ARP announcement is a reply
- Configure the device to change the ARP table only if the gratuitous ARP announcement is a request.

Use the **ip gratuitous-arp-learning** command, in interface configuration command mode, to allow an interface to learn new ARP bindings using gratuitous ARP. The router will learn new ARP bindings from reply, request, or both types of ARP announcements, based upon the option specified in this command.

Gratuitous ARP configuration does not affect normal ARP operations. Normal ARP packets (non gratuitous) will always be learned and updated regardless of gratuitous ARP configuration.

Proxy ARP

Proxy ARP provides for the ability of a device on a given network to answer the ARP queries for a network address that is not on that network. The ARP Proxy, being aware of the traffic destination's location, provides its own MAC address in reply. Serving as an ARP Proxy for another host effectively directs LAN traffic to the Proxy. The "directed" traffic is then typically routed by the proxy to the intended destination via another interface.

Proxy ARP is enabled by default. Typically, proxy arp is only used to reply to requests for hosts that are reachable via a non-default route. Proxy ARP can be configured to respond to ARP requests for hosts that are only reachable via the default route. Proxy ARP can also be configured to respond to ARP requests that are received on the interface to which this command is applied, if the source IP address of the request is reachable on the local interface.

ARP/ND Proxy-All

In an effort to reduce the amount of broadcast and multicast traffic in a broadcast domain, the router can be configured to respond to all ARP and Neighbor Discovery requests. The router will respond with the hardware address of the host that owns the IP address, if the ARP/ND entry for the Target Address exists in the router's ARP/ND cache. If the Target Address does not exist in the ARP/ND cache or if the entry in unresolved, the ARP/ND request will be dropped.

Use the command **arp-nd-proxy-all** in interface configuration mode to enable ARP/ND proxy all.

Removing the Multicast ARP Restriction

As specified in RFC 1812, by default the router must not believe any ARP packet that claims the packet MAC address is broadcast or multicast. The multicast restriction can be removed on the interface using the **ip multicast-arp-learning** command in interface configuration mode.

ARP Configuration Examples

The following example:

- Temporarily configures the IP address 10.21.128.1, MAC address 00:00:5e:00:01:01 binding in the ARP table
- Changes the ARP timeout value to 2800 seconds
- Changes the stale entry timeout value to 900 seconds

```
S Chassis(rw)->set arp 10.21.128.1 00:00:5e:00:01:01 temp
S Chassis(rw)->configure
S Chassis(rw-config)->arp timeout 2800
S Chassis(rw-config)->arp stale-entry-timeout 900
S Chassis(rw-config)->show arp
FLAGS:    U = Unresolved     S = Static
          L = Local          V = VRRP
          * = Stale          B = Best Guess Interface

IP Address      Hardware Address  Flg Age        Updated Interface   Port
--------------- ----------------- --- ---------- ------- ----------- --------
10.21.128.1     00:00:5e:00:01:01   B      4h55m      1m vlan.0.1    ge.1.1
10.21.130.59    00:11:88:0c:9f:78   L      5h05m       - vlan.0.1    host.0.1
```

```
-------------- ---------------- --- ---------- ------- ---------- --------
ARP Entries Found: 2
S Chassis(rw-config)->
```

The following example enables gratuitous ARP and ARP learning for ARP replies on VLAN 1:

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 1
S Chassis(rw-config-intf-vlan.0.1)->ip gratuitous-arp reply
S Chassis(rw-config-intf-vlan.0.1)->ip gratuitous-arp-learning reply
S Chassis(rw-config-intf-vlan.0.1)->exit
S Chassis(rw-config)->
```

The following example enables proxy ARP for both default and local routes:

```
S Chassis(rw)->configure
S Chassis(rw-config)->ip prox
S Chassis(rw-config)->interface vlan 1
S Chassis(rw-config-intf-vlan.0.1)->ip proxy-arp default-route local
S Chassis(rw-config-intf-vlan.0.1)->exit
S Chassis(rw-config)->
```

Procedure 36-4 describes how to configure the ARP table.

Procedure 36-4 Configuring the ARP Table

Step	Task	Command(s)
1.	Add mapping entries to the ARP table with the option of configuring them as temporary.	**set arp** *ip-address mac-address* [**interface** *interface*] [**temp**]
2.	In configuration command mode, add permanent (static) entries to the ARP table.	**arp** *ip-address mac-address* [**interface** *interface*]
3.	Optionally, in configuration command mode, change the duration that temporary ARP entries will stay in the ARP table before expiring.	**arp timeout** *seconds*
4.	Optionally, in configuration command mode, change the duration to wait before retransmitting ARP requests when trying to resolve ARP entries.	**arp retransmit-time** *seconds*
5.	Optionally, in interface configuration command mode, override the default ARP update process. • **ignore** - Ignore all gratuitous ARP frames, no updates will occur. This option will also prevent any new learning from gratuitous arps, if the command ip gratuitous-arp-learning was used. • **reply** - Update from gratuitous arp reply only. • **request** - Update from gratuitous arp request only.	**ip gratuitous-arp** {**ignore** | **reply** | **request**}

Procedure 36-4 Configuring the ARP Table (continued)

Step	Task	Command(s)
6.	Optionally, in interface configuration command mode, allow an interface to learn new ARP bindings using gratuitous ARP. • **both** - Allows learning from both gratuitous arp reply and request. • **reply** - Allows learning from gratuitous arp reply. • **request** - Allows learning from gratuitous arp request.	**ip gratuitous-arp-learning** {**both** | **reply** | **request**}
7.	Optionally, in interface configuration command mode, enable proxy ARP on an interface. • **default-route** - Sets the router to respond to ARP requests for hosts that are only reachable via the default route. Typically, proxy arp is only used to reply to requests for host that are reachable via a non-default route. • **local** - Allows the router to respond to ARP requests that are received on the interface to which this command is applied if the target IP address of the request is reachable on the interface that received the request.	**ip proxy-arp** [**default-route**] [**local**]
8.	Optionally, in interface configuration command mode, remove the multicast restriction on ARP packets.	**ip multicast-arp-learning**
9.	Optionally, in interface configuration mode, configure the router to respond to all ARP and Neighbor Discovery requests.	**arp-nd-proxy-all**

IP Broadcast

Directed Broadcast

A directed broadcast address for each physical network has all ones in the host ID part of the address. The packet originates from a network device that is not part of the destination subnet and is forwarded in the same manner as a unicast packet sent to its destination subnet. When the packet reaches the directed broadcast address, if directed broadcast is enabled on the device, it is sent to every host on the destination network or subnetwork by rewriting the directed broadcast address to that of the standard broadcast address on that destination subnet. If directed broadcast is disabled on the destination interface, directed broadcasts are dropped.

Use the **ip directed-broadcast** command, in interface configuration command mode, to enable directed broadcasts for directed broadcasts sent to this interface.

Directed Broadcast Configuration Example

The following example enables directed broadcasts on VLAN 5:

```
S Chassis(rw)->configure
```

```
S Chassis(rw-config)->interface vlan 5
S Chassis(rw-config-intf-vlan.0.5)->ip directed-broadcast
S Chassis(rw-config-intf-vlan.0.5)->exit
S Chassis(rw-config)->
```

UDP Broadcast Forwarding

Typically, broadcast packets from one interface are not forwarded (routed) to another interface. However, some applications use UDP broadcasts to detect the availability of services, and some protocols, such as BOOTP/DHCP, require broadcast forwarding to provide services to clients on other subnets. Configuring UDP broadcast forwarding on the S-Series device involves enabling it for one or more protocols, and assigning IP helper addresses as described in this section.

Use the **ip forward-protocol** command in configuration command mode to enable UDP broadcast forwarding for the specified port. The following keywords are supported for common UDP ports:

- bootps — Specifies the Bootstrap Protocol server (67) port
- domain — Specifies the Domain Name Service (53) port
- nameserver — Specifies the IEN116 name service (42) port
- netbios-dgm — Specifies the NetBIOS datagram service (138) port
- netbios-ns — Specifies the NetBIOS name service (137) port
- tacacs — Specifies the Terminal Access Controller Access Control System (49) port
- tftp — Specifies the Trivial File Transfer Protocol (69) port
- time — Specifies the Time (37) port

UDP Broadcast Configuration Examples

This example shows how to enable forwarding of Domain Naming System UDP datagrams (port 53):

```
S Chassis(rw-config)->ip forward-protocol udp 53
```

This example shows how to enable forwarding of Domain Naming System UDP datagrams (port 53) by naming the protocol:

```
S Chassis(rw-config)->ip forward-protocol udp domain
```

Procedure 36-5 describes how to configure IP broadcast.

Procedure 36-5 Configuring IP Broadcast

Step	Task	Command(s)
1.	In interface configuration command mode, enable IP directed broadcasts on an interface.	**ip directed-broadcast**

Procedure 36-5 Configuring IP Broadcast (continued)

Step	Task	Command(s)
2.	In configuration command mode, optionally, enable UDP broadcast forwarding and specify the destination port number or keyword that controls the forwarding protocol. • *port* - **1** - **65535** • **bootps** - Specifies the Bootstrap Protocol server (67) port. • **domain** - Specifies the Domain Name Service (53) port. • **nameserver** - Specifies the IEN116 name service (42) port. • **netbios-dgm** - Specifies the NetBIOS datagram service (138) port. • **netbios-ns** - Specifies the NetBIOS name service (137) port. • **tacacs** - Specifies the Terminal Access Controller Access Control System (49) port. • **tftp** - Specifies the Trivial File Transfer Protocol (69) port. • **time** - Specifies the Time (37) port.	**ip forward-protocol udp** [*port*]
3.	In interface configuration command mode, optionally, enable DHCP/BOOTP relay and the forwarding of local UDP broadcasts, specifying a new destination address.	**ip helper-address** *address*
4.	In global or interface configuration command mode, optionally insert relay agent information option 82 and its sub-options into the relay agent DHCP packet.	See "DHCP Relay Agent Information Options" on page 36-27.

DHCP and BOOTP Relay

DHCP/BOOTP relay functionality is applied with the help of UDP broadcast forwarding. A typical situation occurs when a host requests an IP address with no DHCP server located on that segment. A routing module can forward the DHCP request to a server located on another network if:

- UDP broadcast forwarding is enabled
- The address of the DHCP server is configured as a helper address on the receiving interface

The DHCP/BOOTP relay agent function will detect the DHCP request and make the necessary changes to the header, replacing the destination address with the address of the server and the source with its own address, and then send it to the server. When the response comes from the server, the DHCP/BOOTP relay function sends it to the host.

Use the **ip helper-address** command in conjunction with the **ip forward-protocol** command to configure DHCP/BOOTP relay functionality to the specified server(s).

DHCP Relay Agent Information Options

When forwarding local UDP broadcasts from a local client, the DHCP relay agent needs to include information about itself in order for the DHCP server to determine which pool of client addresses

to pull the lease from. Including Option 82 and its sub-options in the DHCP relay information provides the required DHCP relay information.

Several commands are available to configure which DHCP relay agent information options are used by the switch to affect the selection of a lease when it is operating as a DHCP relay agent.

- Use the **ip dhcp relay information option** command, in either global or interface configuration mode, to insert the **circuit-id** (1) and **remote-id** (2) sub-options of the Relay Agent Information option (82) into the relay agent DHCP packets. Refer to RFC 3046 for detailed descriptions of these sub-options.
  - The default **circuit-id** sub-option value inserted into the relay agent DHCP packet is the interface name of the interface receiving the request from the client, in the form of vlan.0.x where x is the VLAN id between 1 and 4094. This default value can be over-ridden at the interface level by using the **ip dhcp relay information option circuit-id** command in interface configuration mode.
  - The **remote-id** sub-option is used to identify the remote host end of the circuit. The default value inserted into the relay agent DHCP packet is the MAC address of the chassis. This default value can be over-ridden by using the **ip dhcp relay information option remote-id** command in global configuration mode or interface configuration mode.
- Use the **ip dhcp relay information option vpn** command, in either global or interface configuration mode, to insert the **virtual subnet selection** (151), **link selection** (5), and **server identifier override** (11) sub-options into the relay agent DHCP packet.
  - The **virtual subnet selection** (VSS) options/sub-options are described in RFC 6607. They are used to pass VSS information about a VPN to the DHCP server to assist in determining the subnet on which to select an address. You can set the VPN id for a VRF with the **vpn id** command. If a VPN id is not configured for the VRF, the virtual subnet selection sub-option will contain the VRF name.
  - The **link selection** sub-option is described in RFC 3527. The link-selection sub-option is used by any DHCP relay agent that desires to specify a subnet/link for a DHCP client request that it is relaying but needs the subnet/link specification to be different from the IP address the DHCP server should use when communicating with the relay agent. By default, the link selection sub-option contains the subnet of the inbound interface to which the client is connected. This default value can be changed with the **ip dhcp relay information option link-selection** command.
  - The **server identifier override** sub-option is described in RFC 5107. This sub-option allows the DHCP relay agent to specify a new value for the server ID option, which is inserted by the DHCP server in the reply packet. This allows the DHCP relay agent to act as the actual DHCP server so that subsequent requests from the client will come to the relay agent rather than to the DHCP server directly. The server identifier override sub-option contains the IP address of the inbound interface to which the client is connected, which is the IP address on the relay agent that is accessible from the client. Using this information, the DHCP client sends all renew and release requests to the relay agent. The relay agent adds all of the appropriate sub-options and then forwards the request packets to the original DHCP server.
- Use the **ip dhcp relay information option server-override** command, in either global or interface configuration mode, to insert only the **link selection** (5), and **server identifier override** (11) sub-options into the relay agent DHCP packet.
- Use the **ip dhcp relay information option link-selection** command in interface configuration mode to specify that the **link selection** sub-option should be included in the Relay Agent Information and to define a different subnet from the interface's primary IP address. This allows you to select a secondary IP address on an interface to be used to help in DHCP pool

selection on the server. The subnet selection can be based on the DHCP client's vendor id (option 60) or its hardware MAC address.

Use the **ip dhcp relay source-interface** command, in global or interface configuration mode, to specify the source interface (VLAN or loopback) to be used used in the Relay Agent packets sent to the DHCP server or other relay agent. This command should be used in conjunction with the **ip dhcp relay information option server-override** or **ip dhcp relay information option vpn** commands, which cause the **server identifier override** (11) sub-option to be added to the Relay Agent DHCP packets sent to the DHCP server.

Note: The source interface specified with the **ip dhcp relay source-interface** command must belong to the same VRF specified with the **ip helper-address** command.

Command Order of Precedence

The following precedence rules apply to the DHCP relay agent information commands that can be configured in both router global and interface configuration modes.

If the **ip dhcp relay information option** commands or **ip dhcp relay source-interface** command are configured in global configuration mode but not configured in interface configuration mode, the global configuration is applied to all interfaces.

If the **ip dhcp relay information option** commands or **ip dhcp relay source-interface** command are configured in both global configuration mode and interface configuration mode, the interface configuration command takes precedence over the global configuration command. However, the global configuration is applied to interfaces that have not been configured at the interface level.

If the **ip dhcp relay information option** commands or **ip dhcp relay source-interface** command are not configured in global configuration mode but are configured in interface configuration mode, only the interfaces that have been configured are affected. All other interfaces are not impacted by the configuration.

DHCP/BOOTP Relay Configuration Examples

The following example shows how to permit UDP broadcasts from hosts on networks 191.168.1.255 and 192.24.1.255 to reach servers on other networks:

```
S Chassis(rw)->configure
S Chassis(rw-config)->ip forward-protocol udp
S Chassis(rw-config)->interface vlan.0.1
S Chassis(rw-config-intf-vlan.0.1)->ip helper-address 192.168.1.255
S Chassis(rw-config-intf-vlan.0.1)->exit
S Chassis(rw-config)->interface vlan.0.2
S Chassis(rw-config-intf-vlan.0.2)->ip helper-address 192.24.1.255
```

This example configures several Option 82 sub-options to be included in the DHCP relay agent information packets. This example also shows how you would use the **link selection** option to tell the DHCP server to assign leases from different sub-networks, depending on information received in the DHCP client request. For example, when the relay agent receives a DHCP client request from a host with MAC address 002654AF123B, the relay agent sets the DHCP relay agent information link selection option value to 10.180.2.0. If the MAC address were 00301E44AC12, the option value would be set to 10.180.3.0.

```
S Chassis(su)->configure
S Chassis(su-config)->interface vlan.0.10
S Chassis(su-config-intf-vlan.0.10)->ip address 10.180.1.8 255.255.255.0 primary
```

```
S Chassis(su-config-intf-vlan.0.10)->ip address 10.180.2.8 255.255.255.0
secondary
S Chassis(su-config-intf-vlan.0.10)->ip address 10.180.3.8 255.255.255.0
secondary
S Chassis(su-config-intf-vlan.0.10)->ip address 10.180.4.8 255.255.255.0
secondary
S Chassis(su-config-intf-vlan.0.10)->ip directed-broadcast
S Chassis(su-config-intf-vlan.0.10)->ip helper-address 11.5.255.255 global
S Chassis(su-config-intf-vlan.0.10)->ip dhcp relay information option
S Chassis(su-config-intf-vlan.0.10)->ip dhcp relay information option vpn
S Chassis(su-config-intf-vlan.0.10)->ip dhcp relay information option remote-id
Shrewsbury
S Chassis(su-config-intf-vlan.0.10)->ip dhcp relay information option circuit-id
engineering
S Chassis(su-config-intf-vlan.0.10)->ip dhcp relay information option
link-selection 10.180.2.0 mac 002654AF123B
S Chassis(su-config-intf-vlan.0.10)->ip dhcp relay information option
link-selection 10.180.2.0 vendor-id "MSFT 5.0"
S Chassis(su-config-intf-vlan.0.10)->ip dhcp relay information option
link-selection 10.180.3.0 mac 00301E44AC12
S Chassis(su-config-intf-vlan.0.10)->ip dhcp relay information option
link-selection 10.180.4.0 mac 001CC504BC34
S Chassis(su-config-intf-vlan.0.10)->exit
S Chassis(su-config)->
```

Router Management and Information Display

Table 36-2 lists routing parameters and their default values.

Table 36-2 Default IP Routing Parameters

Parameter	Description	Default Value
ARP entry type	Specifies whether an ARP table entry is permanent or temporary.	permanent
ARP retransmit time	Specifies the duration in seconds to wait before retransmitting ARP requests.	1 second
ARP stale entry timeout	Specifies the duration in seconds an ARP entry will remain in the stale state before the entry is removed from the ARP table.	1200 seconds
ARP timeout	Specifies the duration in seconds for temporary ARP entries to stay in the ARP table before expiring.	3600 seconds
directed broadcast	The ability to address a destination host such that the arriving packet will be broadcasted to the network as if it was a normal broadcast generated by the receiving host.	disabled

Table 36-2 Default IP Routing Parameters (continued)

Parameter	Description	Default Value
equal cost multipath algorithm	Specifies the algorithm used for selecting the next path used by the equal cost multipath feature.	hash threshold
global router	Specifies the default router used when configuring the router directly from configuration command mode. The current implementation supports a single global router and up to 128 VRF router instances depending upon the system being configured.	global
gratuitous ARP	A feature that overrides the normal ARP updating process by providing an ARP announcement packet containing valid sender hardware and protocol addresses for the host that sent it.	enabled for ARP replies and ARP requests
gratuitous ARP learning	A feature that allows an interface to learn new ARP bindings using gratuitous ARP.	disabled
IP ICMP echo reply	Specifies whether IPv4 ICMP echo-reply messages are sent.	enabled
IP ICMP mask reply	Specifies whether IPv4 ICMP mask reply messages are sent.	enabled
IP ICMP redirection	Specifies whether IPv4 ICMP redirect messages are sent.	enabled
IP ICMP unreachable	Specifies whether IPv4 ICMP unreachable messages are sent.	enabled
IPv4 forwarding	Specifies whether or not the routing interface will forward IPv4 traffic.	enabled
IPv6 address autoconfiguration	Specifies whether IPv6 addresses are auto configured on the interface.	disabled
IPv6 forwarding	Specifies whether or not the routing interface will forward IPv6 traffic.	disabled
neighbor discovery Duplicate Address Detection (DAD)	Specifies the number of DAD messages neighbor discovery will send out to attempt to determine whether the “tentative” address for this interface is a duplicate of another address in the network.	1 attempt
proxy ARP	A feature that provides for the ability of a device on a given network to answer the ARP queries for a network address that is not on that network.	enabled (no local or default-route)

Table 36-3 describes how to manage IP configuration.

Table 36-3 Managing the Router

Task	Command
To clear this router configuration:	**clear router vrf** *vrf-name*

Table 36-3 Managing the Router (continued)

Task	Command
To delete one or all entries from the ARP table:	**clear arp** {*ip-address* | **all**}
To delete all non-static (dynamic) entries from the ARP table:	**clear arp-cache** [*ip-address*] [**interface** *interface*]

Table 36-4 describes how to display IP configuration information and statistics.

Table 36-4 Displaying IP Routing Information and Statistics

Task	Command
To display router configuration:	**show router** [*name*]
To display the application limits for this router:	**show limits** [**vrf** *vrf*] [**application** *application*]
To display non-default, user entered configuration, or all configuration for this router: Supported applications can be determined by entering the **show running-config ?** command.	**show running-config** [**all**] [*application* [**all**]]
To display configuration information for one or more interfaces:	**show interface** [*interface-name*]
To display configuration information for one or more IPv4 routing interfaces:	**show ip interface** [*interface-name*] [**brief**]
To display configuration information for one or more IPv6 routing interfaces:	**show ipv6 interface** [*interface-name* [**prefix**]] [**brief**]
To display information about IP protocols running on this device:	**show ip protocols**
To display information about IP routes:	**show ip route** [**host** [**connected** | **host-address** | **dynamic** | **static**]] [*dest-address* [*prefix-mask*] | *prefix*/*prefix-length* | **connected** | **ospf** | **rip** | **static** | **summary**]
To display the device's ARP table:	**show arp** [*ip-address*] [**interface** *interface*] [**statistics**]
To display debug IP packet utility settings:	**show debugging**
To display debug IP VRRP utility settings:	**debug ip vrrp show**

IP Debug

The IP debug utility provides debug level monitoring of :

- BGP
- IP Packets
- OSPFv2
- VRRP

Within the IP packet debug utility, monitoring can be filtered based upon VLAN, MAC address, Ether type, access list or ARP address using the **debug packet filter** command. Debug message display can be both throttled to a specified number of messages per second or a maximum limit as well as set for a maximum or minimum level of information per message using the **debug packet control** command. If the maximum limit is reached, restart the packet debug utility to restart message display. By default messages display at a verbose level. The information level can also be set to brief to display less information per message.

The **debug ip packet-restart** command restarts the packet logging process. Depending on the packet debug limit configuration, a specified number of logs will be displayed as frames are processed. By default, this is 10 logs. Use the restart command to see another 10 logs.

Use the **debug ip packet** command in configuration command mode to configure IP packet debug.

Use the **debug ip bgp** command to enable the debug IP BGP utility for monitoring BGP timers, messages and routes.

Use the **debug ip ospf** to enable the debug IP OSPFv2 utility for monitoring OSPF adjacencies, LSA generation, packets, and retransmissions.

Use the **debug packet show-statistics** command to display debug statistics for packet and host counters and IPv4 exceptions.

Use the **debug packet clear-statistics** command to clear all debug utility counters.

Use the **show debugging** command to display the current IP debug utility settings.

Table 36-5 describes how to configure IP debug. All IP debug commands are accessed in configuration command mode.

Table 36-5 Configuring IP Debug

Task	Command(s)
Optionally, disable the debug IP packet utility.	**no debug packet**
Optionally, restart the debug IP packet utility.	**debug packet restart**
Optionally, filter the display of debug IP packet messages by the specified criteria.	**debug packet filter** {[**vlan-in-list** *vlan-list*] [**vlan-out-list** *vlan-list*] [**port-in-list** *port-list*] [**port-out-list** *port-list*] [**src-mac** *mac-address*] [**dest-mac** *mac-address*] [**etype** *value*] [**access-list** *access-list*] [**arp** {*ip-address netmask* | *ip-address/length*}]}
Optionally, set debug IP packet utility control parameters that throttle or limit message display and set the amount of information displayed per message.	**debug packet control** {[**throttle** *throttle*] [**limit** *limit*] [**verbose** | **brief**]}
Optionally, enable the debug IP BGP utility.	**debug ip bgp** {**keepalive** | **notification** | **open** | **route-refresh** | **route-add** | **route-ineligible** | **route-remove** | **update** | **dampen** | **timer**}
Optionally, enable the debug IP OSPF utility.	**debug** {**ip**} **ospf** {**adj** | **lsa-generation** | **packet** | **retransmission** | **trace-interface** *trace-interface*}
Optionally, enable the debug IP VRRP utility.	**debug ip vrrp** [**advertisements** | **critical-ip** | **trace-interface** *trace-interface* | **trace-vrid** *vrid*]

Terms and Definitions

Table 36-6 lists terms and definitions used in this IP routing configuration discussion.

Table 36-6 IP Routing Terms and Definitions

Term	Definition
Address Resolution Protocol (ARP)	A protocol providing a method for finding a MAC hardware address when only the IP address is known.
ARP proxy	Provides for the ability of a device on a given network to answer the ARP queries for a network address that is not on that network.
blackhole route	Silently drops packets destined for this route's subnet.
broadcast forwarding	Provides for the ability for rout UDP broadcasts in order to provide services to clients on a different subnet than the one originating the broadcast.
directed broadcast	The ability to address a destination host such that the arriving packet will be broadcasted to the network as if it was a normal broadcast generated by the receiving host.
Duplicate Address Detection (DAD)	An IPv6 neighbor discovery capability that uses neighbor solicitation and neighbor advertisement messages to verify the uniqueness of an address.
general prefix	The ability to assign a name to represent a network prefix from which longer IPv6 addresses can be configured.
global router	The default router from which VRF routing instances are configurable.
gratuitous ARP	A method for overriding the normal ARP process that provides an ARP announcement packet containing valid sender hardware and protocol addresses for the host that sent it. ARP announcements are sent out during startup.
IP address	An address used by the IP protocol to identify a routing interface or routing device.
IP address helper	The ability to specify the IP address the UDP forwarded packet should be sent to.
IP debug	A feature that monitors a set of IP processes and displays messages when configured events occur.
managed address configuration	A DHCPv6 capability that determines whether the interface will send out IPv6 address configuration to an interface with IPv6 autoconfiguration enabled.
management interface	A non-forwarding interface to which an IP subnet can be assigned, allowing the network administrator to create an out-of-band management subnet designed to only pass network management data.
neighbor discovery	An IPv6 protocol defined in FRC4861 that uses ICMPv6 messages to determine the link-layer addresses of nodes residing on the same local link, to locate neighboring routers, to learn certain link and address configuration, and to track the reachability of neighbors.
neighbor unreachability detection	An IPv6 neighbor discovery capability that detects the failure of a neighbor or the failure of the forward path to the neighbor.
relay agent	A DHCPv6 application that provides a means for relaying DHCPv6 requests between a subnet to which no DHCP server is connected to other subnets on which servers are attached.
routing interface	A VLAN or loopback interface configured for IP routing.
static route	An administratively configured IP route consisting of the destination and next-hop IP addresses from the IP router the route is configured on.

Table 36-6 IP Routing Terms and Definitions (continued)

Term	Definition
Virtual Routing and Forwarding (VRF)	Provides a method of partitioning your network into segregated routed domains that may contain unique IP networks, routes, and other configuration that would otherwise conflict if they were all deployed on the same router.

37

Tunneling Configuration

This chapter provides information about configuring and monitoring layer 3 and layer 2 tunneling on S-Series devices.

How to Use Tunneling in Your Network

Tunneling uses network layer tunneling protocols to connect disjoint networks within the same (trusted) enterprise campus network, resulting in the destination address of the tunnel functioning as a logical next hop.

Data is transmitted in the form of IP packets. The information contained in a data packet is called the payload. A data packet header contains the routing information required to transmit the packet to a remote destination. A tunnel is selected as the route interface based upon a route lookup. Tunneling involves the use of a tunnel protocol that encapsulates the payload of the packet entering the tunnel within another (outer) header based upon tunnel parameters. Thus a tunneled packet has an inner and an outer header.

The inner header contains the original packet header. The IP type (IPv4 or IPv6) of the original header is determined by the original packet source and destination address type. The outer delivery header is the tunnel header. The IP type of the tunnel header is determined by the route lookup source and destination IP address type configured for the tunnel.

The tunnel mode is expressed as the inner IP address type over the outer tunnel IP address type. For example, an IPv6 packet encapsulated into an IPv4 tunnel interface would use a tunnel that supports tunnel mode IPv6 over IPv4. Tunnel modes that support IPv6 over IPv4 are GRE and IPv6 over IPv4, configured using the tunnel mode keyword **ipv6ip**.

To create a tunnel, both endpoint devices must support the same tunneling mode.

The S-Series platform supports tunneling modes:

- Generic Routing Encapsulation (GRE) which provides generic support for all supported IPv4 and IPv6 tunnel IP type combinations, as defined in RFC 2784, along with the keyword extensions defined in RFC 2890. The GRE mode should be used if you do not want to limit the tunnel to a specific IP header combination. This implementation does not support RFC 1701.

- IP-IP tunneling which provides support for IPv4 over an IPv4 Layer 3 tunnel interface as defined in RFC 2003.
- IPv6 tunneling which provides support for IPv6 over an IPv6 Layer 3 tunnel interface as defined in RFC 2473.
- IPv4 to IPv6 tunneling which supports IPv4 over an IPv6 Layer 3 tunnel interface as defined in RFC 2473.
- IPv6 to IPv4 tunneling which supports IPv6 over an IPv4 Layer 3 tunnel interface as defined in RFC 2473.

A tunnel interface can be assigned to a static route using the **ip route** or **ipv6** route command, depending upon the route IP type. The tunnel source and destination must be reachable either by a configured static route or a supported routing protocol such as RIP, BGP, or OSPF.

If route lookup selects a route using a tunnel, the underlying delivery interface is determined based upon the destination address of the selected route. The tunnel delivery interface is displayed using the **show tunnel** command.

The S-Series platform supports remote mirroring using a Layer 2 GRE tunnel mode. Refer to "Remote Mirroring Using a Layer 2 GRE Tunnel" on page 8-9 for Remote mirroring Layer 2 GRE tunnel details.

The S-Series platform supports the Virtual Private Port Service feature which is a L2 tunnel mode permitting the user to extend a virtual wire through an arbitrary routed network, using GRE with transparent bridging. Refer to "Virtual Private Port Service" on page 37-5 for Virtual Private Port Service details.

See the **interface** command entry, in the *Extreme Networks S-Series CLI Reference*, for create, enable, and disable tunnel command details.

Implementing Tunneling

Do the following at both ends of the tunnel to implement tunneling:

- Assure that the interfaces for both the tunnel source and destination are reachable using a static route or a routing protocol such as RIP, BGP or OSPF
- Create the interface that will be the source address of the tunnel (usually a loopback interface)
- Create the tunnel interface and enter tunnel configuration mode using the **interface tunnel** command
- Configure an IP address for the tunnel interface
- Configure the source and destination IPv4 or IPv6 addresses for the L3 tunnel used by the outer header that the packet payload is encapsulated into. This step is not appropriate for a L2 tunnel. Assigning the source and destination addresses for a L2 tunnel will force the tunnel into a down state.
- Configure the encapsulation method (tunnel mode) for the tunnel
- If the configured tunnel mode is GRE:
  - Optionally, configure the keepalive transmit interval and the number of keepalive retries for the tunnel
  - Optionally, configure a GRE keyword used by the receiver to authenticate the source of the packet
- Optionally, modify the packet Type of Service

- Optionally, configure a tunnel probe to monitor the destination address associated with the tunnel

Tunneling Overview

Tunnel Source and Destination Reachability

A tunnel has a source and destination tunnel interface associated with it. The source interface can be a VLAN or loopback interface on the router, but is usually a loopback interface. The tunnel source interface must be in an up state and the destination IP address must be reachable for the tunnel to be operational. Reachability can be achieved by creating a static route on the local router to the tunnel destination address or by means of a supported routing protocol such as RIP, BGP, or OSPF.

This example shows how to create a static route to the tunnel destination address of 99.99.99.1 using VLAN 50:

```
S Chassis(su-config)->ip route 99.99.99.1/32 vlan 50
```

With the static route configured, ping the destination address using the **ping** command to assure reachability.

Tunnel Interface

With tunnel destination address reachability established, the tunnel interface is created using the **interface tunnel** command in global configuration command mode, specifying the tunnel ID. Entering the command provides access to the tunnel interface configuration mode. The tunnel ID is in the format **tun.0.***x*, where *x* is the tunnel interface number (1 - 50). Supported tunnel parameters are configured in tunnel configuration mode.

IP Address

The interface IP address is the standard IP address associated with any interface and should not be confused with the tunnel source address which is used by the outer header to route the encapsulated payload.

Use the **ip address** command for IPv4 addressing or the **ipv6 address** command for IPv6 addressing, in tunnel interface configuration mode, to configure an IP address on the interface.

Tunnel Mode

The tunnel mode determines the encapsulation capabilities of the tunnel. GRE mode provides for all four IPv4 and IPv6 encapsulation types. GRE mode is used when you do not want to limit the tunnel to a particular encapsulation type. There is also a tunnel mode specific to each of the four encapsulation types. Use a tunnel mode specific to an encapsulation type if you wish to limit the tunnel to that encapsulation type.

Use the **tunnel mode** command in tunnel configuration mode to configure the tunnel encapsulation type. The supported encapsulation types and their associated command keywords are:

- GRE – **gre**
- IPv4 over IPv4 – **ipip**
- IPv4 over IPv6 – **ipip ipv6**

- IPv6 over IPv4 – **ipv6ip**
- IPv6 over IPv6 – **ipv6ip ipv6**

GRE Keepalive

GRE keepalive is used to monitor the tunnel destination. Unlike a tunnel probe that is only capable of monitoring the state of the specified IP address, GRE keepalive both monitors the state of the IP address and whether the end-point was able to decapsulate the tunnel packet. A failed keepalive causes the tunnel to transition to the down state.

When enabling GRE keepalive, specify the transmit interval that determines the period between the transmission of keepalive messages and the number of GRE keepalive retries.

Use the **tunnel keepalive** command, in tunnel configuration mode, to enable GRE keepalive on a GRE IPv4 over IPv4 tunnel.

GRE Keyword

The GRE keyword, as defined in RFC 2890, is a four octet number inserted by the encapsulator. It may be used by the receiver to authenticate the source of the packet. If a GRE keyword is configured at either end of the tunnel, the keyword configuration must match at both ends of the tunnel. If a mis-match occurs, packets are dropped and an asterisk (*) is displayed to the left of the **show tunnel** command tunnel entry.

Use the tunnel keyword command, in tunnel configuration mode, to specify a GRE keyword for the tunnel.

Tunnel Probe

A tunnel probe is used to monitor a tunnel endpoint IP address. A tunnel probe can be used in any tunnel mode. If a probe fails, the associated tunnel is taken down. A default ICMP tunnel probe exists named **$tunnel_default** or a probe can be configured using the tracked object manager probe facility. See "Configuring a Probe for Server Load Balancing" on page 13-9 for configuration details for creating and configuring a probe.

Note: It is recommended that you use GRE keepalive to monitor a tunnel. GRE keepalive both monitors the state of the IP address and whether the end-point was able to decapsulate the tunnel packet. Tunnel probes are also supported. Do not configure both a GRE keepalive and tunnel probe.

Use the **tunnel probe** command, in tunnel configuration mode, specifying the tunnel destination address, to monitor the tunnel endpoint IP address.

Type of Service (ToS)

By default the packet entering a tunnel inherits the ToS of the original packet payload. The ToS value used by the outer tunnel header can be modified. Use the **tunnel tos** command, in tunnel configuration mode, to modify the ToS value for the packet as it transits the tunnel. When the packet is decapsulated at the tunnel destination, the original packet ToS value applies.

Checkspoof

The checkspoof feature verifies that the source address of the packet received on the interface is reachable from the receiving interface or any interface depending upon the checkspoof

configuration. This feature helps protect against attacks where the source of the attack is unknown to the router. The checkspoof feature can be configured on any interface.

Use the **ip checkspoof** or **ipv6 checkspoof** command in tunnel configuration mode to configure checkspoofing on the tunnel interface for the specified IP type.

See the "Routing Interface Commands" chapter of the *Extreme Networks S-Series CLI Reference* for details on the **ip checkspoof** command and the "IPv6 Interface Commands" chapter of the *Extreme Networks S-Series CLI Reference* for details on the **ipv6 checkspoof** command.

Access-Groups

By applying ACLs to an access-group, access restrictions to inbound or outbound frames can be applied to an interface when operating in router mode. Access-groups can be applied to a tunnel interface.

Use the **ip access-group** command to apply IPv4 ACLs to a tunnel interface and the **ipv6 access-group** command to apply IPv6 ACLs to a tunnel interface in tunnel configuration mode.

Virtual Private Port Service

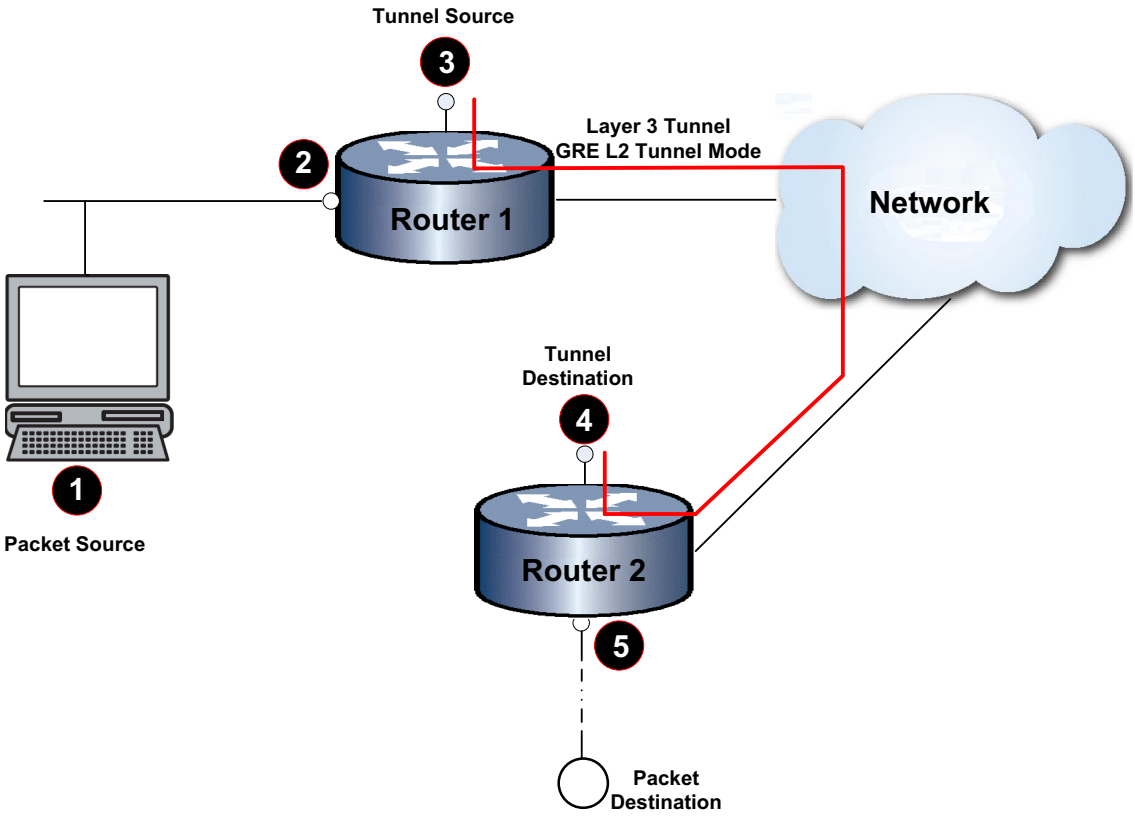
Virtual Private Port Services permit the user to extend a virtual wire through an arbitrary routed network using GRE with transparent bridging. This feature is referred to as a Virtual Private Port Service (VPPS). The configuration on each end of the tunnel specifies a physical port to be connected to the VPPS. Once configured in this manner, any packets arriving on that physical port are immediately encapsulated and routed to the other end of the tunnel. When the packet arrives at the remote end of the tunnel, it is immediately de-encapsulated and sent out the configured port on that end of the tunnel. The net effect is to create a direct connection between each end of the tunnel. No switch or router configuration affects the original packet. The packet arriving at the ingress port is tunneled without change to the tunnel's remote end.

Figure 37-1 on page 37-6 presents a Virtual Private Port Service configuration example. In this example, a packet is sourced at PC1 (callout **1**) and enters the VPPS at port ge.1.1 of Router 1 (callout 2). The VPPS is configured using a GRE L2 tunnel mode configuration on the L3 tunnel between Router 1 and Router 2. In our example the L3 tunnel configuration is limited to the configuration of a source and destination address. Refer to "Tunnel Configuration Example" on page 37-13 for a more detailed L3 tunnel configuration example and walkthrough.

From the perspective of Router 1, the VPPS is configured with a L3 tunnel source of loopback address 88.88.88.1 (callout **3**) and a L3 tunnel destination of loopback address 99.99.99.1 (callout **4**). The VPPS ingress port is configured as Router 1's port ge.1.1. The VPPS egress port is Router 2's port ge.1.2 (callout 5) and is specified when configuring the GRE L2 tunnel mode on Router 2

From the perspective of Router 2, the VPPS is configured with a L3 tunnel source of loopback address 99.99.99.1 and a L3 tunnel destination of loopback address 88.88.88.1. The VPPS ingress port is configured as Router 2's port ge.1.2.

Figure 37-1 Virtual Private Port Service Configuration Example



1	PC 1, Packet Source, IP address 2111::2	**4**	L3 Tunnel Destination Loopback Address 99.99.99.1
2	Virtual Private Port Service Ingress Port ge.1.1	**5**	Virtual Private Port Service Egress Port ge.1.2
3	L3 Tunnel Source Loopback Address 88.88.88.1		

Router 1 VPPS Configuration

This example shows how to set:

- IP address **88.88.88.1** as the GRE L2 tunnel source:
- IP address **99.99.99.2** as the GRE L2 tunnel destination
- Physical port **ge.1.1** as the bound physical port for the GRE L2 tunnel **1**

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface tunnel 1
S Chassis(rw-config-intf-tun.0.1)->tunnel source 88.88.88.1
S Chassis(rw-config-intf-tun.0.1)->tunnel destination 99.99.99.1
S Chassis(rw-config-intf-tun.0.1)->tunnel mode gre l2 ge.1.1
S Chassis(rw-config-intf-tun.0.1)->no shutdown
S Chassis(rw-config-intf-tun.0.1)->
```

Router 2 VPPS Configuration

This example shows how to set:

- IP address **99.99.99.1** as the GRE L2 tunnel source:

- IP address **88.88.88.1** as the GRE L2 tunnel destination
- Physical port **ge.1.2** as the bound physical port for the GRE L2 tunnel **1**

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface tunnel 1
S Chassis(rw-config-intf-tun.0.1)->tunnel source 99.99.99.1
S Chassis(rw-config-intf-tun.0.1)->tunnel destination 88.88.88.1
S Chassis(rw-config-intf-tun.0.1)->tunnel mode gre l2 ge.1.2
S Chassis(rw-config-intf-tun.0.1)->no shutdown
S Chassis(rw-config-intf-tun.0.1)->
```

Virtual Private Port Service (VPPS) MTU Handling and Remote Mirroring

You must assure that a jumbo path exists between the two tunnel endpoints. When packets enter a VPPS, packet size increases by the size of the extra layer 3 IP and the GRE headers. This packet size increase can range from 38 to 64 bytes, depending upon IP and GRE configuration. If an encapsulated packet exceeds the destination MTU, the packet is dropped. ICMP does not report back to the source that the packet exceeded MTU for a tunnel with mirroring enabled. A VPPS will report back to the original source. Refer to "Jumbo Frames" on page 6-8 for Jumbo port configuration details.

Source Address Only Configuration

Multiple tunneled port mirrors can be configured to use a single source address configured L2 tunnel (VPPS) at its destination, by configuring the destination end as an any-remote tunnel. An any-remote tunnel accepts any remote IP as the source IP address, as long as the destination IP address matches this tunnel's source IP. When any-remote is enabled on the destination end of a VPPS tunnel:

- The any-remote configured tunnel accepts any tunneled packet destined to it's tunnel source. It decapsulates the packet and forwards it out the Ethernet port assigned to the tunnel.
- Any packets received on the Ethernet port assigned to the tunnel are switched or routed as normal, and not sent across the VPPS.
- If a destination address is configured on an any-remote enabled L2 tunnel, it has no practical affect, but it must have a route to the destination for the tunnel to be up.

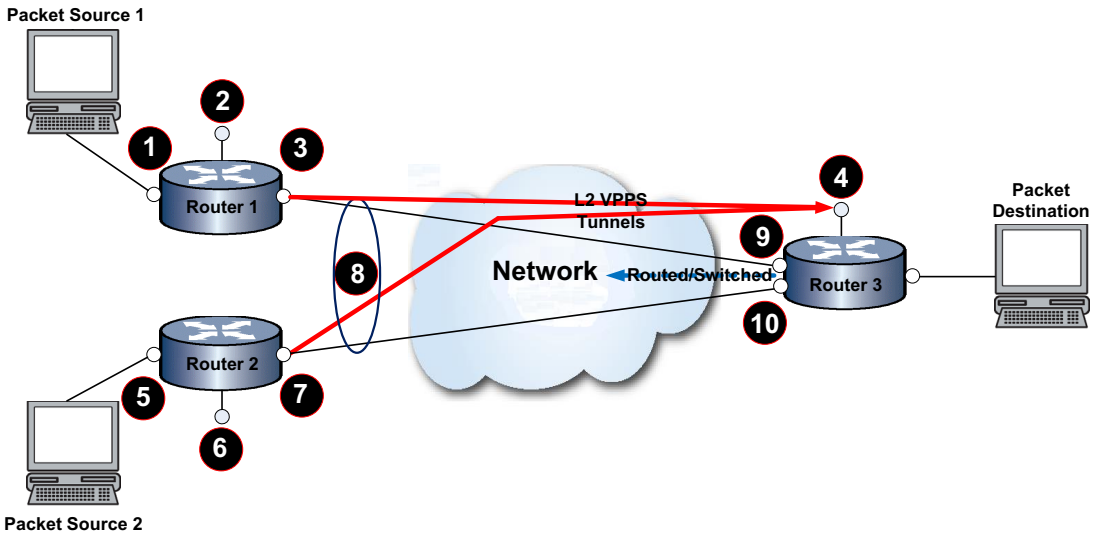
Figure 37-2 on page 37-8 presents an example of an any-remote enabled L2 tunnel. In this example port mirroring is enabled on Router 1 packet sources:

- **Packet Source 1** – Port mirror enabled source port ge.1.1 (Callout 1) and target port tg.1.1 (Callout 3)
- **Packet Source 2** – Port mirror enabled source port ge.1.2 (Callout 4) and target port tg.1.2 (Callout 6)

Two mirror enabled VPPS tunnels (one for each mirrored source) with a single tunnel destination are created on Router1 and Router2:

- tun.0.1 (Router1) – With a tunnel source of IP address 77.77.77.1 on loopback interface loop.0.1 (Callout 2), a bound physical port of tg.1.1 (same as the Packet Source 1 port mirror target), and a tunnel destination of IP address 99.99.99.1 (Callout 4)
- tun.0.2 (Router2) – With a tunnel source of IP address 88.88.88.1 on loopback interface loop.0.2 (Callout 6), a bound physical port of tg.1.2 (same as the Packet Source 2 port mirror target), and a tunnel destination of IP address 99.99.99.1 (Callout 4)

Figure 37-2 Virtual Private Port Service Any-Remote Configuration Example



1	ge.1.1 – tun.0.1 Mirrored Port (Source)	**6**	loop.0.2 – tun.0.2 Source, IP Add 88.88.88.1
2	loop.0.1 – tun.0.1 Source, IP Add 77.77.77.1	**7**	tg.2.1 – tun.0.2 Mirrored Port (Target)
3	tg.1.1 – tun.0.1 Mirrored Port (Target)	**8**	VLAN 20 (specified in static route to tunnel destination)
4	loop.0.1 – tun.0.1-2 Destination, IP Add 99.99.99.1	**9**	ge.1.3 – Any-Remote tun.0.1 Ingress Port
5	ge.1.2 – tun.0.2 Mirrored Port (Source)	**10**	ge.1.4 – Any-Remote tun.0.2 Ingress Port

On the Router 3 destination side of the VPPS tunnel, an any-remote L2 tunnel is created as tun.0.1 with a tunnel source of IP address 99.99.99.1 on loopback interface of loop.0.1 (Callout 4) and a bound physical port of ge.1.3. No tunnel destination is configured and will have on affect if one is configured.

A static route with VPPS tunnel destination as its destination assures a route exists for the VPPS tunnels.

Packets from Packet Source 1 are port mirrored on port ge.1.1 and targeted to port tg.1.1 (Router1) which is the bound physical port for tun.0.1. Packets are tunneled to Router 3 loopback interface loop.0.1. Returning packets will be sourced to loopback interface 1 on Router 3, but will be decapsulated and will be switched or routed out port ge.1.3 on Router 3 to its destination.

Packets from Packet Source 2 are port mirrored on port ge.1.2 and targeted to port tg.1.2 (Router2) which is the bound physical port for tun.0.2. Packets are tunneled to Router 3 loopback interface loop.0.1. Returning packets will be sourced to loopback interface 1 on Router 3, but will be decapsulated and will be switched or routed out port ge.1.3 on Router 3 to its destination.

Router 1 VPPS Configuration

This example shows how to set:

- Loopback interface 1 is used as the tunnel sources for VPPS tunnel 1
- VLAN interface 20 to be used with the static route that assures a route exists to the tunnel destination
- Port mirror enabled VPPS tunnel 1 with destination 99.99.99.1
- A static route to the VPPS tunnel destination

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface loopback 1
S Chassis(rw-config-intf-loop.0.1)->ip address 77.77.77.1 255.255.255.255 primary
```

```
S Chassis(rw-config-intf-loop.0.1)->ipv6 address 2007::1/128
S Chassis(rw-config-intf-loop.0.1)->no shutdown
S Chassis(rw-config-intf-loop.0.1)->exit
S Chassis(rw-config)->interface vlan 20
S Chassis(rw-config-intf-vlan.0.20)->ip address 6.1.1.1 255.255.255.0 primary
S Chassis(rw-config-intf-vlan.0.20)->ipv6 address 2666::1/64
S Chassis(rw-config-intf-vlan.0.20)->ipv6 nd ra suppress
S Chassis(rw-config-intf-vlan.0.20)->ipv6 forwarding
S Chassis(rw-config-intf-vlan.0.20)->no shutdown
S Chassis(rw-config-intf-vlan.0.20)->exit
S Chassis(rw-config)->interface tunnel 1
S Chassis(rw-config-intf-tun.0.1)->tunnel source 77.77.77.1
S Chassis(rw-config-intf-tun.0.1)->tunnel destination 99.99.99.1
S Chassis(rw-config-intf-tun.0.1)->tunnel mode gre l2 tg.1.1
S Chassis(rw-config-intf-tun.0.1)->tunnel mirror enable
S Chassis(rw-config-intf-tun.0.1)->no shutdown
S Chassis(rw-config-intf-tun.0.1)->exit
S Chassis(rw-config)->ip route 99.99.99.1/32 6.1.1.2 interface vlan.0.20 1
```

Router 2 VPPS Configuration

This example shows how to set:

- Loopback interface 2 is used as the tunnel sources for VPPS tunnel 2
- VLAN interface 20 to be used with the static route that assures a route exists to the tunnel destination
- Port mirror enabled VPPS tunnel 2 is configured with destination 99.99.99.1
- A static route to the VPPS tunnel destination

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface loopback 2
S Chassis(rw-config-intf-loop.0.2)->ip address 88.88.88.1 255.255.255.255 primary
S Chassis(rw-config-intf-loop.0.2)->ipv6 address 2008::1/128
S Chassis(rw-config-intf-loop.0.2)->no shutdown
S Chassis(rw-config-intf-loop.0.2)->exit
S Chassis(rw-config)->interface vlan 20
S Chassis(rw-config-intf-vlan.0.20)->ip address 6.1.1.2 255.255.255.0 primary
S Chassis(rw-config-intf-vlan.0.20)->ipv6 address 2666::2/64
S Chassis(rw-config-intf-vlan.0.20)->ipv6 nd ra suppress
S Chassis(rw-config-intf-vlan.0.20)->ipv6 forwarding
S Chassis(rw-config-intf-vlan.0.20)->no shutdown
S Chassis(rw-config-intf-vlan.0.20)->exit
S Chassis(rw-config)->interface tunnel 2
S Chassis(rw-config-intf-tun.0.2)->tunnel source 88.88.88.1
S Chassis(rw-config-intf-tun.0.2)->tunnel destination 99.99.99.1
S Chassis(rw-config-intf-tun.0.2)->tunnel mode gre l2 tg.1.2
S Chassis(rw-config-intf-tun.0.2)->tunnel mirror enable
```

```
S Chassis(rw-config-intf-tun.0.2)->no shutdown
S Chassis(rw-config-intf-tun.0.2)->exit
S Chassis(rw-config)->ip route 99.99.99.1/32 6.1.1.2 interface vlan.0.20 1
```

Router 2 Any-Remote Configuration

This example shows how to set:

- Loopback interface 1 to be used as the VPPS tunnel destination for VPPS tunnels 1 and 2 and the Any-Remote tunnel source
- VLAN interface 20 to be used with the static route that assures a route exists to the VPPS tunnel destination
- Any-Remote tunnel 1

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface loopback 1
S Chassis(rw-config-intf-loop.0.1)->ip address 99.99.99.1 255.255.255.255 primary
S Chassis(rw-config-intf-loop.0.1)->ipv6 address 2009::1/128
S Chassis(rw-config-intf-loop.0.1)->no shutdown
S Chassis(rw-config-intf-loop.0.1)->exit
S Chassis(rw-config)->interface vlan 20
S Chassis(rw-config-intf-vlan.0.20)->ip address 6.1.1.3 255.255.255.0 primary
S Chassis(rw-config-intf-vlan.0.20)->ipv6 address 2666::3/64
S Chassis(rw-config-intf-vlan.0.20)->ipv6 nd ra suppress
S Chassis(rw-config-intf-vlan.0.20)->ipv6 forwarding
S Chassis(rw-config-intf-vlan.0.20)->no shutdown
S Chassis(rw-config-intf-vlan.0.20)->exit
S Chassis(rw-config)->interface tunnel 1
S Chassis(rw-config-intf-tun.0.1)->tunnel source 99.99.99.1
S Chassis(rw-config-intf-tun.0.1)->tunnel mode gre l2 ge.1.3
S Chassis(rw-config-intf-tun.0.1)->tunnel any-remote enable
S Chassis(rw-config-intf-tun.0.1)->no shutdown
S Chassis(rw-config-intf-tun.0.1)->exit
```

Layer 2 Tunnel Bridge Port (Virtual Private Ethernet Service)

A tunnel bridge port (Virtual Private Ethernet Service) is a virtual bridge port attached to a layer 2 tunnel router interface. Unlike a Virtual Private Port, which is a logical connection of the ingress port of one device to the egress port on another device located elsewhere in the network, a tunnel bridge port permits the transparent connection of two disjoint bridge infrastructures over an intermediate routed network, while preserving the simplicity of a bridged network and providing all of the failover features provided in the bridge protocols between the two disjoint bridged networks.

The tunnel bridgeport is bound to the routing tunnel interface using the **tunnel mode gre L2** command.

The tunnel bridge port is specified as tun.0.*y* where *y* is the tunnel bridge port number.

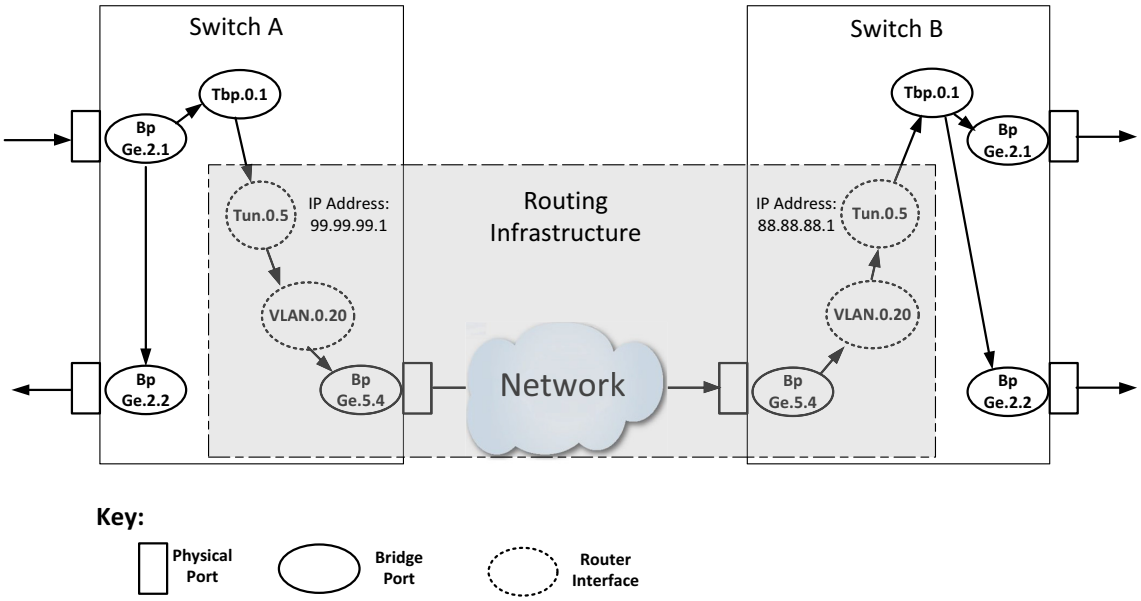
Figure 37-3 on page 37-11 displays a L2 tunnel bridge port configuration example. In this example, the switch configurations include:

- The VLAN 10 egress list configured for bridge ports **ge.2.1**, **ge.2.2**, **tbp.0.1**

- The VLAN 20 egress list configured for bride port **ge.5.4**.

A packet arrives at bridge port **ge.2.1** and is flooded to the egress list of VLAN 10. The tunnel bridge port **tbp.0.1** is on this egress list, so a copy of the original packet is encapsulated and routed to switch B, where it is decapsulated and flooded to VLAN 10 on that switch. Since the SMAC of the first packet has now been learned on bridge port **ge.2.1** on switch A and **tbp.0.1** on switch B, packets returning along this path do not flood.

Figure 37-3 L2 Tunnel Bridge Port Configuration Example



This example shows how to configure Switch A for tunnel bridge port **tbp.0.1** bound to GRE L2 tunnel **5**:

- IP address **99.99.99.1** as the GRE L2 tunnel source:
- IP address **88.88.88.1** as the GRE L2 tunnel destination
- Tunnel bridge port **tbp.0.1** bound to GRE L2 tunnel **5**

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface tunnel 5
S Chassis(rw-config-intf-tun.0.5)->tunnel source 99.99.99.1
S Chassis(rw-config-intf-tun.0.5)->tunnel destination 88.88.88.1
S Chassis(rw-config-intf-tun.0.5)->tunnel mode gre l2 tbp.0.1
S Chassis(rw-config-intf-tun.0.5)->no shutdown
S Chassis(rw-config-intf-tun.0.5)->
```

Configuration on Switch B is the same except for reversing the tunnel source and destination addresses.

Tunneling in a NAT Context

A L3 or L2 tunnel, including tunnel bridge ports, can be configured as the inside or outside interface in a NAT context by entering the appropriate IPv4 or IPv6 NAT inside or outside command within the tunnel interface configuration mode.

Use the {**ip** | **ipv6**} **nat inside** command, in tunnel interface command mode, to enable the tunnel interface as a NAT inside interface.

Use the {**ip** | **ipv6**} **nat outside** command, in tunnel interface command mode, to enable the tunnel interface as a NAT outside interface.

See Chapter 45, **Network Address Translation (NAT) Configuration** for NAT configuration details.

Tunneling in a TWCB Context

A L3 or L2 tunnel, including tunnel bridge ports, can be configured as the TWCB outbound interface by entering the appropriate TWCB redirect out command within the tunnel interface configuration mode.

Use the {**ip** | **ipv6**} **twcb** *webcache-name* **redirect out**, in tunnel interface command mode, to enable the interface as a TWCB outbound interface.

See Chapter 47, **Transparent Web Cache Balancing (TWCB) Configuration** for TWCB configuration details.

Configuring Tunneling

Procedure 37-1 describes tunneling configuration.

Procedure 37-1 Tunneling Configuration

Step	Task	Command(s)
1.	In global configuration mode, create the tunnel interface and enter tunnel interface configuration mode.	**interface tunnel** *tunnel-id*
2.	In tunnel configuration mode, specify the tunnel source IPv4 or IPv6 address.	**tunnel source** *ip-address*
3.	In tunnel configuration mode, specify the tunnel destination IPv4 or IPv6 address.	**tunnel destination** *ip-address*
4.	In tunnel configuration mode, specify the tunnel interface IPv4 or IPv6 address. Do not configure an IP address when configuring the tunnel as a VPPS L2 tunnel in step 6.	**ip address** *ip-address*
5.	In tunnel configuration mode, configure the encapsulation method (tunnel mode) for the tunnel.	**tunnel mode** {**gre** | **ipip** [**ipv6**] | **ipv6ip** [**ipv6**]}
6.	Optionally, in tunnel configuration mode, configure the tunnel as a VPPS L2 tunnel.	**tunnel mode gre l2** *port-name*
7.	Optionally, in tunnel configuration mode, configure the tunnel as a L2 tunnel bridge port (Virtual Private Ethernet Service).	**tunnel mode gre l2** *tb-port-name*
8.	Optionally, in tunnel configuration mode, configure the tunnel as a L2 GRE mirrored tunnel.	**tunnel mirror** {**enable** | **disable**}

Procedure 37-1 Tunneling Configuration (continued)

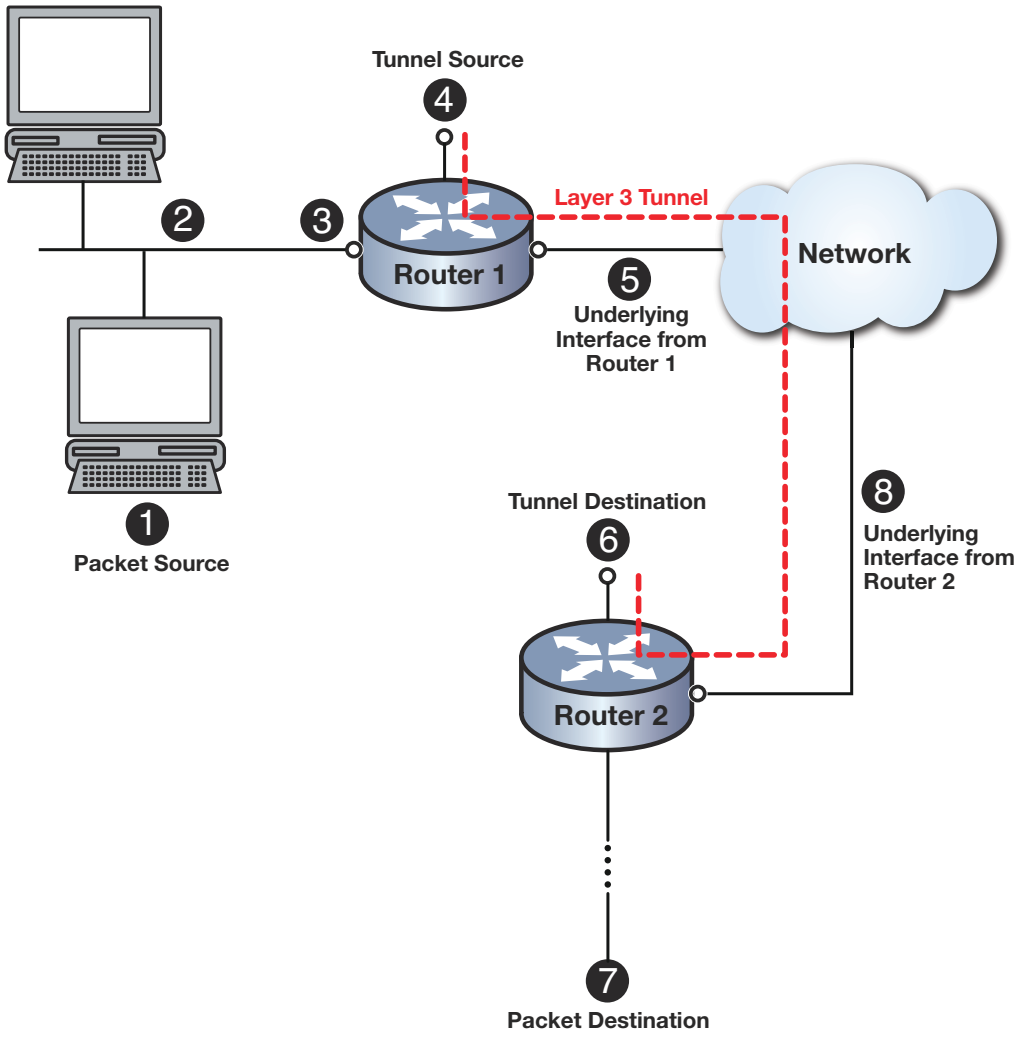
Step	Task	Command(s)
9.	In tunnel configuration mode, If the tunnel mode is GRE and IPv4 over IPv4 encapsulation will be used, optionally configure the keepalive interval and number of keepalive retries.	**tunnel keepalive** *seconds retries*
10.	In tunnel configuration mode, If the tunnel mode is GRE, optionally configure the GRE keyword.	**tunnel keyword** *keyword*
11.	In tunnel configuration mode, optionally modify the outer tunnel header ToS value.	**tunnel tos** *tos*
12.	In tunnel configuration mode, optionally configure a tunnel probe to monitor an IP address associated with the tunnel.	**tunnel probe probe-name** {**default** | *probe-name*}
13.	In global configuration mode, optionally configure an IPv4 or IPv6 static route specifying the route destination address and tunnel interface. If a static route is not configured, assure that reachability from the tunnel source to the the tunnel destination exists using a supported routing protocol.	**ip route** {*prefix mask* | *prefix*/*prefix-length*} **interface** *interface-name* [*distance*] [**tag** *tag-id*] **or** **ipv6 route** *prefix*/*length* **interface** *interface-name* [*distance*] [**tag** *tag-id*]
14.	In tunnel configuration mode, optionally enable checkspoofing to verify that the source address of the packet is reachable from the receiving interface.	**ip checkspoof** {**strict-mode** | **loose-mode**} **ipv6 checkspoof** {**strict-mode** | **loose-mode**}
15.	In tunnel configuration mode, optionally apply an IPv4 or IPv6 ACL to the interface.	**ip access-group** {*access-list-number* | *name*} {**in** | **out**} **ipv6 access-group** *name* {**in** | **out**}

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

Tunnel Configuration Example

This tunnel configuration example configures a GRE mode tunnel capable of encapsulating and transmitting an IPv6 header and payload over an IPv4 network to the tunnel destination address as depicted in Figure 37-4 on page 37-14.

Figure 37-4 Tunnel Configuration Example



1	PC 1, IP address 2111::2	**5**	Router 1 underlying tunnel interface: VLAN 50
2	Router 1 IPv6 ingress address 2111::1	**6**	Router 2, Loopback 1, IP address 99.99.99.1
3	Router 1 packet ingress interface: VLAN 11	**7**	Packet Destination, IP address 2333::2
4	Router 1, Loopback 1, IP address 88.88.88.1	**8**	Router 2 underlying tunnel interface: VLAN 50

Configuration Example Packet Transit Discussion

What follows is a discussion of how the source packet transits the network to its destination using the tunnel:

1. The packet is sourced at **PC 1** with a source IPv6 address **2111::2** and a destination IPv6 address **2333::2**. These are the original packet header source and destination addresses and will be unchanged when the original packet header is encapsulated into the outer tunnel header.
2. The packet is transmitted to Router 1 using VLAN 11.
3. The packet ingresses Router 1 on VLAN 11 at IPv6 address 2111::1. At this point a standard route table lookup occurs.

4. The route table lookup determines that the best next hop route is using a tunnel that is sourced from loopback 1 using IP address 88.88.88.1 as the source address. The original packet header and payload is encapsulated into the outer tunnel header that has a source address of 88.88.88.1 and a destination address of 99.99.99.1. In this case, the GRE tunnel is functioning as an IPv6 over IPv4 tunnel.
5. The route table lookup determines that the underlying interface for the tunnel is VLAN 50, because a static route exists for VLAN 50 specifying the tunnel destination address 99.99.99.1 as its route destination.
6. The tunnel encapsulated packet is transmitted to the tunnel destination address 99.99.99.1 as a logical single hop from the point of view of the original encapsulated packet header. At the tunnel destination, the outer tunnel header is removed and routing lookup determines the next hop, based upon the best next hop to the destination address of the original packet header.
7. The packet is routed, using a standard route lookup, however many hops required to get to the packet destination.
8. A returning packet that is routed over the tunnel will use the tunnel underlying interface from the point of view of Router 2 when transiting the tunnel. In this case, the initial underlying interface for the tunnel is VLAN 50.

Configuration Example CLI Input

This tunnel configuration example provides the CLI input for both Router 1 and Router 2 as displayed in Figure 37-4 on page 37-14:

Router 1

1. Configures loopback interface 1 with an IP address of 88.88.88.1, to be used as the source for the tunnel from the perspective of Router 1
2. Creates tunnel 1 (tun.0.1) configured for:
   - GRE mode
   - Source address **88.88.88.1**
   - Destination address **99.99.99.1**
   - Tunnel interface IPv6 address **2111::10/64**
   - A default tunnel probe to monitor the tunnel destination address **99.99.99.1**
   - A GRE keyword of **123456**
3. Establishes reachability with the tunnel destination address using a static route with the tunnel destination address **99.99.99.1** as the route destination over VLAN **50**
4. Configures an IPv6 static route to prefix **2333::0/64** over tunnel 1

```
S Chassis(su)->configure
S Chassis(su-config)->interface loopback 1
S Chassis(su-config-intf-loop.0.1)->ip address 88.88.88.1 255.255.255.255 primary
S Chassis(su-config-intf-loop.0.1)->no shutdown
S Chassis(su-config-intf-loop.0.1)->exit
S Chassis(su-config)->interface tunnel 1
S Chassis(su-config-tun.0.1)->tunnel mode gre
S Chassis(su-config-tun.0.1)->tunnel source 88.88.88.1
S Chassis(su-config-tun.0.1)->tunnel destination 99.99.99.1
```

```
S Chassis(su-config-tun.0.1)->ipv6 address 2111::10/64
S Chassis(su-config-tun.0.1)->tunnel probe 99.99.99.1 probe-name default
S Chassis(su-config-tun.0.1)->tunnel keyword 123456
S Chassis(su-config-tun.0.1)->no shutdown
S Chassis(su-config-tun.0.1)->exit
S Chassis(su-config)->ip route 99.99.99.1/32 vlan 50
S Chassis(su-config)->ipv6 route 2333::0/64 interface tun.0.1
```

Router 2

1. Configures loopback interface 1 with an IP address of 99.99.99.1, to be used as the source for the tunnel from the perspective of Router 2
2. Creates tunnel 1 (tun.0.1) configured for:
   - GRE mode
   - Source Address **99.99.99.1**
   - Destination Address **88.88.88.1**
   - Tunnel interface IP address **2.2.2.2**
   - A default tunnel probe to monitor the tunnel destination address **88.88.88.1**
   - A GRE keyword of **123456**
3. Establishes reachability with the tunnel destination address using a static route with the tunnel destination address **88.88.88.1** as the route destination over VLAN **50**
4. Configures an IPv6 static route to prefix **2111::0/64** over tunnel 1

```
S Chassis(su)->configure
S Chassis(su-config)->interface loopback 1
S Chassis(su-config-intf-loop.0.1)->ip address 99.99.99.1 255.255.255.255 primary
S Chassis(su-config-intf-loop.0.1)->no shutdown
S Chassis(su-config-intf-loop.0.1)->exit
S Chassis(su-config)->interface tunnel 1
S Chassis(su-config-tun.0.1)->tunnel mode gre
S Chassis(su-config-tun.0.1)->tunnel source 99.99.99.1
S Chassis(su-config-tun.0.1)->tunnel destination 88.88.88.1
S Chassis(su-config-tun.0.1)->ipv6 address 2111::10
S Chassis(su-config-tun.0.1)->tunnel probe 88.88.88.1 probe-name default
S Chassis(su-config-tun.0.1)->tunnel keyword 123456
S Chassis(su-config-tun.0.1)->no shutdown
S Chassis(su-config-tun.0.1)->exit
S Chassis(su-config)->ip route 88.88.88.1/32 vlan 50
S Chassis(su-config)->ipv6 route 2111::0/64 interface tun.0.1
```

Terms and Definitions

Table 37-1 lists terms and definitions used in this tunnel configuration discussion.

Table 37-1 Tunneling Configuration Terms and Definitions

Term	Definition
Generic Routing Encapsulation (GRE)	A tunnel mode that supports all combinations of IP tunnel encapsulation.
Tunnel	The use of network layer tunneling protocols to connect disjoint networks within the same (trusted) enterprise campus network, resulting in the destination address of the tunnel functioning as a logical next hop.
Payload	The original packet data and header that gets encapsulated into the tunnel outer header.
Virtual Private Port L2 tunnel	Virtual Private Ports permit the user to extend a virtual wire through an arbitrary routed network using GRE with transparent bridging. Any packets arriving on the tunnel physical port are immediately encapsulated and routed to the other end of the tunnel where the packets are de-encapsulated and sent out the end-tunnel physical port.
Tunnel Destination Address	The destination IP address used by the outer encapsulating header as the packet transits the tunnel.
Tunnel Keepalive	A means of monitoring both whether the tunnel endpoint is up and whether the packet has been decapsulated at the tunnel endpoint.
Tunnel Keyword	A GRE tunnel mode supported numeric password scheme.
Tunnel Mode	Specifies the encapsulation type(s) supported by the tunnel as GRE (any IP type combination) or a specific original packet IP type header over the tunnel IP type header.
Tunnel Probe	A means of monitoring whether the tunnel endpoint of any tunnel mode type is up.
Tunnel Source Address	The source IP address used by the outer encapsulating header as the packet transits the tunnel.

38

Layer 3 Virtual Private Network (VPN) Configuration

This chapter provides information about configuring and monitoring Layer 3 VPN on S-Series devices.

How to Use Layer 3 VPN in Your Network

The Layer 3 Virtual Private Network (L3 VPN) extends a private data network using a public IP infrastructure as a conduit for connecting sites by means of Native MPLS, L3 tunneling or SPBV. L3 VPN uses internal multi-protocol BGP (MP-iBGP) to carry VPN routes and labels. Forwarding between VPN sites is done using Native MPLS, MPLS in IP tunneling, GRE encapsulation, or SPBV methods for both IPv4 and IPv6 VPN address families. Public infrastructure is defined as a single backbone core enterprise network connecting various businesses such as airport services or stores within a shopping mall.

VPN services are based upon the L3 VPN open standard RFC 4364 *BGP/MPLS IP Virtual Private Networks (VPNs)*.

A L3 VPN can be established directly between VRFs across a campus LAN. This is referred to as the VRF-lite model. In this model no additional encapsulation is required. Scaling is limited in this model to 16 IGP protocol VRF instances and a total of 128 VRFs per router. A VLAN interface is assigned to a single VRF. Internet access and services can be either separate or shared using the global VRF router instance. In a VRF-lite model, all routers in the routing domain must be VRF aware of each endpoint VRF that will use the router. Core routers in the domain quickly use up the 16 IGP protocol maximum allowed. Limited scaling makes the VRF-lite model only viable for small enterprise networks.

The scaling issue inherent in the VRF-lite model can be overcome using L3 tunneling, Native MPLS, or SPBV between the global VRFs of Provider Edge (PE) routers at the edge of the enterprise core. The remainder of the discussion in this chapter relates to L3 VPNs using L3 tunnels, Native MPLS, or SPBV.

L3 VPN using L3 Tunnels or Native MPLS

Using L3 tunnels, routers in the enterprise core are no longer part of the VRF configuration. The core routers transparently forward L3 VPN traffic to the tunnel endpoint using static routes or an IGP such as OSPF. The PE router uses a tunnel interface per BGP peer and encapsulates L3 VPN data as defined in RFC 4023 *Encapsulating MPLS in IP or Generic Routing Encapsulation (GRE)*.

The Native MPLS encapsulation method implements the complete L3 VPN solution by replacing the need for tunnel interface in the L3 VPN configuration when MPLS is enabled on the routing interface. The MPLS label assigned by the MPLS router using the Label Distribution Protocol (LDP) contains the egress router path. All routers within the backbone must be MPLS capable routers when using Native MPLS. The Label Switch Router (LSR) uses the MPLS label to forward packets within the tunnel to the VPN egress router.

The required fully integrated services on the PE router for either a L3 tunnel or Native MPLS VPN network to establish are:

- VRF instance – Virtual Routing and Forwarding (VRF) provides for partitioning a router into segregated domains for the routed forwarding of packets.
- Route distinguisher (RD) – A 64 bit identifier prepended to the IP address making the address globally unique across the L3 VPN network and when stored in the MP-iBGP route table
- Route target – An identifier that determines which routes are advertised by a VRF and inserted into a VRF
- A L3 tunnel (VPN using L3 tunnel) or MPLS enabled (VPN using Native MPLS)
- For a L3 VPN using Native MPLS, enable MPLS LDP on all routers in the L3 VPN domain
- IGP (OSPF) or static routes providing reachability for all LSRs within the tunneled domain
  - (VPN using L3 tunnels only) Force BGP traffic to use the tunnel interface by creating a static route with the remote BGP address as destination, so that the remote peer's loopback address prefers the tunneled interface over the VLAN interface as the next-hop
- MP-iBGP – An internal multi-protocol border gateway routing protocol that carries the VPN routes and labels within a single autonomous system

There are three router types in a typical L3 VPN network using L3 tunnels:

- Provider Edge (PE) router – These routers are the endpoints of the public network that:
  - Are configured for a VRF for each L3 VPN
  - Interconnect with other PE routers using L3 tunnels or Native MPLS configured in the global VRF
  - Directly connect and redistribute learned BGP L3 VPN routes to private customer routers over L2 and L3 links.
- Provider Core (P) router – These routers reside in the public core and provide core routing using an IGP such as OSPF unless the PEs are directly connected. VPN traffic transparently passes through the P router over the L3 VPN.
- Customer Edge (CE) router – These routers reside in the private customer network. When routes are learned on the PE from local CEs, they are redistributed to other PEs using IBGP.

Imported L3 VPN routes learned on the PE router VRFs are redistributed to the customer edge routers using BGP.

Figure 38-1 on page 38-3 provides an overview of two L3 tunnel VPN networks, one for Customer A and one for Customer B, configured on two PE routers, with L3 tunneling providing the connectivity across the public core network.

Each PE router is configured for two VRFs, one for each L3 VPN. The L3 VPN for Customer A uses VRFs **VRF1** and **VRF3**. The L3 VPN for Customer B uses **VRF2** and **VRF4**. For a L3 VPN to operate, each VRF must be configured with an RD. Each VRF must be configured with at least one route target that imports or exports L3 VPN routes, or both imports and exports L3 VPN routes. Each PE peering must be configured with a L3 tunnel on the global VRF. An IGP protocol such as OSPF or static routes must be configured to provide reachability between all LSRs within the tunneled domain. Configure a static route with the remote BGP address as the destination, so that the remote peer's loopback address prefers the tunneled interface as the next-hop over the VLAN interface the tunnels use for BGP information. BGP must be configured at the global VRF to redistribute routes from each VRF to the linked CE router and to forward L3 VPN traffic over the L3 tunnels. The appropriate IPv4 or IPv6 BGP L3 VPN address family must be enabled. Within the appropriate BGP L3 VPN address family, activate each BGP neighbor.

Figure 38-1 Layer 3 VPN Using L3 Tunneling Overview

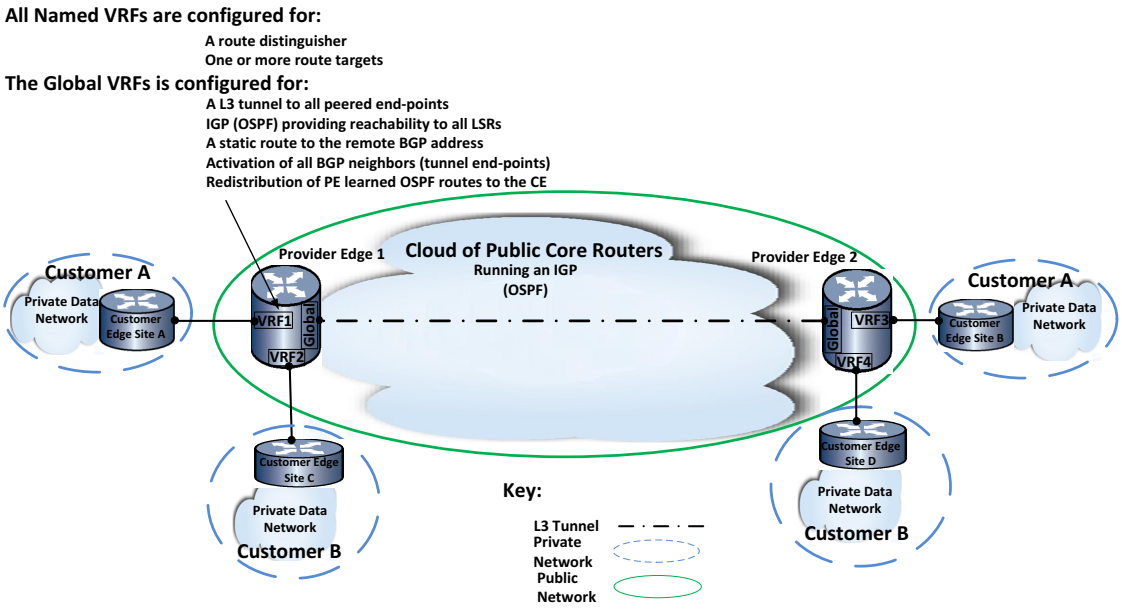
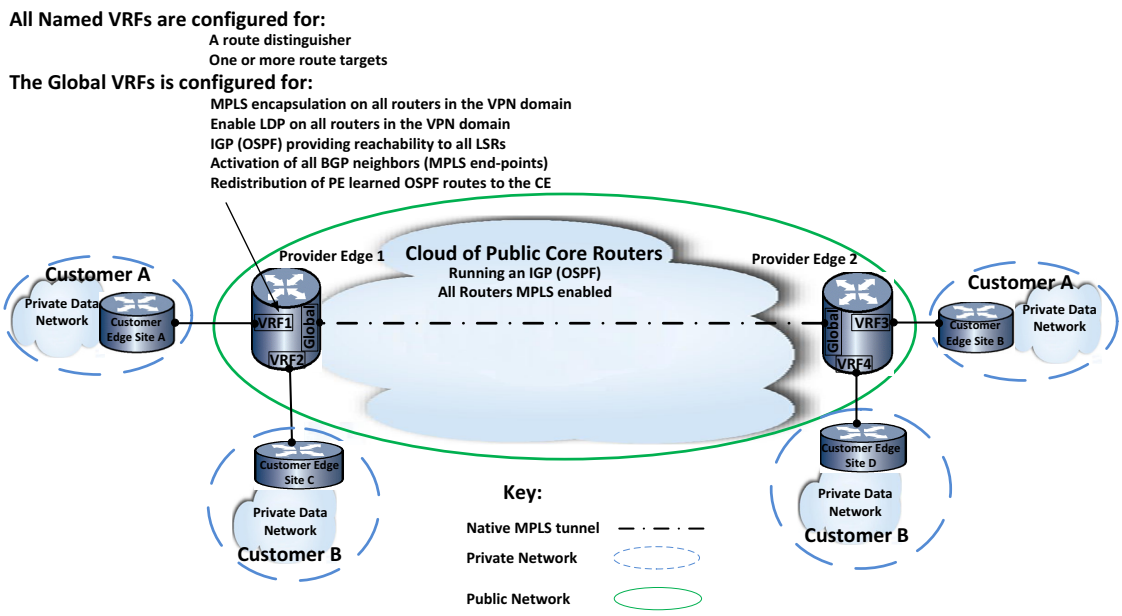


Figure 38-2 on page 38-4 provides an overview of a L3 Native MPLS VPN network, with one Native MPLS tunnel for Customer A and one for Customer B, configured on two PE routers, with L3 Native MPLS tunneling providing the connectivity across the public core network.

Each PE router is configured for two VRFs, one for each L3 VPN. The L3 VPN for Customer A uses VRFs **VRF1** and **VRF3**. The L3 VPN for Customer B uses **VRF2** and **VRF4**. For a L3 VPN to operate, each VRF must be configured with an RD. Each VRF must be configured with at least one route target that imports or exports L3 VPN routes, or both imports and exports L3 VPN routes. Each PE and Core router in the L3 MPLS domain must be enabled for MPLS encapsulation at the global VRF. An IGP protocol such as OSPF or static routes must be configured to provide reachability between all LSRs within the tunneled domain. On each PE router, BGP must be configured at the global VRF to redistribute routes from each VRF to the linked CE router and to forward L3 VPN traffic over the L3 MPLS tunnels. The appropriate IPv4 or IPv6 BGP L3 VPN address family must be enabled. Within the appropriate BGP L3 VPN address family, activate each BGP neighbor.

Figure 38-2 Layer 3 VPN Using Native MPLS Overview



L3 VPN over SPBV

Using SPBV, the PE router is directly connected to its BGP peers, so an IGP (OSPF) is not needed. The required fully integrated services on the PE router for an SPBV network to establish are:

- VRF instance – Virtual Routing and Forwarding (VRF) provides for partitioning a router into segregated domains for the routed forwarding of packets.
- Route distinguisher (RD) – A 64 bit identifier prepended to the IP address making the address globally unique across the L3 VPN network and when stored in the MP-iBGP route table.
- Route target – An identifier that determines which routes are advertised by a VRF and inserted into a VRF.
- SPBV – A L2 protocol that assures data traffic transits a shortest cost path between any two switches in the SPBV region. SPBV is configured on all devices in the SPBV cloud including PE routers, and is enabled on all PE Global router ports.
- MP-iBGP – An internal multi-protocol border gateway routing protocol that carries the VPN routes and labels within a single autonomous system

There are two router types in a typical L3 VPN over SPBV network:

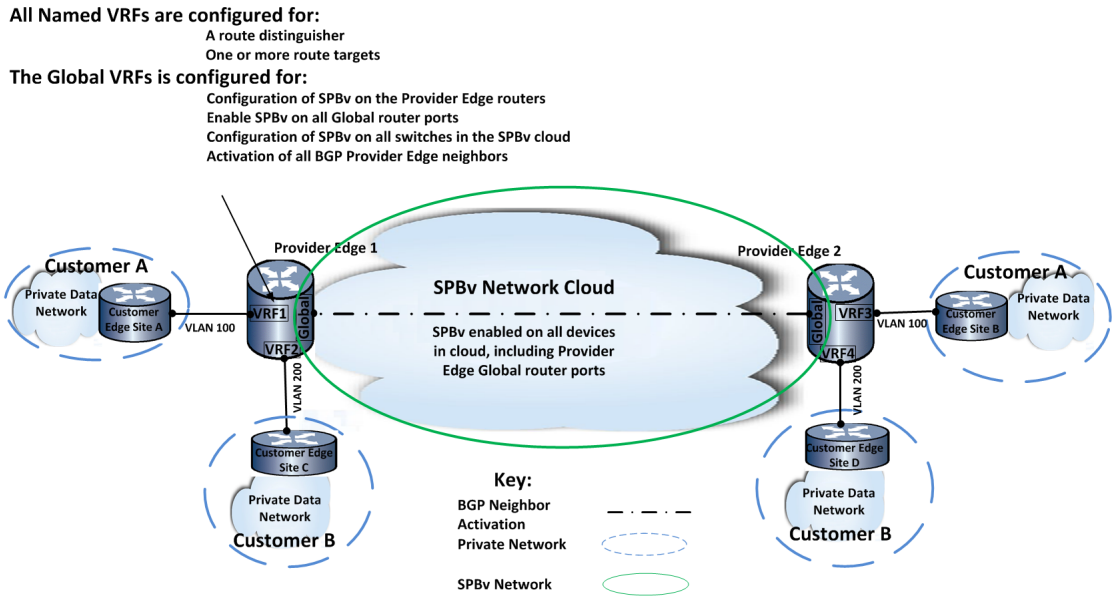
- Provider Edge (PE) router – These routers are the endpoints of the public network that:
  - Are configured for a VRF for each L3 VPN
  - Interconnect with other PE routers using SPBV configured in the global VRF
  - Directly connect and redistribute learned BGP L3 VPN routes to private customer routers over L2 and L3 links.
- Customer Edge (CE) router – These routers reside in the private customer network. When routes are learned on the PE from local CEs, they are redistributed to other PEs using IBGP.

  Imported L3 VPN routes learned on the PE router VRFs are redistributed to the customer edge routers using BGP.

Figure 38-3 on page 38-5 provides an overview of a L3 VPN over SPBV network. There are two L3 VPNs: Customer A on base-VLAN 100 and Customer B on base-VLAN 200.

Each PE router is configured for two VRFs, one for each L3 VPN. The L3 VPN for Customer A uses VRFs **VRF1** and **VRF3**. The L3 VPN for Customer B uses **VRF2** and **VRF4**. For a L3 VPN to operate, each VRF must be configured with an RD. Each VRF must be configured with at least one route target that imports or exports L3 VPN routes, or both imports and exports L3 VPN routes. Each PE must be configured for SPBV with SPBV enabled on all Global router ports. All devices within the SPBV cloud must be configured for SPBV. On each PE router, BGP must be configured at the global VRF to redistribute routes from each VRF to the linked CE router. The appropriate IPv4 or IPv6 BGP L3 VPN address family must be enabled. Within the appropriate BGP L3 VPN address family, activate each BGP neighbor.

Figure 38-3 Layer 3 VPN over SPBv Overview



Implementing Layer 3 VPN using L3 Tunneling

Do the following to implement L3 VPN using L3 tunneling in your network:

- Configure the VRF for the L3 VPN
- Optionally, configure MPLS label mode to allocate a unique label for each prefix route (defaults to a single label for the named VRF)
- Configure the route distinguisher for each VRF associated with a L3 VPN
- Configure one or more route targets to identify the L3 VPN routes to import and export for each VRF associated with a L3 VPN
- Optionally, specify a VPN identifier to uniquely identify this VPN to which a packet flow belongs to other network features outside of the VPN
- On the global VRF, configure the routed interfaces:
  - A loopback interface with an IPv4 or IPv6 address
  - One or more VLAN interfaces
  - A L3 tunnel interface to each VPN edge router for each BGP peering session

- In global VRF mode, configure an IGP (OSPF) or static routes allowing all LSRs to be reachable within the tunneled domain
- In global VRF mode, configure a static route with the remote BGP address as the destination so that the remote peer loopback address prefers the tunneled interface as the next-hop and not the VLAN interfaces the tunnel uses.
- In global VRF mode, configure BGP to propagate routes from the VRF routing protocol tables
  - Enable the appropriate (IPv4 or IPv6) BGP L3 VPN address family
  - Within the appropriate BGP L3 VPN address family, activate BGP neighbors
  - In non-L3 VPN global BGP address family configuration mode, redistribute routes to the CE routers

Implementing Layer 3 VPN using Native MPLS Tunneling

Do the following to implement L3 VPN using Native MPLS tunneling in your network:

- Configure the VRF for the L3 VPN
- Optionally, configure MPLS label mode to allocate a unique label for each prefix route (defaults to a single label for the named VRF)
- Configure the route distinguisher for each VRF associated with a L3 VPN
- Configure one or more route targets to identify the L3 VPN routes to import and export for each VRF associated with a L3 VPN
- Optionally, specify a VPN identifier to uniquely identify this VPN to which a packet flow belongs to other network features outside of the VPN
- In global configuration mode on the global VRF, enable MPLS encapsulation for IPv4 or IPv6 routing
- In global configuration mode on the global VRF, enable LDP as the active label distribution protocol on all MPLS enabled routers
  - Optionally, filter label allocation to BGP or to host routes only
  - Optionally, change the LDP retention mode to retain label mappings only if they will be used to explicitly forward packets to the next-hop
- In global VRF mode, configure the routed interfaces:
  - A loopback interface with an IPv4 or IPv6 address
  - One or more VLAN interfaces
  - Enable MPLS encapsulation on each interface used by the L3 VPN
- In global VRF mode, configure an IGP (OSPF) or static routes allowing all LSRs to be reachable within the tunneled domain
- In global VRF mode, enable the propagation of TTL from IPv4 and IPv6 headers to the MPLS label for forwarded packets, local packets, or both
- In global VRF mode, configure BGP to propagate routes from the VRF routing protocol tables
  - Enable the appropriate (IPv4 or IPv6) BGP L3 VPN address family
  - Within the appropriate BGP L3 VPN address family, activate BGP neighbors
  - In non-L3 VPN global BGP address family configuration mode, redistribute routes to the CE routers

Implementing Layer 3 VPN over SPBV

Do the following to implement L3 VPN over SPBV in your network:

- Configure the VRF for the L3 VPN
- Configure the route distinguisher for each VRF associated with a L3 VPN
- Configure one or more route targets to identify the L3 VPN routes to import and export for each VRF associated with a L3 VPN
- Optionally, specify a VPN identifier to uniquely identify this VPN to which a packet flow belongs to other network features outside of the VPN
- Configure SPBV on all PEs and all devices within the SPBV region (see Chapter 22, **Shortest Path Bridging (SPB) Configuration** for SPBV configuration details):
  - Configure Spanning Tree on all devices in the region by: setting the Spanning Tree version to SPT on all devices in the SPB region and configuring the same MST configuration name on all devices in the SPB region
  - Configure an SPVID pool for this SPB region (same VLAN range for all devices in the region).
  - Enable SPB on all ports that will take part in the SPB region, including the Provider Edge Global router ports.
  - Assign the base-VLANs that will be used to ingress and egress the SPB region to SID **4093** or SID **spbv**.
  - Optionaly, assign the same ECT algorithm for a given SPBV region to each configured base-VLAN (unless the default algorithm is desired).
  - Optionally, administratively assign the base-VLAN to SPVID mapping for the base-VLAN on each device in the SPB region. When administratively assigning the base-VLAN to SPVID mapping, change the SPB VLAN mode to manual.
- In global VRF mode, configure BGP to propagate routes from the VRF routing protocol tables
  - Enable the appropriate (IPv4 or IPv6) BGP L3 VPN address family
  - Within the appropriate BGP L3 VPN address family, activate BGP neighbors

Layer 3 VPN Overview

This section discusses each network component required to establish and operate a L3 VPN using L3 tunnels.

PE Router Overview

PE routers are located at the edge of the public network and interface with the CE router using an IGP such as OSPF. The PE router is aware of each customer's VPNs and associated network prefixes. Each PE maintains separate routing tables that are completely independent of each other. The routing table belonging to a specific customer site resides in the customer VRF. This separation allows duplicate addresses among the various VPN customers and eliminates routing ambiguity by applying an RD specific to each VPN to each prefix in the routing table. Aspects of the PE router L3 VPN configuration are performed in both the VRF instance for the L3 VPN and the global VRF.

The Virtual Routing and Forwarding (VRF) Instance

The VPN allows an enterprise to maintain data privacy when transmitted over a public network. The VRF provides a separate routing domain within the public PE router for that enterprise by partitioning the router into segregated routing domains for the forwarding of packets. A VRF dedicated to a L3 VPN is a requirement, but the naming of that VRF has descriptive significance only. Refer to Chapter 35, **Virtual Routing and Forwarding (VRF) Configuration** for a detailed discussion on how to create and configure a VRF.

Once the VRF is created, within that VRF:

- Configure any interfaces required to attach to the CE router.
- Configure the IGP instance, such as OSPF, to communicate with the attached CE router.
- Configure the RD for purposes of uniquely identifying the IP name-space and also required to handle overlapping IP addresses (See "The Route Distinguisher (RD)" on page 38-9).
- Configure route targets that define policies for importing and exporting VPN addresses for this VRF (See "The Route Target" on page 38-10).
- Optionally, modify the MPLS label mode to allocate a unique MPLS label for each prefix route in the routing table (Defaults to one MPLS label per VRF). See "MPLS Label Mode" on page 38-14.

The Global VRF

PE global VRF networking components:

- Loopback interface with IPv4 or IPv6 address
- VLAN interfaces used by the L3 VPN that point towards the public core or PE neighbor routers
- Enable LDP for either IPv4 or IPv6 as the label distribution protocol
- If using L3 tunneling: L3 Tunnel – Refer to "The L3 Tunnel" on page 38-10 for a L3 tunnel discussion
- If using Native MPLS tunneling: Enable MPLS encapsulation both in global configuration mode and for each interface used by L3 VPN (see "Native MPLS" on page 38-11)
- If using L3 tunneling: configure static routes to each remote peer loopback address so it will be preferred as the next-hop of the tunnel
- Configure the IGP such as OSPF, used in the public core
- L3 VPN BGP elements – Refer to "Multi-protocol Internal BGP" on page 38-13 for the BGP configuration relating to the L3 VPN

The Route Distinguisher (RD)

The route distinguisher is a 64 bit identifier attribute that gets prepended to the user IPv4 or IPv6 address and makes the IP address globally unique across the VPN network and within the BGP routing table. The RD is a required component when defining a L3 VPN. Its significance is local to the device. Assign one RD to each VRF that will use the L3 VPN. The BGP VPN-IPv4 or VPN-IPv6 address families are defined by combining the RD, user IP address, and the MPLS Label (see "MPLS Label Mode" on page 38-14 for label mode configuration).

RDs must be unique for each L3 VPN VRF on a device. The same RD can be used on multiple devices belonging to the VPN. Combining the VRF RD with the user IP address, even when that IP address is an unregistered private address, serves to uniquely identify the user.

Three data fields make up the eight bytes (64-bits) of the RD attribute:

- **RD Type** – A non-configurable two-byte field that identifies the format used by the administrator and assigned fields as the packet transits the network. Valid values are 0, 1, or 2.
- **Administrator Field** – A two- or four-byte field (depending upon the RD type) allowing a network administrator to uniquely identify the VRF as a:
  - Two-byte autonomous system number (RD type 0). Valid values are **1** - **65535**.
  - Four-byte IPv4 address (RD type 1)
  - Four-byte autonomous system number (RD type 2). Valid values are **65536** - **4294967295**.
- **Assigned Number Field** – A two- or four-byte field (depending upon the RD type) assigned by the provider network:
  - Four-byte autonomous system number (RD type 0). Valid values are **1** - **4294967295**.
  - Two-byte autonomous system number (RD types 1 and 2). Valid values are **1** - **65535**.

It is recommended that non-private autonomous system numbers be used when configuring the RD. If the BGP autonomous system number is a private AS between 64512-65534, use RD type 1 specifying an IPv4 address.

Non-private autonomous system numbers are assigned by IANA to service providers. Non-private autonomous system numbers use either a two-byte or four-byte number in the following formats:

- Type 0 – **1** - **65535:1** - **4294967295**
- Type 1 – *IPv4-address*:**1** - **65535**
- Type 2 – **65536** - **4294967295:1** - **65535**

This example shows how to assign a type 0 route distinguisher **1:52** to VRF **vpnA**:

```
S Chassis(rw)->router vpnA
S Chassis(su-vpnA)->configure
S Chassis(su-vpnA-config)->rd 1:52
S Chassis(su-vpnA-config)->
```

This example shows how to assign a type 1 route distinguisher **10.10.100.1:53** to VRF **vpnB**:

```
S Chassis(rw)->router vpnB
S Chassis(su-vpnB)->configure
S Chassis(su-vpnB-config)->rd 10.10.100.1:53
S Chassis(su-vpnB-config)->
```

The Route Target

The route target determines which routes are inserted into a VRF. A VRF can be configured for one or more route targets for import, export, or both. At least one configured route target for import or export is a required component when defining a L3 VPN VRF. All routes exported by the VRF are tagged with each route target identifier configured for export on the VRF. Only VRFs configured to import routes tagged with the route target identifier will import the route. This allows you to configure one VRF to export multiple route targets and another VRF to be configured to import only a subset of the routes the first VRF exports.

- An **export** route target – BGP advertises VPN-IPv4 and VPN-IPv6 address family prefixes, along with extended community names and tags the advertisement with the route target identifier. A redistribute rule must be created under the appropriate IPv4 or IPv6 address family in the BGP global configuration mode for each routing protocol, static, or connected route to be exported. See the address-family command entry in the "Border Gateway Protocol Commands" chapter of the *Extreme Networks S-Series CLI Reference* for the BGP global configuration mode address family command details.
- An **import** route target – Import route targets specify that this VRF will import any BGP advertised routes that are tagged with the specified route target identifier, updating the VRF routing and forwarding tables with the advertised VPN-IPv4 or VPN-IPv6 addresses. IPv4 to IPv6 tunneling which supports IPv4 over an IPv6 Layer 3 tunnel interface as defined in RFC 2473 is not supported for L3 VPNs. When the VRF BGP router receives an update, it examines the extended community names for each set of prefixes. If an update matches a configured import route target for this named VRF, BGP installs the matching set of prefixes into the routing and forwarding tables as BGP learned routes, after removing the 64-bit RD.
- **Both** an import and export route target – This VRF will both import routing updates that match configured import route targets and export VPN address family prefixes tagged with the specified route target(s).

This example shows how to export VPN address family prefixes and tag them with route target **1:1000**:

```
S Chassis(su)->router vpnA
S Chassis(su-vpnA)->configure
S Chassis(su-vpnA-config)->route-target export 1:1000
S Chassis(su-vpnA-config)->
```

This example shows how to both import BGP VPN updates tagged with the route target **10.10.176.25:1000** and tag any BGP VPN advertisements with the route target **10.10.176.25:1000**:

```
S Chassis(su)->router vpnA
S Chassis(su-vpnA)->configure
S Chassis(su-vpnA-config)->route-target both 10.10.176.25:1000
S Chassis(su-vpnA-config)->
```

The L3 Tunnel

Layer 3 tunneling uses network layer tunneling protocols to connect the PE with each of its VPN peers in the public network with matching VPN requirements, resulting in the destination address of the Layer 3 tunnel (The PE neighbor) functioning as a logical next hop.

Tunneling involves the use of a tunnel protocol that encapsulates the payload of the packet entering the tunnel within another (outer) header based upon tunnel parameters. Thus a tunneled packet has an inner and an outer header. The inner header contains the original packet header. The IP type (IPv4 or IPv6) of the original header is determined by the original packet source and destination address type. The outer delivery header is the tunnel header. The IP type of the tunnel

header is determined by the route lookup source and destination IP address type configured for the tunnel. The L3 tunnel prevents and P router in the public core from having any knowledge of the VPN labels. P routers use the destination address of the outer IP header to forward packets. Only the PE router defines the tunnel interfaces and support VPN's MPLS label.

The S-Series platform supports Layer 3 tunneling modes:

- Generic Routing Encapsulation (GRE) which provides generic support for all supported IPv4 and IPv6 tunnel IP type combinations, as defined in RFC 2784, along with the keyword extensions defined in RFC 2890. The GRE mode should be used if you do not want to limit the tunnel to a specific IP header combination. This implementation does not support RFC 1701.
- IP-IP tunneling which provides support for IPv4 over an IPv4 Layer 3 tunnel interface as defined in RFC 2003.
- IPv6 tunneling which provides support for IPv6 over an IPv6 Layer 3 tunnel interface as defined in RFC 2473.
- IPv4 to IPv6 tunneling which supports IPv4 over an IPv6 Layer 3 tunnel interface as defined in RFC 2473.
- IPv6 to IPv4 tunneling which supports IPv6 over an IPv4 Layer 3 tunnel interface as defined in RFC 2473.

A Layer 3 tunnel interface can be assigned to a static route using the **ip route** or **ipv6 route** command, depending upon the route IP type. The Layer 3 tunnel source and destination must be reachable either by a configured static route or a supported routing protocol such as RIP, BGP, or OSPF.

See the **interface** command entry, in the *Extreme Networks S-Series CLI Reference*, for create, enable, and disable Layer 3 tunnel command details. Refer to Chapter 37, **Tunneling Configuration** for L3 tunnel configuration details.

Native MPLS

For a L3 VPN using Native MPLS, MPLS encapsulation must be enabled on all routers on the Label Switched Path (LSP) between Label Edge Routers (LER). When MPLS encapsulation is enabled, an MPLS label stack follows the Ethernet header and contains an outer label path to the egress VPN router and an inner label identifying the VPN. The outer label egress VPN router path is assigned to the MPLS router by the Label Distribution Protocol (LDP) and is used by the receiving Label Switch Router (LSR) to determine the next hop on the LSP. The LSR removes the MPLS label from the header and replaces it with a new label before the packet is forwarded to the next LSR in the LSP.

You enable MPLS encapsulation in the global configuration mode of the global VRF.

Use the **mpls ip** command to enable IPv4 MPLS encapsulation in global configuration mode.

In interface configuration mode, the **mpls ip** command enables MPLS encapsulation for both IPv4 and IPv6 on the interface.

Use the **mpls ipv6** command, specifying the IPv6 transport address, to enable IPv6 MPLS encapsulation in global configuration mode.

L3 VPN Using Native MPLS LDP

L3 VPN using Native MPLS uses LDP as the label distribution protocol and must be enabled on all routers in the L3 VPN tunneling domain. Specify the IPv4 address type if LDP will be used in an IPv4 network. Specify the IPv6 address type if LDP will be used in an IPv6 network.

LDP discovers its distribution peers by broadcasting an HELLO message via UDP to a well-known port in the network. After discovering its peers, LDP proceeds to form sessions with each peer using TCP. Operational modes are negotiated at the time of session establishment with the resultant sessions used to distribute label mapping data.

Label distribution mode can be broken down into two functional areas:

- MPLS LDP label retention
- MPLS LDP Label distribution control

Use the **mpls label-protocol-ldp** command in global configuration mode on the global VRF to enable LDP on the router.

MPLS LDP Label Retention Mode

The MPLS LDP label retention mode specifies under what conditions the label mappings advertised by any peer will be kept. There are two modes:

- **Conservative** – Advertised label mappings are retained only if used to explicitly forward packets to their next-hop. These mappings are received from a valid next-hop router.
- **Liberal** – All advertised label mappings from each peer LSR are kept regardless of whether the peer is a next-hop router or not.

An advantage to using conservative label retention mode is that only labels required for packet forwarding to next-hops are allocated and maintained. An operational disadvantage of conservative mode is that a new label must be obtained from the new next-hop router if routing changes the next-hop for a given destination.

The main advantage of the Liberal Label retention mode is that reaction to routing changes is swift, since labels needed for such changes already exist and maintained. The disadvantage of liberal mode is the large amount of unneeded label mappings that are routinely maintained and distributed.

MPLS LDP label retention mode defaults to **liberal**.

Use the **mpls ldp-label-retention-mode** command in global configuration mode on the global router to change the MPLS LDP label retention mode for the router.

MPLS LDP Label Distribution Control

The S-Series platform supports ordered distribution control mode as defined in RFC 5036. In ordered distribution control mode, an LSR may initiate the transmission of a label mapping only for a FEC for which it has a label mapping for the FEC next hop, or for which the LSR is the egress. If neither of these conditions holds, the LSR must wait until a label from a downstream LSR is received before mapping the FEC and passing corresponding labels to upstream LSRs.

The ordered approach helps to provide loop prevention, but at the cost of requiring a longer amount of time to set up a label switched path.

MPLS LDP label distribution control is not administratively configurable in this release and is hard coded for ordered distribution control mode.

The LDP LSR ID

An LDP LSR ID is configured to identify the LDP instance. In an IPv4 network, the related LDP LSR ID is automatically set to the highest IPv4 address associated with the router interface, with loopback addresses taking precedence over VLAN addresses. You must explicitly configure an LDP LSR ID for an IPv6 network.

Use the **mpls ldp-lsr-id** command in global router configuration mode to configure an LDP LSR ID. The LDP LSR ID is configured in the a.b.c.d format the same as an IPv4 address. The LDP LSR ID is not an address. It is used as an identifier of the LDP instance.

Multi-protocol Internal BGP

This section details a few considerations specific to the L3 VPN configuration.

BGP is configured on the global VRF. Configure BGP for your network as you would if you were not configuring L3 VPN.

The VPN address family for the IP type you are using must be enabled. Enter the VPN address family mode for the IP type you are using on the VPN using the **address-family vpnv4** or **address-family vpnv6** command. In the appropriate VPN address family configuration mode, use the **enable** command to enable the address family.

BGP peers within either the IPv4 or IPv6 L3 VPN address family must be administratively activated using the **neighbor activate** command.

BGP routes associated with the neighbor must be redistributed to the CE router on the VRF that will use the L3 VPN. To redistribute routes learned from the PE router neighbor for this VRF, enter the BGP global configuration address family using the **address-family ipv4 vrf** command or the **address-family ipv6 vrf** command. Once in the BGP global configuration address family mode, use the appropriate **redistribute** command for the routes to be redistributed to the CE router.

This example shows how to:

- Configure BGP neighbor 100.10.10.5 and set the neighbor source to the BGP router ID
- Enter the L3 VPN IPv4 address family
- Enable the L3 VPN IPv4 address family
- Activate the peer for the IPv4 L3 VPN address family
- Enter the BGP global configuration IPv6 address family for the **vr1** VRF
- Redistribute the IPv4 static routes for VRF vr1

```
S Chassis(rw)->
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 100.10.10.1
S Chassis(su-config-bgp)->neighbor 100.10.10.5 remote-as 65151
S Chassis(su-config-bgp)->neighbor 100.10.10.5 update-source 100.10.10.1
S Chassis(su-config-bgp)->address-family vpnv4
S Chassis(su-config-bgp-af-vpn)->enable
S Chassis(su-config-bgp-af-vpn)->100.10.10.5 neighbor activate
S Chassis(su-config-bgp-af-vpn)->exit
S Chassis(su-config-bgp)->address-family ipv4 vrf vr1
S Chassis(su-config-bgp-af-vrf)->redistribute static
S Chassis(su-config-bgp-af-vrf)->exit
S Chassis(su-config-bgp)->exit
S Chassis(rw-config)->
```

MPLS Label Mode

The MPLS label mode determines whether MPLS labels are allocated on a VRF or a prefix basis. By default, a single MPLS label is allocated for the VRF. MPLS label allocation on a per prefix basis can be configured using the **mpls label mode per-prefix** command.

LDP Label Allocation Filtering

By default MPLS LDP allocates labels for all routes except BGP. LDP label allocation filtering allows you to either add the allocation of labels for BGP routes or to allocate labels only for host routes.

Use the **mpls ldp-label-allocate** command in global router or named VRF global configuration mode to configure LDP label allocation filtering.

For a modification of the LDP label allocation configuration to take affect, the MPLS/LDP session must be reset. Use the **no mpls ip** command followed by the **mpls ip** command to reset the MPLS session. Use the **mpls label-protocol-ldp** command to re-enable the LDP session.

Time-To-Live (TTL) Header Propagation

By default the TTL from the IPv4 and IPv6 headers is not propagated to the MPLS label. Propagation of TTL from the IPv4 and IPv6 headers can be configured to only propagate local packets, only propagate forwarded packets or to propagate both local and forwarded packets.

Use the **mpls ip propagate-ttl** command in global router or named VRF global configuration mode to configure TTL IPv4 and IPv6 header propagation to the MPLS label.

For a modification of the MPLS TTL configuration to take affect, the MPLS/LDP session must be reset. Use the **no mpls ip** command followed by the **mpls ip** command to reset the MPLS session. Use the **mpls label-protocol-ldp** command to re-enable the LDP session.

Configuring Layer 3 VPN

Procedure 38-1 describes named VRF Layer 3 VPN configuration. This procedure takes place on the PE router and assumes you have configured the VRF for the L3 VPN (Refer to Chapter 35, **Virtual Routing and Forwarding (VRF) Configuration**) and are in the executive command mode for that VRF. All commands are entered in the VRF to which the L3 VPN belongs.

Procedure 38-1 Layer 3 VPN Named VRF Configuration

Step	Task	Command(s)
1.	In named VRF configuration mode, assign a route distinguisher for the VRF.	**rd** {*asn:num* | *ipv4Addr:num*}
2.	In named VRF configuration mode, configure one or more route targets to identify the routes to import and export for the L3 VPN.	**route-target** {**import** | **export** | **both**} *oui:num*
3.	Optionally, in named VRF configuration mode, configure a VPN identifier, if network features outside of the L3 VPN need to identify the VPN to which a client packet flow belongs.	**vpn id** *oui:vpn-index*
4.	Optionally, when using Native MPLS, change the MPLS label mode to allocate a unique label for each prefix route in the routing table. Defaults to a single label for the VRF.	**mpls label mode** {**per-prefix** | **per-vrf**}

Procedure 38-2 describes global VRF Layer 3 VPN configuration. This procedure takes place in global VRF global configuration mode on the PE router.

Procedure 38-2 Layer 3 VPN using MPLS Global VRF Configuration

Step	Task	Command(s)
1.	When configuring a L3 VPN using Native MPLS, enable Native MPLS for the appropriate address family.	**mpls ip** **mpls ipv6** *transport-address*
2.	When configuring a L3 VPN using Native MPLS, enable LDP as the active label distribution protocol for the appropriate address family for this L3 VPN.	**mpls label-protocol-ldp** {**ipv4** | **ipv6**}
3.	Optionally, configure LDP label allocation filtering.	**mpls ldp-label-allocate** {**bgp-routes** | **host-routes**}
4.	When configuring a L3 VPN using Native MPLS, optionally, configure the LDP label retention mode	**mpls ldp-label-retention-mode** {**liberal** | **conservative**}
5.	Optionally, enable the propagation of TTL from IPv4 and IPv6 headers to the MPLS label for forwarded packets, local packets or both.	**mpls ip propagate-ttl** [**forwarded** | **local**]
6.	Configure a network wide unique LSR ID for the router. Optional for an IPv6 system, required for an IPv4 system	**mpls ldp-lsr-id** *lsr-id*
7.	Configure a loopback interface for this L3 VPN and specify the IPv4 or IPv6 for this interface.	**interface** *loopback-name* **ip address** *ipv4-address* **ipv6 address** *ipv6-address*
8.	Configure any VLANs associated with this router's L3 VPN domain.	**interface** *vlan-name*
9.	When configuring a L3 VPN using L3 tunnels, configure a L3 tunnel to each VPN edge router for each BGP peering session. Configure the IPv4 or IPv6 address associated with the tunnel, as well as, the tunnel source and destination addresses.	**interface** *tunnel-name* **ip** | **ipv6 address** *ip-address* **tunnel source** *ip-address* **tunnel destination** *ip-address*
10.	When configuring a L3 VPN using L3 tunnels, configure a static route so that the remote peer's loopback address prefers the tunneled interface as the next-hop over the VLAN interface the tunnels use.	**ip route** *prefix*/*prefix-length* **interface** *tunnel-name* **ipv6 route** *prefix*/*prefix-length* **interface** *tunnel-name*
11.	Configure OSPFv2 or OSPFv3 to provide reachability for all LSRs within the tunneled domain. Reachability can also be achieved using static routes.	See Chapter 41, **Open Shortest Path First (OSPFv2) Configuration** for OSPF configuration details for an IPv4 L3 VPN. Chapter 42, **Open Shortest Path First Version 3 (OSPFv3) Configuration** for OSPF configuration details for an IPv6 L3 VPN.

Procedure 38-3 describes global VRF Layer 3 VPN configuration. This procedure takes place in global VRF global configuration mode on the PE router.

Procedure 38-3 Global Router BGP Configuration

Step	Task	Command(s)
1.	In BGP configuration mode, Enter the BGP IPv4 or IPv6 L3 VPN address family.	**address-family** {**vpnv4** | **vpnv6**}
2.	In IPv4 or IPv6 BGP L3 VPN address family mode, activate peers for this L3 VPN.	**neighbor** *ip-address* **activate**
3.	In IPv4 or IPv6 BGP L3 VPN address family mode, enable the address family.	**enable**
4.	In BGP configuration mode, enter the IPv4 or IPv6 address mode for the VRF	**address-family** [**ipv4** | **ipv6**] [**unicast** | **multicast** | **both**] [**vrf** *vrf-name*]
5.	If OSPF configuration is present, redistribute L3 VPN routes learned on this VRF to the CE router(s).	Refer to "Using Redistribution" on page 44-6.

To configure SPBV in a L3 VPN over SPBV network, see "Configuring Shortest Path Bridging VLAN" on page 22-6.

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

L3 VPN Using L3 Tunnels or Native MPLS Example Configuration

This section steps you through both a L3 VPN using L3 tunneling and a L3 VPN using Native MPLS. Most steps in the L3 VPN configuration are the same for both L3 VPN types with the following exceptions:

- Native MPLS encapsulation using the **mpls ip** or **mpls ipv6** commands is only enabled for L3 VPN using Native MPLS
- LDP and associated label commands are only configured for L3 VPN using Native MPLS
- L3 tunneling and associated static routes are only configured for L3 VPN using L3 tunneling

In our L3 VPN example we will configure two customers on three PE routers. Our example assumes that:

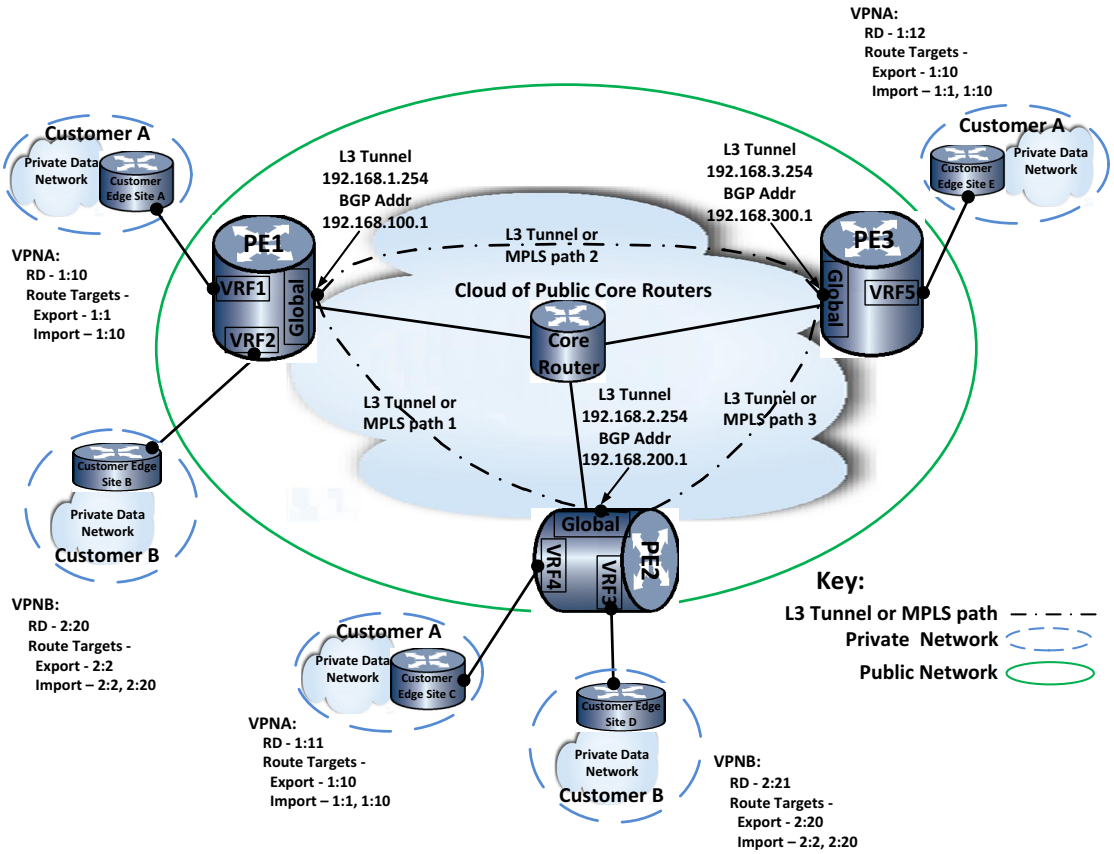
- VRFs are configured on the PE as shown in Figure 38-4 on page 38-17 (Refer to Chapter 35, **Virtual Routing and Forwarding (VRF) Configuration** for VRF configuration details)
- Any interfaces connecting PE routers to directly connected CE routers are configured on the VRFs and interfaces to core routers are configured on the global VRF
- OSPF is configured on each VRF and redistributes the global BGP routes to the CEs using the **redistribute bgp global** command (Refer to Chapter 41, **Open Shortest Path First (OSPFv2) Configuration** for IPv4 and Chapter 42, **Open Shortest Path First Version 3 (OSPFv3) Configuration** for IPv6 OSPF configuration details)
- MPLS label mode will use the default (a single MPLS label is allocated for each VRF)

Note: This example uses the IPv4 address family. For IPv6 configurations:

- Use IPv6 addressing
- Enable MPLS encapsulation using the **mpls ipv6** command, explicitly configuring the IPv6 transport address
- Configure any required static routes using the **ipv6 route** command
- If required, enable LDP using the **mpls label-protocol-ldp ipv6** command
- If LDP is required, explicitly set the LDP LSR ID using the **mpls ldp-lsr-id** command
- Enable the VPNv6 BGP addressing family instead of VPNv4

Figure 38-4 presents the basic configuration requirements for customer A (VPNA) and customer B (VPNB) on the three PE routers (Ignore L3 tunnel addressing when using Native MPLS).

Figure 38-4 Layer 3 VPN using L3 Tunnels or Native MPLS Example



PE Router 1 (PE1)

PE1 connects to both customer A and customer B.

On customer A VRF1 configure:

- RD – 1:10
- Route target – export 1:1 and 1:10

On customer 2 VRF2 configure

- RD – 2:20

- Route target – export 2:2 and import 2:2 and 2:20

In the global VRF (using L3 Tunneling):

- Configure a loopback interface for each L3 tunnel to be used as the tunnel source interface, specifying the global VRF BGP address as secondary
- Configure L3 tunnels 1 (tun.0.1) connecting PE1 and PE2 and L3 tunnel 2 (tun.0.2) connecting PE1 and PE3
- Configure a static route to the loopback interface BGP address to force BGP traffic to use the tunnel instead of the associated VLAN
- Enable the VPNV4 address family in BGP configuration mode (if using IPv6 enable VPNV6)
- Activate VPNV4 neighbors
- Redistribute OSPF in the BGP configuration mode IPv4 address family for both VRF1 and VRF2

In the global VRF (using Native MPLS):

- Enable Native MPLS for the appropriate IP address type
- Enable MPLS LDP for the appropriate IP address type
- Configure a loopback interface specifying the global VRF BGP address as primary
- Enable the VPNV4 address family in BGP configuration mode (if using IPv6 enable VPNV6)
- Activate VPNV4 neighbors
- Redistribute OSPF in the BGP configuration mode IPv4 address family for both VRF1 and VRF2

CLI Input for PE1

VRF1:

```
PE1(su)->router vrf1
PE1(su-vrf1)->configure
PE1(su-vrf1-config)->rd 1:10
PE1(su-vrf1-config)->route-target export 1:1
PE1(su-vrf1-config)->route-target import 1:10
PE1(su-vrf1-config)->exit
PE1(su-vrf1)->exit
```

VRF2

```
PE1(su)->router vrf2
PE1(su-vrf2)->configure
PE1(su-vrf2-config)->rd 2:20
PE1(su-vrf2-config)->route-target both 2:2
PE1(su-vrf2-config)->route-target import 2:20
PE1(su-vrf2-config)->exit
PE1(su-vrf2)->exit
```

Global VRF (L3 Tunnel)

Use the following example input when configuring L3 VPN using L3 tunnels. If you are configuring L3 VPN using Native MPLS see “Global VRF (Native MPLS)” on page 38-19.

```
PE1(su)->configure
PE1(su-config)->interface loopback 1
PE1(su-config-intf-loop.0.1)->ip address 192.168.1.254 255.255.255.255 primary
PE1(su-config-intf-loop.0.1)->ip address 192.168.100.1 255.255.255.255 secondary
PE1(su-config-intf-loop.0.1)->no shutdown
PE1(su-config-intf-loop.0.1)->exit
PE1(su-config)->interface vlan 4001
PE1(su--config-intf-vlan.0.4001)->ip address 10.10.1.1 255.255.255.0 primary
PE1(su--config-intf-vlan.0.4001)->no shutdown
PE1(su--config-intf-vlan.0.4001)->exit
PE1(su-config)->interface tunnel 1
PE1(su-config-tun.0.1)->tunnel mode gre
PE1(su-config-tun.0.1)->tunnel destination 192.168.2.254
PE1(su-config-tun.0.1)->tunnel source 192.168.1.254
PE1(su-config-tun.0.1)->no shutdown
PE1(su-config-tun.0.1)->exit
PE1(su-config)->ip route 192.168.200.1/32 interface tun.0.1 1
PE1(su-config)->interface tunnel 2
PE1(su-config-tun.0.2)->tunnel mode gre
PE1(su-config-tun.0.2)->tunnel destination 192.168.3.254
PE1(su-config-tun.0.2)->tunnel source 192.168.1.254
PE1(su-config-tun.0.2)->no shutdown
PE1(su-config-tun.0.2)->exit
PE1(su-config)->ip route 192.168.300.1/24 interface tun.0.2 1
```

Global VRF (Native MPLS)

Use the following example input when configuring L3 VPN using Native MPLS. If you are configuring L3 VPN using L3 tunnels see “Global VRF (L3 Tunnel)” on page 38-18.

```
PE1(su)->configure
PE1(su-config)->mpls ip
PE1(su-config)->mpls label-protocol-ldp ipv4
PE1(su-config)->interface loopback 1
PE1(su-config-intf-loop.0.1)->ip address 192.168.100.1 255.255.255.255 primary
PE1(su-config-intf-loop.0.1)->no shutdown
PE1(su-config-intf-loop.0.1)->exit
PE1(su-config)->interface vlan 4001
PE1(su--config-intf-vlan.0.4001)->ip address 10.10.1.1 255.255.255.0 primary
PE1(su--config-intf-vlan.0.4001)->no shutdown
PE1(su--config-intf-vlan.0.4001)->exit
```

Global VRF (BGP)

```
PE1(su-config)->router bgp 64520
PE1(su-config-bgp)->bgp router-id 192.168.100.1
PE1(su-config-bgp)->log-up-down
PE1(su-config-bgp)->neighbor 192.168.200.1 remote-as 64520
```

```
PE1(su-config-bgp)->neighbor 192.168.200.1 update-source 192.168.100 1
PE1(su-config-bgp)->neighbor 192.168.300.1 remote-as 64520
PE1(su-config-bgp)->neighbor 192.168.300.1 update-source 192.168.100 1
PE1(su-config-bgp)->address-family vpnv4
PE1(su-config-bgp-af-vpn)->enable
PE1(su-config-bgp-af-vpn)->192.168.200.1 activate
PE1(su-config-bgp-af-vpn)->exit
PE1(su-config-bgp)->address-family ipv4 vrf vrf1
PE1(su-config-bgp-af-vrf)->redistribute ospf 1
PE1(su-config-bgp-af-vrf)->exit
PE1(su-config-bgp)->address-family ipv4 vrf vrf2
PE1(su-config-bgp-af-vrf)->redistribute ospf 1
PE1(su-config-bgp-af-vrf)->exit
PE1(su-config-bgp)->exit
PE1(su-config)->exit
PE1(su)->
```

PE Router 2 (PE2)

PE2 connects to both customer A and customer B.

On customer B VRF3 configure:

- RD – 2:21
- Route target – export 2:20 and import 2:2 and 2:20

On customer A VRF4 configure:

- RD – 1:11
- Route target – export 1:10 and import 1:1 and 1:10

In the global VRF (using L3 Tunneling):

- Configure a loopback interface for each L3 tunnel to be used as the tunnel source interface, specifying the global VRF BGP address as secondary
- Configure L3 tunnels 1 (tun.0.1) connecting PE2 and PE1 and L3 tunnel 2 (tun.0.2) connecting PE2 and PE3
- Configure a static route to the loopback interface BGP address to force BGP traffic to use the tunnel instead of the associated VLAN
- Enable the VPNV4 address family in BGP configuration mode (if using IPv6 enable VPNV6)
- Activate VPNV4 neighbors
- Activate VPNV4 neighbors
- Redistribute OSPF in the BGP configuration mode IPv4 address family for both VRF3 and VRF4

In the global VRF (using Native MPLS):

- Enable Native MPLS for the appropriate IP address type
- Enable MPLS LDP for the appropriate IP address type
- Configure a loopback interface specifying the global VRF BGP address as primary

- Enable the VPNV4 address family in BGP configuration mode (if using IPv6 enable VPNV6)
- Activate VPNV4 neighbors
- Redistribute OSPF in the BGP configuration mode IPv4 address family for both VRF1 and VRF2

CLI Input for PE2

VRF3:

```
PE2(su)->router vrf3
PE2(su-vrf3)->configure
PE2(su-vrf3-config)->rd 2:21
PE2(su-vrf3-config)->route-target both 2:20
PE2(su-vrf3-config)->route-target import 2:2
PE2(su-vrf3-config)->exit
PE2(su-vrf3)->exit
```

VRF4

```
PE2(su)->router vrf4
PE2(su-vrf4)->configure
PE2(su-vrf4-config)->rd 1:11
PE2(su-vrf4-config)->route-target both 1:10
PE2(su-vrf4-config)->route-target import 1:1
PE2(su-vrf4-config)->exit
PE2(su-vrf4)->exit
```

Global VRF (L3 Tunnel)

Use the following example input when configuring L3 VPN using L3 tunnels. If you are configuring L3 VPN using Native MPLS see “Global VRF (Native MPLS)” on page 38-22.

```
PE2(su)->configure
PE2(su-config)->mpls ip (Native MPLS only)
PE2(su-config)->mpls label-protocol-ldp ipv4 (Native MPLS only)
PE2(su-config)->interface loopback 1
PE2(su-config-intf-loop.0.1)->ip address 192.168.2.254 255.255.255.255 primary
PE2(su-config-intf-loop.0.1)->ip address 192.168.200.1 255.255.255.255 secondary
PE2(su-config-intf-loop.0.1)->no shutdown
PE2(su-config-intf-loop.0.1)->exit
PE2(su-config)->interface vlan 4002
PE2(su--config-intf-vlan.0.4002)->ip address 10.10.1.2 255.255.255.0 primary
PE2(su--config-intf-vlan.0.4002)->no shutdown
PE2(su--config-intf-vlan.0.4002)->exit
PE2(su-config)->interface tunnel 1
PE2(su-config-tun.0.1)->tunnel mode gre
PE2(su-config-tun.0.1)->tunnel destination 192.168.1.254
PE2(su-config-tun.0.1)->tunnel source 192.168.2.254
PE2(su-config-tun.0.1)->no shutdown
PE2(su-config-tun.0.1)->exit
```

```
PE2(su-config)->ip route 192.168.100.1/24 interface tun.0.1 1
PE2(su-config)->interface tunnel 3
PE2(su-config-tun.0.2)->tunnel mode gre
PE2(su-config-tun.0.2)->tunnel destination 192.168.3.254
PE2(su-config-tun.0.2)->tunnel source 192.168.2.254
PE2(su-config-tun.0.2)->no shutdown
PE2(su-config-tun.0.2)->exit
PE2(su-config)->ip route 192.168.300.1/24 interface tun.0.3 1
```

Global VRF (Native MPLS)

Use the following example input when configuring L3 VPN using Native MPLS. If you are configuring L3 VPN using L3 tunnels see "Global VRF (L3 Tunnel)" on page 38-21.

```
PE2(su)->configure
PE2(su-config)->mpls ip
PE2(su-config)->mpls label-protocol-ldp ipv4
PE2(su-config)->interface loopback 1
PE2(su-config-intf-loop.0.1)->ip address 192.168.200.1 255.255.255.255 primary
PE2(su-config-intf-loop.0.1)->no shutdown
PE2(su-config-intf-loop.0.1)->exit
PE2(su-config)->interface vlan 4002
PE2(su--config-intf-vlan.0.4002)->ip address 10.10.1.2 255.255.255.0 primary
PE2(su--config-intf-vlan.0.4002)->no shutdown
PE2(su--config-intf-vlan.0.4002)->exit
```

Global VRF (BGP)

```
PE2(su-config)->router bgp 64520
PE2(su-config-bgp)->bgp router-id 192.168.200.1
PE2(su-config-bgp)->log-up-down
PE2(su-config-bgp)->neighbor 192.168.100.1 remote-as 64520
PE2(su-config-bgp)->neighbor 192.168.300.1 remote-as 64520
PE2(su-config-bgp)->address-family vpnv4
PE2(su-config-bgp-af-vpn)->enable
PE2(su-config-bgp-af-vpn)->192.168.100.1 activate
PE2(su-config-bgp-af-vpn)->exit
PE2(su-config-bgp)->address-family ipv4 vrf vrf3
PE2(su-config-bgp-af-vrf)->redistribute ospf 1
PE2(su-config-bgp-af-vrf)->exit
PE2(su-config-bgp)->address-family ipv4 vrf vrf4
PE2(su-config-bgp-af-vrf)->redistribute ospf 1
PE2(su-config-bgp-af-vrf)->exit
PE2(su-config-bgp)->exit
PE2(su-config)->exit
PE2(su)->
```

PE Router 3 (PE3)

PE3 connects to customer A.

On customer A VRF5 configure:

- RD – 1:12
- Route target – export 1:10 and import 1:1 and 1:10

In the global VRF (using L3 Tunneling):

- Configure a loopback interface for each L3 tunnel to be used as the tunnel source interface, specifying the global VRF BGP address as secondary
- Configure L3 tunnels 2 (tun.0.2) connecting PE3 and PE1 and L3 tunnel 3 (tun.0.3) connecting PE3 and PE2
- Configure a static route to the loopback interface BGP address to force BGP traffic to use the tunnel instead of the associated VLAN
- Enable the VPNV4 address family in BGP configuration mode (if using IPv6 enable VPNV6)
- Activate VPNV4 neighbors
- Redistribute OSPF in the BGP configuration mode IPv4 address family for both VRF3 and VRF5

In the global VRF (using Native MPLS):

- Enable Native MPLS for the appropriate IP address type
- Enable MPLS LDP for the appropriate IP address type
- Configure a loopback interface specifying the global VRF BGP address as primary
- Enable the VPNV4 address family in BGP configuration mode (if using IPv6 enable VPNV6)
- Activate VPNV4 neighbors
- Redistribute OSPF in the BGP configuration mode IPv4 address family for both VRF1 and VRF2

CLI Input for PE3

VRF5

```
PE3(su)->router vrf5
PE3(su-vrf5)->configure
PE3(su-vrf5-config)->rd 1:12
PE3(su-vrf5-config)->route-target both 1:10
PE3(su-vrf5-config)->route-target import 1:1
PE3(su-vrf5-config)->exit
PE3(su-vrf5)->exit
```

Global VRF (L3 Tunnel)

Use the following example input when configuring L3 VPN using L3 tunnels. If you are configuring L3 VPN using Native MPLS see “Global VRF (Native MPLS)” on page 38-24.

```
PE3(su)->configure
PE3(su-config)->mpls ip (Native MPLS only)
PE3(su-config)->mpls label-protocol-ldp ipv4 (Native MPLS only)
PE3(su-config)->interface loopback 1
```

```
PE3(su-config-intf-loop.0.1)->ip address 192.168.3.254 255.255.255.255 primary
PE3(su-config-intf-loop.0.1)->ip address 192.168.300.1 255.255.255.255 secondary
PE3(su-config-intf-loop.0.1)->no shutdown
PE3(su-config-intf-loop.0.1)->exit
PE3(su-config)->interface vlan 4002
PE3(su--config-intf-vlan.0.4002)->ip address 10.10.1.3 255.255.255.0 primary
PE3(su--config-intf-vlan.0.4002)->no shutdown
PE3(su--config-intf-vlan.0.4002)->exit
PE3(su-config)->interface tunnel 2
PE3(su-config-tun.0.2)->tunnel mode gre
PE3(su-config-tun.0.2)->tunnel destination 192.168.1.254
PE3(su-config-tun.0.2)->tunnel source 192.168.3.254
PE3(su-config-tun.0.2)->no shutdown
PE3(su-config-tun.0.1)->exit
PE3(su-config)->ip route 192.168.100.1/24 interface tun.0.2 1
PE3(su-config)->interface tunnel 3
PE3(su-config-tun.0.3)->tunnel mode gre
PE3(su-config-tun.0.3)->tunnel destination 192.168.2.254
PE3(su-config-tun.0.3)->tunnel source 192.168.3.254
PE3(su-config-tun.0.3)->no shutdown
PE3(su-config-tun.0.3)->exit
PE3(su-config)->ip route 192.168.200.1/32 interface tun.0.3 1
```

Global VRF (Native MPLS)

Use the following example input when configuring L3 VPN using Native MPLS. If you are configuring L3 VPN using L3 tunnels see “Global VRF (L3 Tunnel)” on page 38-23.

```
PE3(su)->configure
PE3(su-config)->mpls ip
PE3(su-config)->mpls label-protocol-ldp ipv4
PE3(su-config)->interface loopback 1
PE3(su-config-intf-loop.0.1)->ip address 92.168.300.1 255.255.255.255 primary
PE3(su-config-intf-loop.0.1)->no shutdown
PE3(su-config-intf-loop.0.1)->exit
PE3(su-config)->interface vlan 4002
PE3(su--config-intf-vlan.0.4002)->ip address 10.10.1.3 255.255.255.0 primary
PE3(su--config-intf-vlan.0.4002)->no shutdown
PE3(su--config-intf-vlan.0.4002)->exit
```

Global VRF (BGP)

```
PE3(su-config)->router bgp 64520
PE3(su-config-bgp)->bgp router-id 192.168.300.1
PE3(su-config-bgp)->log-up-down
PE3(su-config-bgp)->neighbor 192.168.100.1 remote-as 64520
PE3(su-config-bgp)->neighbor 192.168.200.1 remote-as 64520
PE3(su-config-bgp)->address-family vpnv4
```

```
PE3(su-config-bgp-af-vpn)->enable
PE3(su-config-bgp-af-vpn)->exit
PE3(su-config-bgp)->address-family ipv4 vrf vrf5
PE3(su-config-bgp-af-vrf)->redistribute ospf 1
PE3(su-config-bgp-af-vrf)->exit
PE3(su-config-bgp)->exit
PE3(su-config)->exit
PE3(su)->
```

L3 VPN Over SPBV Example Configuration

This section steps you through a L3 VPN over SPBV configuration example. In our L3 VPN example we will configure two customers on three PE routers. Our example assumes that:

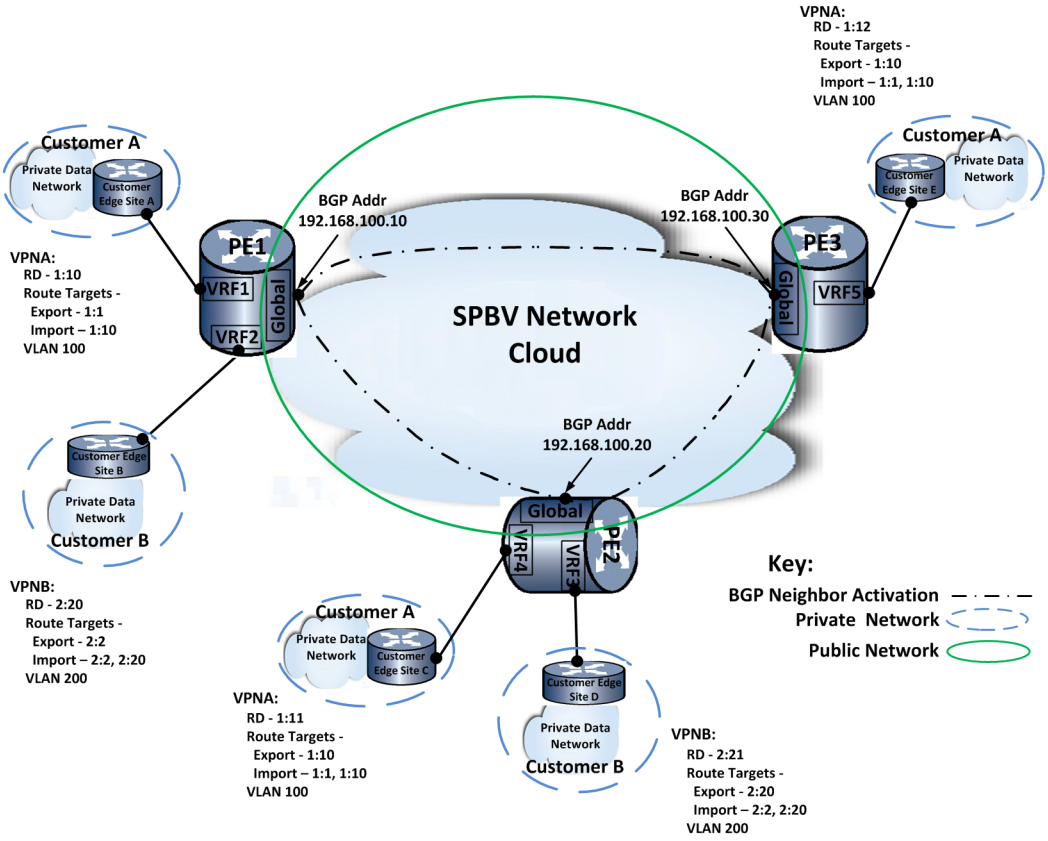
- VRFs are configured on the PE as shown in Figure 38-5 on page 38-26 (Refer to Chapter 35, **Virtual Routing and Forwarding (VRF) Configuration** for VRF configuration details)
- Any interfaces connecting PE routers to directly connected CE routers are configured on the Named VRFs and SPBV boundary interfaces are configured on the global VRF
- SPBV is configured on all PE routers, with all Global router ports enabled for SPBV
- SPBV is configured on all devices in the SPBV network cloud, with all ports enabled for SPBV

Note: This example uses the IPv4 address family. For IPv6 configurations:

- Use IPv6 addressing
- Enable the VPNv6 BGP addressing family instead of VPNv4

Figure 38-5 presents the basic configuration requirements for customer A (VPNA) and customer B (VPNB) on the three PE routers.

Figure 38-5 Layer 3 VPN over SPBV Example



PE Router 1 (PE1)

PE1 connects to both customer A and customer B.

In executive command mode, configure SPBV on the router:

- Set Spanning Tree version to SPT
- Configure the Spanning Tree MST configuration name to spbvRegion1
- Configure the SPVID pool for 2000 - 2200 and set the pool to SID **spvid** (**4095**)
- Map customer VLANs 100 and 200 to SID **spbv** (**4093**)
- For the ECT algorithm, the default value of 1 is used.
- Configure all Global router ports, ge.1.10-20 are enabled for SPBV

On customer A VRF1 configure:

- RD – 1:10
- Route target – export 1:1 and 1:10

On customer 2 VRF2 configure

- RD – 2:20
- Route target – export 2:2 and import 2:2 and 2:20

In the global VRF:

- Configure a loopback interface specifying the global VRF BGP address as primary
- Enable the VPNV4 address family in BGP configuration mode (if using IPv6 enable VPNV6)
- Activate VPNV4 neighbors

CLI Input for PE1

SPBV:

```
PE1(su)->set spantree version spt
PE1(su)->set spantree mstcfgid cfgname spbvRegion1
PE1(su)->set spantree mstmap 2000-2200 sid spvid
PE1(su)->set spantree mstmap 100,200 sid spbv
PE1(su)->set spb port ge.1.10-20 status enable
```

VRF1:

```
PE1(su)->router vrf1
PE1(su-vrf1)->configure
PE1(su-vrf1-config)->rd 1:10
PE1(su-vrf1-config)->route-target export 1:1
PE1(su-vrf1-config)->route-target import 1:10
PE1(su-vrf1-config)->exit
PE1(su-vrf1)->exit
```

VRF2

```
PE1(su)->router vrf2
PE1(su-vrf2)->configure
PE1(su-vrf2-config)->rd 2:20
PE1(su-vrf2-config)->route-target both 2:2
PE1(su-vrf2-config)->route-target import 2:20
PE1(su-vrf2-config)->exit
PE1(su-vrf2)->exit
```

Global VRF

Use the following example input when configuring L3 VPN over SPBV.

```
PE1(su)->configure
PE1(su-config)->interface loopback 1
PE1(su-config-intf-loop.0.1)->ip address 192.168.100.10 255.255.255.255 primary
PE1(su-config-intf-loop.0.1)->no shutdown
PE1(su-config-intf-loop.0.1)->exit
PE1(su-config)->interface vlan 100
PE1(su--config-intf-vlan.0.100)->ip address 192.168.100.110 255.255.255.0 primary
PE1(su--config-intf-vlan.0.100)->no shutdown
PE1(su--config-intf-vlan.0.100)->exit
PE1(su-config)->interface vlan 200
PE1(su--config-intf-vlan.0.200)->ip address 192.168.100.120 255.255.255.0 primary
PE1(su--config-intf-vlan.0.200)->no shutdown
PE1(su--config-intf-vlan.0.200)->exit
```

```
PE1(su-config)->
```

Global VRF (BGP)

```
PE1(su-config)->router bgp 64520
PE1(su-config-bgp)->bgp router-id 192.168.100.10
PE1(su-config-bgp)->log-up-down
PE1(su-config-bgp)->neighbor 192.168.100.20 remote-as 64520
PE1(su-config-bgp)->neighbor 192.168.100.20 update-source 192.168.100.10
PE1(su-config-bgp)->neighbor 192.168.100.30 remote-as 64520
PE1(su-config-bgp)->neighbor 192.168.100.30 update-source 192.168.100.10
PE1(su-config-bgp)->address-family vpnv4
PE1(su-config-bgp-af-vpn)->enable
PE1(su-config-bgp-af-vpn)->192.168.100.20 activate
PE1(su-config-bgp-af-vpn)->192.168.100.30 activate
PE1(su-config-bgp-af-vpn)->exit
PE1(su-config-bgp)->exit
PE1(su-config)->exit
PE1(su)->
```

PE Router 2 (PE2)

PE2 connects to both customer A and customer B.

In executive command mode, configure SPBV on the router:

- Set Spanning Tree version to SPT
- Configure the Spanning Tree MST configuration name to spbvRegion1
- Configure the SPVID pool for 2000 - 2200 and set the pool to SID **spvid** (**4095**)
- Map customer VLANs 100 and 200 to SID **spbv** (**4093**)
- For the ECT algorithm, the default value of 1 is used.
- All Global router ports, ge.1.10-20 are enabled for SPBV

On customer B VRF3 configure:

- RD – 2:21
- Route target – export 2:20 and import 2:2 and 2:20

On customer A VRF4 configure:

- RD – 1:11
- Route target – export 1:10 and import 1:1 and 1:10

In the global VRF (using L3 Tunneling):

- Configure a loopback interface for each L3 tunnel to be used as the tunnel source interface, specifying the global VRF BGP address as secondary
- Configure L3 tunnels 1 (tun.0.1) connecting PE2 and PE1 and L3 tunnel 2 (tun.0.2) connecting PE2 and PE3
- Configure a static route to the loopback interface BGP address to force BGP traffic to use the tunnel instead of the associated VLAN

- Enable the VPNV4 address family in BGP configuration mode (if using IPv6 enable VPNV6)
- Activate VPNV4 neighbors
- Activate VPNV4 neighbors
- Redistribute OSPF in the BGP configuration mode IPv4 address family for both VRF3 and VRF4

In the global VRF (using Native MPLS):

- Enable Native MPLS for the appropriate IP address type
- Enable MPLS LDP for the appropriate IP address type
- Configure a loopback interface specifying the global VRF BGP address as primary
- Enable the VPNV4 address family in BGP configuration mode (if using IPv6 enable VPNV6)
- Activate VPNV4 neighbors
- Redistribute OSPF in the BGP configuration mode IPv4 address family for both VRF1 and VRF2

CLI Input for PE2

SPBV:

```
PE1(su)->set spantree version spt
PE1(su)->set spantree mstcfgid cfgname spbvRegion1
PE1(su)->set spantree mstmap 2000-2200 sid spvid
PE1(su)->set spantree mstmap 100,200 sid spbv
PE1(su)->set spb port ge.1.10-20 status enable
```

VRF3:

```
PE2(su)->router vrf3
PE2(su-vrf3)->configure
PE2(su-vrf3-config)->rd 2:21
PE2(su-vrf3-config)->route-target both 2:20
PE2(su-vrf3-config)->route-target import 2:2
PE2(su-vrf3-config)->exit
PE2(su-vrf3)->exit
```

VRF4

```
PE2(su)->router vrf4
PE2(su-vrf4)->configure
PE2(su-vrf4-config)->rd 1:11
PE2(su-vrf4-config)->route-target both 1:10
PE2(su-vrf4-config)->route-target import 1:1
PE2(su-vrf4-config)->exit
PE2(su-vrf4)->exit
```

Global VRF

Use the following example input when configuring L3 VPN over SPBV.

```
PE1(su)->configure
PE1(su-config)->interface loopback 1
```

```
PE1(su-config-intf-loop.0.1)->ip address 192.168.100.20 255.255.255.255 primary
PE1(su-config-intf-loop.0.1)->no shutdown
PE1(su-config-intf-loop.0.1)->exit
PE1(su-config)->interface vlan 100
PE1(su--config-intf-vlan.0.100)->ip address 192.168.100.112 255.255.255.0 primary
PE1(su--config-intf-vlan.0.100)->no shutdown
PE1(su--config-intf-vlan.0.100)->exit
PE1(su-config)->interface vlan 200
PE1(su--config-intf-vlan.0.200)->ip address 192.168.100.122 255.255.255.0 primary
PE1(su--config-intf-vlan.0.200)->no shutdown
PE1(su--config-intf-vlan.0.200)->exit
```

Global VRF (BGP)

```
PE2(su-config)->router bgp 64520
PE2(su-config-bgp)->bgp router-id 192.168.100.20
PE2(su-config-bgp)->log-up-down
PE2(su-config-bgp)->neighbor 192.168.100.10 remote-as 64520
PE2(su-config-bgp)->neighbor 192.168.100.10 update-source 192.168.100.20
PE2(su-config-bgp)->neighbor 192.168.100.30 remote-as 64520
PE2(su-config-bgp)->neighbor 192.168.100.30 update-source 192.168.100.20
PE2(su-config-bgp)->address-family vpnv4
PE2(su-config-bgp-af-vpn)->enable
PE2(su-config-bgp-af-vpn)->192.168.100.10 activate
PE2(su-config-bgp-af-vpn)->192.168.100.30 activate
PE2(su-config-bgp-af-vpn)->exit
PE2(su-config-bgp)->exit
PE2(su-config)->exit
PE2(su)->
```

PE Router 3 (PE3)

PE3 connects to customer A.

In executive command mode, configure SPBV on the router:

- Set Spanning Tree version to SPT
- Configure the Spanning Tree MST configuration name to spbvRegion1
- Configure the SPVID pool for 2000 - 2200 and set the pool to SID **spvid** (**4095**)
- Map customer VLAN 100 to SID **spbv** (**4093**)
- For the ECT algorithm, there is only a single SPBV region, so the default value of 1 is used.
- All Global router ports, ge.1.10-20 are enabled for SPBV

On customer A VRF5 configure:

- RD – 1:12
- Route target – export 1:10 and import 1:1 and 1:10

In the global VRF (using L3 Tunneling):

- Configure a loopback interface for each L3 tunnel to be used as the tunnel source interface, specifying the global VRF BGP address as secondary
- Configure L3 tunnels 2 (tun.0.2) connecting PE3 and PE1 and L3 tunnel 3 (tun.0.3) connecting PE3 and PE2
- Configure a static route to the loopback interface BGP address to force BGP traffic to use the tunnel instead of the associated VLAN
- Enable the VPNV4 address family in BGP configuration mode (if using IPv6 enable VPNV6)
- Activate VPNV4 neighbors
- Redistribute OSPF in the BGP configuration mode IPv4 address family for both VRF3 and VRF5

In the global VRF (using Native MPLS):

- Enable Native MPLS for the appropriate IP address type
- Enable MPLS LDP for the appropriate IP address type
- Configure a loopback interface specifying the global VRF BGP address as primary
- Enable the VPNV4 address family in BGP configuration mode (if using IPv6 enable VPNV6)
- Activate VPNV4 neighbors
- Redistribute OSPF in the BGP configuration mode IPv4 address family for both VRF1 and VRF2

CLI Input for PE3

SPBV:

```
PE1(su)->set spantree version spt
PE1(su)->set spantree mstcfgid cfgname spbvRegion1
PE1(su)->set spantree mstmap 2000-2200 sid spvid
PE1(su)->set spantree mstmap 100 sid spbv
PE1(su)->set spb port ge.1.10-20 status enable
```

VRF5

```
PE3(su)->router vrf5
PE3(su-vrf5)->configure
PE3(su-vrf5-config)->rd 1:12
PE3(su-vrf5-config)->route-target both 1:10
PE3(su-vrf5-config)->route-target import 1:1
PE3(su-vrf5-config)->exit
PE3(su-vrf5)->exit
```

Global VRF

Use the following example input when configuring L3 VPN over SPBV.

```
PE1(su)->configure
PE1(su-config)->interface loopback 1
PE1(su-config-intf-loop.0.1)->ip address 192.168.100.30 255.255.255.255 primary
PE1(su-config-intf-loop.0.1)->no shutdown
PE1(su-config-intf-loop.0.1)->exit
```

```
PE1(su-config)->interface vlan 100
PE1(su--config-intf-vlan.0.100)->ip address 192.168.100.113 255.255.255.0 primary
PE1(su--config-intf-vlan.0.100)->no shutdown
PE1(su--config-intf-vlan.0.100)->exit
PE1(su-config)->interface vlan 200
PE1(su--config-intf-vlan.0.200)->ip address 192.168.100.123 255.255.255.0 primary
PE1(su--config-intf-vlan.0.200)->no shutdown
PE1(su--config-intf-vlan.0.200)->exit
```

Global VRF (BGP)

```
PE2(su-config)->router bgp 64520
PE2(su-config-bgp)->bgp router-id 192.168.100.30
PE2(su-config-bgp)->log-up-down
PE2(su-config-bgp)->neighbor 192.168.100.10 remote-as 64520
PE2(su-config-bgp)->neighbor 192.168.100.10 update-source 192.168.100.30
PE2(su-config-bgp)->neighbor 192.168.100.20 remote-as 64520
PE2(su-config-bgp)->neighbor 192.168.100.20 update-source 192.168.100.30
PE2(su-config-bgp)->address-family vpnv4
PE2(su-config-bgp-af-vpn)->enable
PE2(su-config-bgp-af-vpn)->192.168.100.10 activate
PE2(su-config-bgp-af-vpn)->192.168.100.30 activate
PE2(su-config-bgp-af-vpn)->exit
PE2(su-config-bgp)->exit
PE2(su-config)->exit
PE2(su)->
```

Terms and Definitions

Table 38-1 lists terms and definitions used in this VRF configuration discussion.

Table 38-1 VRF Configuration Terms and Definitions

Term	Definition
BGP L3 VPN address family	An IPv4 or IPv6 address family used to activate BGP neighbors for this L3 VPN.
Customer Equipment (CE) router	A router on the edge of the customer private network, running an IGP such as OSPF that is directly connected to the public network PE on which L3 VPN is configured.
L3 Tunnel	A Layer 3 tunnel through which L3 VPN packets pass transparently through the public core routers between the PE end-point routers.
Provider (P) core router	A router in the core of the public network running an IGP such as OSPF through which the L3 tunnel connecting PE routers passes.
Provider Edge (PE) router	A router on the edge of the public network connected to the private network customer equipment on which the L3 VPN configuration takes place
Route Distinguisher (RD)	A 64 bit identifier attribute that gets prepended to the user IPv4 or IPv6 address and makes the IP address globally unique across the VPN network and within the BGP routing table.

Table 38-1 VRF Configuration Terms and Definitions (continued)

Term	Definition
Route Target	Determines which L3 VPN routes are inserted into the VRF.
VPN ID	A virtual private network identifier used by non-VPN network resources to identify the VPN to which a client packet flow belongs.
VRF instance	A segregated routing domain for the routed forwarding of packets managed by the global router.
Native MPLS	An encapsulation method that provides the egress router path using an MPLS label.
MPLS LDP	A label distribution protocol capable of determining the egress router path assigned to the MPLS label.
LDP LSR ID	The label distribution protocol Label Switch Router Identifier.

39

Routing Information Protocol (RIP) Configuration

This document describes the RIP feature and its configuration on Extreme Networks S-Series devices.

Using RIP in Your Network

The S-Series device supports Routing Information Protocol (RIP) Version 2. RIP is a distance-vector routing protocol for use in small networks; it is not intended for complex networks. RIP is described in RFC 2453. A router, running RIP broadcasts, updates at set intervals. Each update contains paired values where each pair consists of an IP network address and an integer distance to that network. RIP uses a hop count metric to measure the distance to a destination and is not appropriate for situations where routes need to be chosen based on real-time parameters such as a measured delay, reliability, or load.

The S-Series device implements plain text and MD5 authentication methods for RIP Version 2.

RIP Overview

This section provides an overview of RIP configuration.

Enabling RIP on the device starts the RIP process which then begins populating its routing table and sending and receiving routing updates. Use the **router rip** command in configuration command mode to both enable RIP on the device and enter RIP configuration command mode.

Within RIP configuration command mode:

- Attach one or more networks to the RIP process specifying the IP address of the directly connected network, followed by the wildcard mask for this network. RIP network wildcard masks are reverse networks (use 1's for don't care bits). Use the **network** command to attach one or more networks to this RIP process.
- Optionally change the preference value for using RIP as the routing protocol for this device by changing the RIP administrative distance value using the **distance** command.
- Optionally specify interfaces which will not transmit any RIP update packets using the **passive-interface** command.

- Optionally adjust routing timers associated with:
  - The frequency of routing updates by specifying the interval, in seconds, at which routing updates are sent
  - The expiration of routes by specifying the interval, in seconds, from the point of the last update after which a route times out and is marked as expired
  - The deletion of routes by specifying the interval in seconds from the point of a routes expiration after which a route is deleted from the routing table

  Using the **timers** command. Use the **show ip protocols** command to display RIP timer values.
- Specify route types that can be redistributed in RIP update messages using the **redistribute** command. The S-Series supports redistribution of connected and static routes, optionally specifying the hop count metric for these routes, or specifying OSPF using the process ID to redistribute.

RIP Configuration Example

The following configuration example:

- Enables the RIP process on this device and enters RIP configuration command mode
- Attaches the 10.10.20.0 and 10.10.50.0 networks to this RIP process
- Configures VLANs 10 and 20 as passive-interfaces
- Changes the RIP timers to a 25 second update time, a 150 second expiration interval, and a 100 second flush time:
- Configures the redistribution of OSPF process ID 16546 routes over this RIP process

```
S Chassis(rw-config)->router rip
S Chassis(rw-config-rip)->network 10.10.20.0 0.0.0.255
S Chassis(rw-config-rip)->network 10.10.50.0 0.0.0.255
S Chassis(rw-config-rip)->passive-interface vlan 10
S Chassis(rw-config-rip)->passive interface vlan 20
S Chassis(rw-config-rip)->timers basic 25 150 100
S Chassis(rw-config-rip)->redistribute ospf 16546
S Chassis(rw-config-rip)->exit
S Chassis(rw-config-)->
```

Configuring RIP Authentication

At the interface command level, RIP supports authentication configuration.

The authentication mode applied to the interface can be either clear text or encrypted MD5. Use the **ip rip authentication** mode command to specify the authentication mode for this interface.

Authentication parameters are specified in a key chain. The key chain can be configured for up to 255 keys. A key contains the key authentication string that is sent and received in RIP packets, an accept-lifetime that specifies the period during which an authentication key is valid to be received, and a send-lifetime which specifies the time period during which an authentication key is valid to be sent.

Use the **key chain** command in configuration command mode to enter key chain configuration command mode.

Use the **key** command in key chain configuration command mode to configure a key chain key and enter key configuration command mode.

Use the **key-string** command in key configuration command mode to specify the key string associated with this key.

Use the **accept-lifetime** command in key configuration command mode to specify the time period during which this key can be received for authentication by interface this key chain is associated with.

Use the **send-lifetime** command in key configuration command mode to specify the time period during which this key can be sent by the interface this key chain is associated with.

Use the **ip rip authentication keychain** command in interface configuration command mode to specify the named key chain this interface will use when authenticating RIP packets.

The following example:

- configures key **3** on key chain **md5key**, with a key string of **password**, an accept-lifetime and send-lifetime from the current time to infinite
- Configures VLAN 5 for RIP MD5 authentication
- Applies the md5key key chain to VLAN 5

```
S Chassis(rw-config)->key chain md5key
S Chassis(rw-config-keychain)->key 3
S Chassis(rw-config-keychain-key)->key-string password
S Chassis>Router(config-keychain-key)->accept-lifetime 02:30:00 jul 30 2009
infinite
S Chassis(rw-config-keychain-key)->send-lifetime 02:30:00 jul 30 2009 infinite
S Chassis(rw-config-keychain-key)->show running config
.
.
.
!
 key chain md5key
  key 3
   key-string password
   accept-lifetime 02:30:00 Jul 30 2009 06:28:14 Feb 7 2106
   send-lifetime 02:30:00 Jul 30 2009 06:28:14 Feb 7 2106
   exit
  exit
!
S Chassis(rw-config-keychain-key)->exit
S Chassis(rw-config-keychain)->exit
S Chassis(rw-config)->interface vlan 5
S Chassis(rw-config-intf-vlan.0.5)->ip rip authentication mode md5
S Chassis(rw-config-intf-vlan.0.5)->ip rip authentication keychain md5key
S Chassis(rw-config-intf-vlan.0.5)->exit
S Chassis(rw-config)->
```

Configuring RIP Offset

In interface command mode, an offset can be added to the hop metric of an incoming or outgoing route learned by RIP. Use the **ip rip offset** command, specifying an offset value and whether the offset applies to incoming or outgoing route.

The following example configures VLAN 1 with a RIP offset of 5 for incoming RIP learned routes:

```
S Chassis(rw-config)->interface vlan 1
S Chassis(rw-config-intf-vlan.0.1)->ip rip offset in 1
```

Configuring RIP

This section provides details for the configuration of RIP on the S-Series products.

Table 39-1 lists RIP parameters and their default values.

Table 39-1 Default RIP Parameters

Parameter	Description	Default Value
RIP process	The RIP Router process on this device.	disabled
distance	The administrative distance that specifies the preference for RIP routing over other routing types on this device.	120
update interval	Specifies the interval between routing updates.	30 seconds
expiration interval	Specifies the interval from the point of the last update after which a route times out and is marked to expire.	180 seconds
flush interval	Specifies the interval from the point of a routes expiration after which a route is deleted from the routing table.	120 seconds

Procedure 39-1 describes how to configure RIP.

Procedure 39-1 Configuring RIP

Step	Task	Command(s)
1.	In configuration command mode, enable the RIP process for this device.	**router rip**
2.	In RIP configuration command mode, attach one or more networks to this RIP process.	**network** *ip-address wild-card-bits*
3.	Optionally, in RIP configuration command mode, change the administrative distance for RIP routing on this device.	**distance** *weight*
4.	Optionally, in interface configuration command mode, add an offset to the hop metric of an incoming or outgoing RIP route for this interface.	**ip rip offset** {**in** | **out**} *value*

Procedure 39-1 Configuring RIP (continued)

Step	Task	Command(s)
5.	Optionally, in RIP configuration command mode, change the basic timers associated with RIP: • Update interval • Route expiration interval • Route flush interval	**timers basic** *update-seconds invalid-seconds flush-seconds*
6.	Optionally, in configuration command mode, name a RIP authentication key chain and enter key chain configuration command mode.	**key chain** *name*
7.	Optionally, in key chain configuration command mode, create a RIP authentication key for this key chain and enter authentication key configuration command mode.	**key** *key-id*
8.	Optionally, In authentication key configuration command mode, specify a key-string for this key that will be used by RIP to authenticate sent and received RIP packets.	**key-string** *text*
9.	Optionally, in key configuration command mode, specify a time period during which an authentication key is valid to be received.	**accept-lifetime** *start-time month date year* {**duration** *seconds* | *end-time* | **infinite**}
10.	Optionally, in key configuration command mode, specify a time period during which an authentication key is valid to be sent.	**send-lifetime** *start-time month date year* {**duration** *seconds* | *end-time* | **infinite**}
11.	Optionally, in interface configuration command mode, apply a RIP authentication key chain to an interface.	**ip rip authentication keychain** *name*
12.	Optionally, in interface configuration command mode, set the authentication mode when a key chain is present on this interface.	**ip rip authentication mode** {**text** | **md5**}
13.	Optionally, in RIP configuration command mode, specify an interface that will be prevented from transmitting RIP update packets.	**passive-interface vlan** *vlan-id*
14.	In RIP configuration command mode, specify the non-RIP protocols to be distributed in RIP update messages.	**redistribute** {**connected** | **ospf** *process-id* | **static**} [**metric** *metric-value*]

Terms and Definitions

Table 39-2 lists terms and definitions used in this RIP configuration discussion.

Table 39-2 RIP Configuration Terms and Definitions

Term	Definition
Routing Information Protocol (RIP)	A distance-vector routing protocol for use in small networks that broadcasts route updates at set intervals using a hop metric to determine route preference.
distance	An administrative value that sets the preference for the routing protocols on this device.

Table 39-2 RIP Configuration Terms and Definitions (continued)

Term	Definition
RIP offset	A value that is added to the hop metric of an incoming or outgoing route learned by RIP for the configured interface.
update interval	Sets the interval that determines the frequency of routing updates.
expiration interval	Sets the interval that determines the expiration of a route based upon the point of the last update.
flush interval	Sets the interval for the deletion of an expired route based upon the point of expiration.
key chain	A named chain that holds RIP authentication keys.
key	A key chain member that contains the key string used to authenticate RIP packets, accept-lifetime, and send-lifetime.
key string	A text string that is sent with RIP packets which must agree at both ends of the transmission for authentication to take place.
accept-lifetime	Specifies the time period during which an authentication key on a key chain is valid to be received by this device.
send-lifetime	Specifies the time period during which an authentication key on a key chain is valid to be sent by this device.
passive-interface	An interface configured to not transmit RIP update packets.

40

Routing Information Protocol Next Generation (RIPng) Configuration

This document describes the RIPng feature and its configuration on Extreme Networks S-Series devices.

Using RIPng in Your Network

The S-Series device supports Routing Information Protocol Next Generation (RIPng). RIPng is a distance-vector routing protocol for use in small networks; it is not intended for complex networks. RIPng is an Interior Gateway Protocol (IGP) in that it is used within a single Autonomous System (AS). RIPng is described in RFC 2080. A router, running RIPng broadcasts, updates at set intervals. Each update contains paired values where each pair consists of an IP network address and an integer distance to that network. RIPng uses a hop count metric to measure the distance to a destination and is not appropriate for situations where routes need to be chosen based on real-time parameters such as a measured delay, reliability, or load.

RIPng is conceptually the same as RIPv2 for IPv4. In essence, the IPv4 address is expanded into an IPv6 address. RIPng replaces the IPv4 subnet with an IPv6 prefix length. The next-hop header field is eliminated but the functionality is preserved. The route tag field is preserved. The maximum diameter (metric value) of the network is 15, assuming that a cost of 1 is used for each network; 16 still means the route is unreachable.

RIPng uses fixed metrics to compare alternative routes. The RIPng metric of a network is an integer value range of 1 - 15. Given that the maximum path limit is 15, the metric value is usually set to 1.

RIPng uses a UDP-based protocol and sends and receives datagrams on UDP port 521, which is used for all communications with another router's RIPng process. RIPng messages are either a request for all or a part of the receiving router's route table or a response that contains all or part of a sending router's route table.

Unsolicited response messages containing the complete routing table of the sending router are sent out every 30 seconds by default to every neighboring router. There are two timers associated with routing table entries. The expiration timer specifies when a route has expired. The expiration timer is initialized when a route is first established and each time an update is received for that route. The flush timer specifies when an expired route should be removed from the route table.

Each network has an IPv6 destination address prefix and prefix length associated with it.

Authentication has been removed from RIPng.

RIPng Configuration Overview

This section provides an overview of RIPng configuration.

Enabling RIPng on the device starts the RIPng process which then begins populating its routing table and sending and receiving routing updates. Use the **ipv6 router rip** command in configuration command mode to both enable RIPng on the device and enter RIPng configuration command mode.

Within RIPng configuration command mode:

- Optionally change the preference value for using RIPng as the routing protocol for this device by changing the RIPng administrative distance value using the **distance** command.
- Optionally assign standard ACLs to a distribution list to filter networks received and to suppress networks from being advertised in RIPng updates. Use the **distribute-list** command to assign an ACL to the list, specifying an IPv6 routing configured VLAN and whether the affected routing updates are incoming or outgoing.
- Optionally specify interfaces which will not transmit any RIPng update packets using the **passive-interface** command.
- Specify route types that can be redistributed in RIPng update messages using the **redistribute** command. The S-Series supports redistribution of BGP, connected, OSPF (specifying the process ID), and static routes, optionally specifying the hop count metric for these routes or a route map.
- Optionally, using the **timers** command, adjust routing timers associated with:
  - The frequency of routing updates by specifying the interval, in seconds, at which routing updates are sent
  - The expiration of routes by specifying the interval, in seconds, from the point of the last update after which a route times out and is marked as expired
  - The deletion of routes by specifying the interval in seconds from the point of a routes expiration after which a route is deleted from the routing table

Because IPv6 addressing prefix is ambiguous concerning network addressing compared to IPv4, you do not directly specify a network destination address. Instead, you enable RIPng on the interface using the **ipv6 rip enable** command and the interface address is automatically set as the network destination address.

An offset can be added or removed to the hop metric for all incoming or outgoing RIPng routes for a given interface. Adding an offset is used for the purpose of making an interface a backup. Use the **ipv6 rip offset** command in interface configuration mode to add or remove an offset for either incoming or outgoing RIPng routes.

RIPng Configuration Example

The following configuration example:

- Enables the RIPng process on this device and enters RIPng configuration command mode
- Configures VLANs **10** as a passive-interface
- Changes the RIPng timers to a **25** second update time, a **150** second expiration interval, and a **100** second flush time:
- Configures the redistribution of OSPF process ID **16546** routes over this RIPng process

- Configures IPv6 access-list **ipv6list1** with a rule to deny route **2001:0:0:0:21f:45ff:fe3d:21be/64** and applies the ACL to the distribution-list for outgoing packets on VLAN **20**
- Enables RIPng on VLANs **10** and **20**

```
S Chassis(rw-config)->ipv6 router rip
S Chassis(rw-config-ripng)->passive-interface vlan 10
S Chassis(rw-config-ripng)->timers basic 25 150 100
S Chassis(rw-config-ripng)->redistribute ospf 16546
S Chassis(rw-config-ripng)->exit
S Chassis(rw-config)->ipv6 access-list standard ipv6list1
S Chassis(rw-cfg-ipv6-std-acl)->deny 2001:0:0:0:21f:45ff:fe3d:21be/64
S Chassis(rw-cfg-ipv6-std-acl)->exit
S Chassis(rw-config)->ipv6 router rip
S Chassis(rw-config-ripng)->distribute-list ipv6list1 out vlan 20
S Chassis(rw-config-ripng)->exit
S Chassis(rw-config)->interface vlan 10
S Chassis(rw-config-intf-vlan.0.10)->ipv6 rip enable
S Chassis(rw-config-intf-vlan.0.10)->exit
S Chassis(rw-config-)->interface vlan 20
S Chassis(rw-config-intf-vlan.0.20)->ipv6 rip enable
S Chassis(su-config-intf-vlan.0.20)->exit
S Chassis(rw-config)->
```

Configuring RIPng

This section provides details for the configuration of RIPng on the S-Series products.

Table 40-1 lists RIPng parameters and their default values.

Table 40-1 Default RIPng Parameters

Parameter	Description	Default Value
RIPng process	The RIPng Router process on this device.	disabled
distance	The administrative distance that specifies the preference for RIPng routing over other routing types on this device.	120
update interval	Specifies the interval between routing updates.	30 seconds
expiration interval	Specifies the interval from the point of the last update after which a route times out and is marked to expire.	180 seconds
flush interval	Specifies the interval from the point of a routes expiration after which a route is deleted from the routing table.	120 seconds

Procedure 40-1 describes how to configure RIPng.

Procedure 40-1 Configuring RIPng

Step	Task	Command(s)
1.	In configuration command mode, enable the RIPng process for this device.	**ipv6 router rip**
2.	Optionally, in RIPng configuration command mode, change the administrative distance for RIPng routing on this device.	**distance** *weight*
3.	Optionally, in RIPng configuration command mode, change the basic timers associated with RIPng: • Update interval • Route expiration interval • Route flush interval	**timers basic** *update-seconds expiration-seconds flush-seconds*
4.	Optionally, in RIPng configuration command mode, specify an interface that will be prevented from transmitting RIPng update packets.	**passive-interface vlan** *vlan-id*
5.	Optionally, in RIPng configuration command mode, specify a standard IPv6 ACL to be added to the distribute-list to filter networks received and to suppress networks from being advertised in RIPng updates.	**distribute-list** *access-list-name* {**in vlan** *vlan-id* | **out vlan** *vlan-id*}
6.	In RIPng configuration command mode, specify the non-RIPng protocols to be distributed in RIPng update messages.	**redistribute** {**bgp** | **connected** | **ospf** *process-id* | **static**} [**metric** *metric-value*] [**route-map** *route-map*]
7.	In interface configuration mode, enable RIPng on all interfaces that will use the protocol.	**ipv6 rip enable**
8.	Optionally, in interface configuration command mode, add an offset to the hop metric of an incoming or outgoing RIPng route for this interface.	**ipv6 rip offset** {**in** | **out**} *value*

Terms and Definitions

Table 40-2 lists terms and definitions used in this RIPng configuration discussion.

Table 40-2 RIPng Configuration Terms and Definitions

Term	Definition
Routing Information Protocol Next Generation (RIPng)	A distance-vector routing protocol for use in small IPv6 networks that broadcast route updates at set intervals using a hop metric to determine route preference.
distance	An administrative value that sets the preference for the routing protocols on this device.
RIPng offset	A value that is added to the hop metric of an incoming or outgoing route learned by RIPng for the configured interface.
update interval	Sets the interval that determines the frequency of routing updates.

Table 40-2 RIPng Configuration Terms and Definitions (continued)

Term	Definition
expiration interval	Sets the interval that determines the expiration of a route based upon the point of the last update.
flush interval	Sets the interval for the deletion of an expired route based upon the point of expiration.
passive-interface	An interface configured to not transmit RIPng update packets.

41

Open Shortest Path First (OSPFv2) Configuration

This chapter provides the following information about configuring and monitoring OSPFv2 on Extreme Networks S-Series devices:

Using the OSPF Protocol in Your Network

The Open Shortest Path First (OSPF) Link-state routing protocol is considered a TCP/IP internet routing Interior Gateway Protocol (IGP). OSPF distributes routing information between routers belonging to a single Autonomous System (AS). The OSPF protocol is based on link-state or SPF technology. The advantages associated with a link-state routing protocol are:

- Rapid convergence
- Reduced routing updates traffic over traditional distance-vector protocols

This OSPF implementation supports RFC 2328 *OSPF Version 2*.

The OSPF protocol is designed expressly for the TCP/IP internet environment. It provides for the authentication of routing updates, and utilizes IP multicast when sending and receiving the updates.

OSPF routes IP packets based solely on the destination IP address found in the IP packet header. IP packets are not encapsulated in any further protocol headers as they transit the Autonomous System. OSPF is a dynamic routing protocol in that it quickly detects topological changes in the AS, such as router interface failures, and calculates new loop-free routes after a period of convergence. This period of convergence is short and involves a minimum of routing traffic. In a link-state routing protocol, each router maintains a database describing the AS's topology. This database is referred to as the link-state database. Each participating router has an identical database. Each individual piece of this database is a particular router's local state made up of such information as the router's usable interfaces and reachable neighbors. The router distributes its local state throughout the AS by flooding.

Each network that has at least two attached routers has a designated router. The designated router generates an LSA for the network and has other special responsibilities in the running of the protocol, enabling a reduction in the number of adjacencies required on a network. This in turn reduces the amount of routing protocol traffic and the size of the link-state database.

All routers run the exact same algorithm, in parallel. From the link-state database, each router constructs a tree of shortest paths with itself as root. This shortest-path tree provides the route to each destination in the AS. Externally derived routing information appears on the tree as leaves. When several equal-cost routes to a destination exist, traffic is distributed equally among them. The cost of a route is described by a single dimensionless metric.

OSPF allows sets of networks to be grouped together. Such a grouping is called an area. The topology of an area is hidden from the rest of the AS. This information hiding enables a significant reduction in routing traffic. Also, routing within the area is determined only by the area's own topology, lending the area protection against bad routing data. An area is a generalization of an IP subnetted network. OSPF enables the flexible configuration of IP subnets. Each route distributed by OSPF has a destination and mask. Two different subnets of the same IP network number may have different masks providing a different range of addresses for that subnet. This is commonly referred to as Variable Length Subnet Masking (VLSM). A packet is routed to the longest or most specific match. Host routes are considered to be subnets whose masks are "all ones" (0xffffffff).

All OSPF protocol exchanges are authenticated. This means that only trusted routers can participate in the AS's routing. The S-Series platform supports either simple or MD5 authentication schemes. Separate authentication schemes can be configured for each IP subnet.

Route redistribution is supported for RIP, connected, BGP, and static routes.

The Bidirectional Forwarding Detection (BFD) protocol providing sub-second failure detection on OSPF forwarding interfaces is enabled by default on all OSPF interfaces.

An OSPF Customer Edge (CE) router can be configured as a peer to a Provider Edge (PE) router by enabling the PE-CE protocol on the PE-CE associated routers.

Implementing OSPF

To implement OSPF in your network:

- Map out the AS including routers, network subnets, and the areas to which they belong
- Configure each routing interface on each router with an IP address and mask
- Create an OSPF routing instance for this AS
- Configure the network addresses, masks, and areas for each router in the AS
- Configure each router with a router ID
- Optionally determine which router will be the designated router and backup and configure OSPF priority values accordingly
- Optionally configure OSPF timers
- Optionally, configure the protocols and route types that will be redistributed over this AS
- Optionally configure interface cost
- Optionally modify the administrative distance for OSPF routes
- Optionally configure either simple or MD5 authentication per interface
- Optionally configure areas including virtual-links, stub, and NSSA
- Optionally enable graceful-restart
- Optionally enable the BFD protocol on all OSPF interfaces
- Optionally enable the PE-CE protocol on the router, with PE-CE enabled
  - Optionally, configure a domain tag for this router
  - Optionally, configure a primary or secondary domain ID for this router

– Optionally redistribute BGP discovered routes over OSPF

OSPF Overview

OSPF is enabled by creating an OSPF instance. Once an instance is created, the router's OSPF settings are configured with respect to the Instance ID and IP interfaces. By default, OSPF is disabled on the S-Series device. Be aware that unspecified parameters use their default values, and any parameters specified at the interface level will override the values specified at the area level.

Configuring Basic OSPF Parameters

Basic OSPF configuration consists of:

- Entering interface configuration mode for the routing interfaces for this device
- Configuring each routing interface with an IP address and mask
- Enabling the interface
- Creating an OSPF routing instance
- Configuring the network address, mask, and area for this routing instance

Configuring an IP Address

An IP address must be associated with any interface that will route traffic on the router. In interface configuration mode, configure the IP address for each routing interface using the **ip address** command specifying the IP address and mask. For example, IP address 10.10.10.1 would be specified as 10.10.10.1 255.255.255.0. Enable the interface using the **no shutdown** command.

Configuring a Routing Instance

OSPF routing configuration takes place within a routing instance. Configure a routing instance using the **router ospf** command in global configuration command mode. Executing this command places you in the OSPF router configuration command mode for the specified OSPF router instance.

Configuring Networks

A network is made up of a number of IP routers that belong to the same IP network, subnet, or supernet as determined by a device's combined IP address and mask. An edge connecting a router to a network indicates that the router has an interface on the network. Networks can be either transit or stub networks. Transit networks are those capable of carrying data traffic that is neither locally originated nor locally destined. A stub network has only incoming edges.

Use the **network** command in the OSPF router configuration command mode to configure networks and associated areas for this router. See section "Configuring OSPF Areas" on page 41-9 for information on OSPF areas and their configuration.

Note: OSPF network wildcard masks are reverse networks. This means that wherever there is a 1 in a regular netmask, use a 0 in a wildcard mask. For example, if the network mask is 255.255.255.0 (/24), specify a wildcard mask of **0.0.0.255**.

Basic OSPF Topology

Figure 41-1 provides an overview of a basic OSPF topology. This topology displays two areas: a backbone area which must exist in any OSPF topology and a directly connected area 1. See "Configuring OSPF Areas" on page 41-9 for a full discussion of OSPF area configuration. This

basic configuration requires the configuration of three interfaces and associated IP addresses, three networks, and two routers on a single OSPF router instance.

Example

The following example configures the basic OSPF topology as displayed in Figure 41-1 on page 41-5:

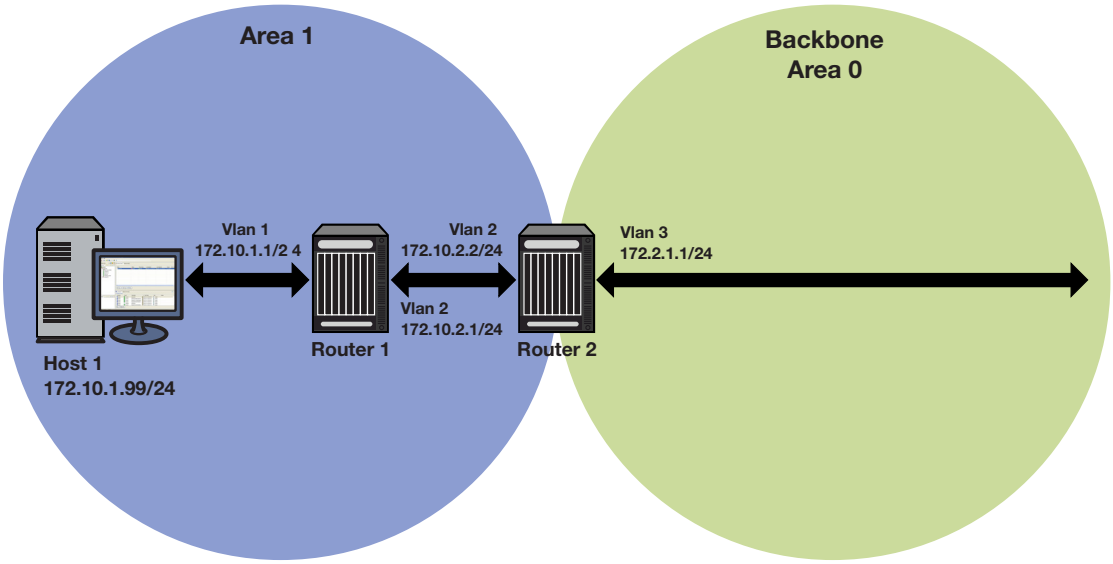
Router 1 CLI Input

```
Router 1(rw)->configure
Router 1(rw-config)->interface vlan 1
Router 1(rw-config-intf-vlan.0.1)->ip address 172.10.1.1 255.255.255.0
Router 1(rw-config-intf-vlan.0.1)->exit
Router 1(rw-config)->interface vlan 2
Router 1(rw-config-intf-vlan.0.2)->ip address 172.10.2.1 255.255.255.0
Router 1(rw-config-intf-vlan.0.2)->exit
Router 1(rw-config)->router ospf 1
Router 1(rw-config-ospf-1)->network 172.10.1.0 0.0.0.255 area 1
Router 1(rw-config-ospf-1)->network 172.10.2.0 0.0.0.255 area 1
Router 1(rw-config-ospf-1)->exit
Router 1(rw-config)->
```

Router 2 CLI Input

```
Router 2(rw)->configure
Router 2(rw-config)->interface vlan 2
Router 2(rw-config-intf-vlan.0.1)->ip address 172.10.2.2 255.255.255.0
Router 2(rw-config-intf-vlan.0.1)->exit
Router 2(rw-config)->interface vlan 3
Router 2(rw-config-intf-vlan.0.2)->ip address 172.2.1.1 255.255.255.0
Router 2(rw-config-intf-vlan.0.2)->exit
Router 2(rw-config)->router ospf 1
Router 2(rw-config-ospf-1)->network 172.10.2.0 0.0.0.255 area 1
Router 2(rw-config-ospf-1)->network 172.2.1.0 0.0.0.255 area 0
Router 2(rw-config-ospf-1)->exit
Router 2(rw-config)->
```

Figure 41-1 Basic OSPF Topology



Configuring the Router ID

OSPF initially assigns all routers a router ID based on the highest loopback IP address of the interfaces configured for IP routing. If there is no loopback interface configured then it will be the highest VLAN IP address configured. This unique value, which is included in the hello packet transmitted in Link State Advertisements (LSA), identifies one router to another and helps establish adjacencies among OSPF routers. When you specify an interface as the router ID, this value supersedes the default ID.

Example

The following example configures the router ID topology as displayed in Figure 41-2 on page 41-6:

Router 1

```
Router 1(rw)->configure
Router 1(rw-config)->interface loopback 1
Router 1(su-config-intf-loop.0.1)->ip address 1.1.1.1 255.255.255.255
Router 1(rw-config-intf-loop.0.1)->exit
Router 1(rw-config)->router ospf 1
Router 1(rw-config-ospf-1)->network 10.1.2.2 0.0.0.255 area 1
Router 1(rw-config-ospf-1)->router-id 1.1.1.1
Router 1(rw-config-ospf-1)->exit
Router 1(rw-config)->
```

Router 2

```
Router 2(rw)->configure
Router 2(rw-config)->interface loopback 1
Router 2(su-config-intf-loop.0.1)->ip address 2.2.2.2 255.255.255.255
Router 2(rw-config-intf-loop.0.1)->exit
Router 2(rw-config)->router ospf 1
Router 2(rw-config-ospf-1)->network 10.1.2.1 0.0.0.255 area 1
```

```
Router 2(rw-config-ospf-1)->network 10.2.3.1 0.0.0.255 area 0
Router 2(rw-config-ospf-1)->router-id 2.2.2.2
Router 2(rw-config-ospf-1)->exit
Router 2(rw-config)->
```

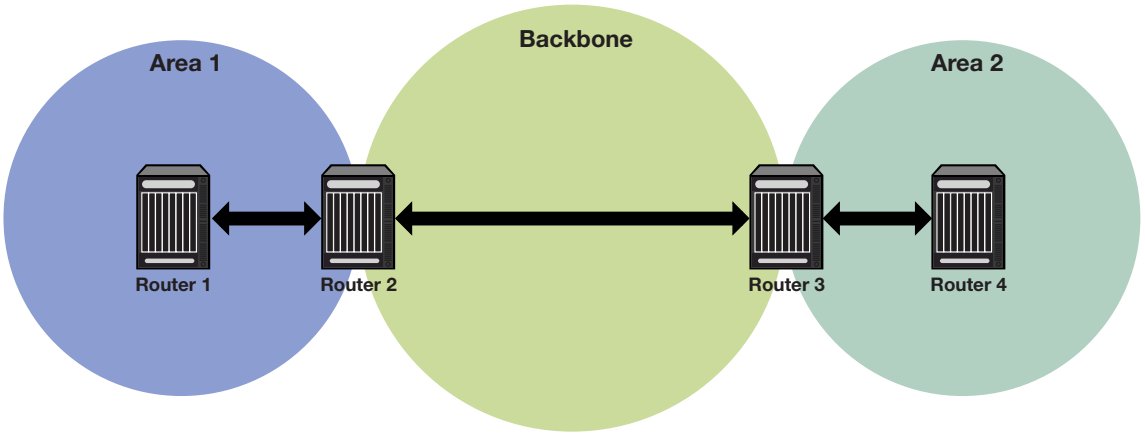
Router 3

```
Router 3(rw)->configure
Router 3(rw-config)->interface loopback 1
Router 3(su-config-intf-loop.0.1)->ip address 3.3.3.3 255.255.255.255
Router 3(rw-config-intf-vlan.0.1)->exit
Router 3(rw-config)->router ospf 1
Router 3(rw-config-ospf-1)->network 10.3.4.1 0.0.0.255 area 2
Router 3(rw-config-ospf-1)->network 10.2.3.2 0.0.0.255 area 0
Router 3(rw-config-ospf-1)->router-id 3.3.3.3
Router 3(rw-config-ospf-1)->exit
Router 3(rw-config)->
```

Router 4

```
Router 4(rw)->configure
Router 4(rw-config)->interface loopback 1
Router 4(su-config-intf-loop.0.1)->ip address 4.4.4.4 255.255.255.255
Router 4(rw-config-intf-vlan.0.1)->exit
Router 4(rw-config)->router ospf 1
Router 4(rw-config-ospf-1)->network 10.3.4.2 0.0.0.255 area 2
Router 4(rw-config-ospf-1)->router-id 4.4.4.4
Router 4(rw-config-ospf-1)->exit
Router 4(rw-config)->
```

Figure 41-2 OSPF Router ID Topology



Configuring the Designated Router

In the process of implementing OSPF, a large number of multi-access links to routers across the network may cause too many adjacencies to form. To avoid this problem, a Designated Router (DR) is elected per multi-access network to build adjacencies to all other routers on that network.

A Backup Designated Router (BDR) is also elected in case the Designated Router (DR) fails, in which case the BDR will become the DR.

Note: A DR is required only for multi-access networks. Point-to-Point links do not need a DR because only a single adjacency is required.

To elect a DR from a host of candidates on the network, each router multicasts a hello packet and examines the priority of hello packets received from other routers. The router with the highest priority is elected the DR, and the router with the next highest priority is elected the BDR. Any router with a priority of 0 will opt out of the DR election process. See the "Configuring Router Priority" on page 41-7 for details on configuring router priority. If DR candidates all share non-zero priorities, OSPF applies the router ID as a tie-breaker where the highest ID is chosen DR and the next highest ID is chosen BDR.

Configuring Router Priority

When two routers attached to a network both attempt to become the designated router, the one with the highest router priority takes precedence. A router whose router priority is set to 0 is ineligible to become the designated router on the attached network. Router priority is specified per router interface and is advertised in hello packets sent out by the interface.

Use the **ip ospf priority** command in interface configuration command mode to specify the router priority that will be specified for LSAs going out this interface. See "Configuring the Designated Router" on page 41-6 for a router priority configuration example.

Figure 41-3 on page 41-8 displays a designated router topology example. The example will configure the four displayed routers with the following priorities:

- Router 1 = 25
- Router 2 = 10
- Router 3 = 30
- Router 4 = 0

Router 4 will not take part in the election process at all. Router 3 has the highest priority and therefore will be elected DR. Router 1 has the second highest priority and will be elected BDR.

Example

The following example provides the input required to configure the designated router topology as displayed in Figure 41-3 on page 41-8:

Router 1

```
Router 1(rw)->configure
Router 1(rw-config)->interface vlan 1
Router 1(rw-config-intf-vlan.0.1)->ip ospf priority 25
Router 1(rw-config-intf-vlan.0.1)->exit
Router 1(rw-config)->
```

Router 2

```
Router 2(rw)->configure
Router 2(rw-config)->interface vlan 1
Router 2(rw-config-intf-vlan.0.1)->ip ospf priority 10
Router 2(rw-config-intf-vlan.0.1)->exit
Router 2(rw-config)->
```

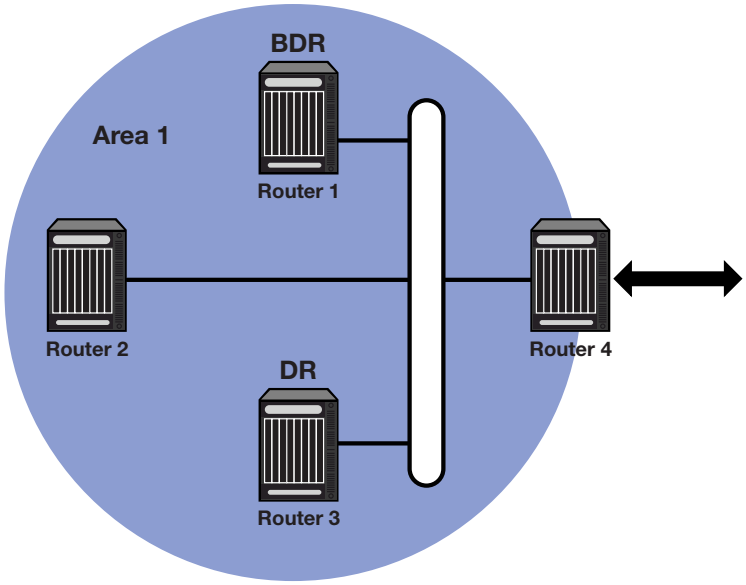
Router 3

```
Router 3(rw)->configure
Router 3(rw-config)->interface vlan 1
Router 3(rw-config-intf-vlan.0.1)->ip ospf priority 30
Router 3(rw-config-intf-vlan.0.1)->exit
Router 3(rw-config)->
```

Router 4

```
Router 4(rw)->configure
Router 4(rw-config)->interface vlan 1
Router 4(rw-config-intf-vlan.0.1)->ip ospf priority 0
Router 4(rw-config-intf-vlan.0.1)->exit
Router 4(rw-config)->
```

Figure 41-3 OSPF Designated Router Topology



Configuring the Administrative Distance for OSPF Routes

If several routes coming from different protocols are presented to the Route Table Manager (RTM), the protocol with the lowest administrative distance will be chosen for route installation. The S-Series platform supports connected, static, OSPF, and RIP routes.

The table below displays the default distance for these routing protocols.

Route Source	Default Distance
Connected	0
Static	1
BGP	20 - Routes external to the AS 200 - Routes internal to the AS

Route Source	Default Distance
OSPF	110
RIP	120

Use the **distance ospf** command in OSPF router configuration command mode to change the administrative distance assigned to the OSPF protocol. This command provides for the configuration of separate values for OSPF external and intra-area routes.

Configuring OSPF Areas

OSPF allows collections of contiguous networks and hosts to be grouped together. Such a group, together with the routers having interfaces to any one of the included networks, is called an area. Each area runs a separate copy of the basic link-state routing algorithm. This means that each area has its own link-state database.

The topology of an area is invisible from the outside of the area, and routers internal to a given area know nothing of the detailed topology external to the area. This isolation of area detail enables the protocol to effect a marked reduction in routing traffic as compared to treating the entire Autonomous System as a single link-state domain. A router has a separate link-state database for each area it is connected to. Routers connected to multiple areas are called Area Border Routers (ABR). Two routers belonging to the same area have, for that area, identical area link-state databases.

An autonomous system can have one or more areas. A multiple area AS must define one of the areas as the backbone with an area ID of **0**. Area IDs are assigned during network configuration using the **network** command (see "Configuring Networks" on page 41-3). All non-backbone areas in a multiple area AS must either be contiguous to the backbone or connected using a virtual-link. The backbone is responsible for distributing routing information between non-backbone areas. The backbone must be contiguous. However, it need not be physically contiguous; backbone connectivity can be established and maintained through the configuration of virtual links.

Virtual links can be configured between any two backbone routers that have an interface to a common non-backbone area. Such virtual links belong to the backbone. The protocol treats two routers joined by a virtual link as if they were connected by an unnumbered point-to-point backbone network.

See RFC 2328 *OSPF Version 2* for further details on inter-area connectivity.

An Area ID can be any value from 0 - 4294967295, but is converted into the 32-bit dotted-quad format (area 50 would be displayed as 0.0.0.50; area 3546 would be displayed as 0.0.13.218)

Configuring Area Range

An area range is a form of address summarization that defines a range of addresses to be used by the backbone ABRs when they communicate routes to other areas. Area range is a critical tool that pares the route tables and update traffic, as well as reduces network recalculation by the Dijkstra algorithm. Area range configuration summarizes by aggregating an areas' internal networks to advertise a single network. Backbone routers see only one update, representing an entire range of subnets. Area ranges can be configured for purposes of network advertisement as well as summarization of subnets that should not be advertised.

Use the **area range** command in OSPF configuration command mode to configure an area network summarization.

Example

The following example provides the input required to configure summarization of the three area topology as displayed in Figure 41-4 on page 41-11:

Area 1

```
ABR1(rw)->configure
ABR1(rw-config)->router ospf 1
ABR1(rw-config-ospf-1)->area 1 range 10.2.0.0 255.255.0.0
ABR1(rw-config-ospf-1)->exit
ABR1(rw-config)->
```

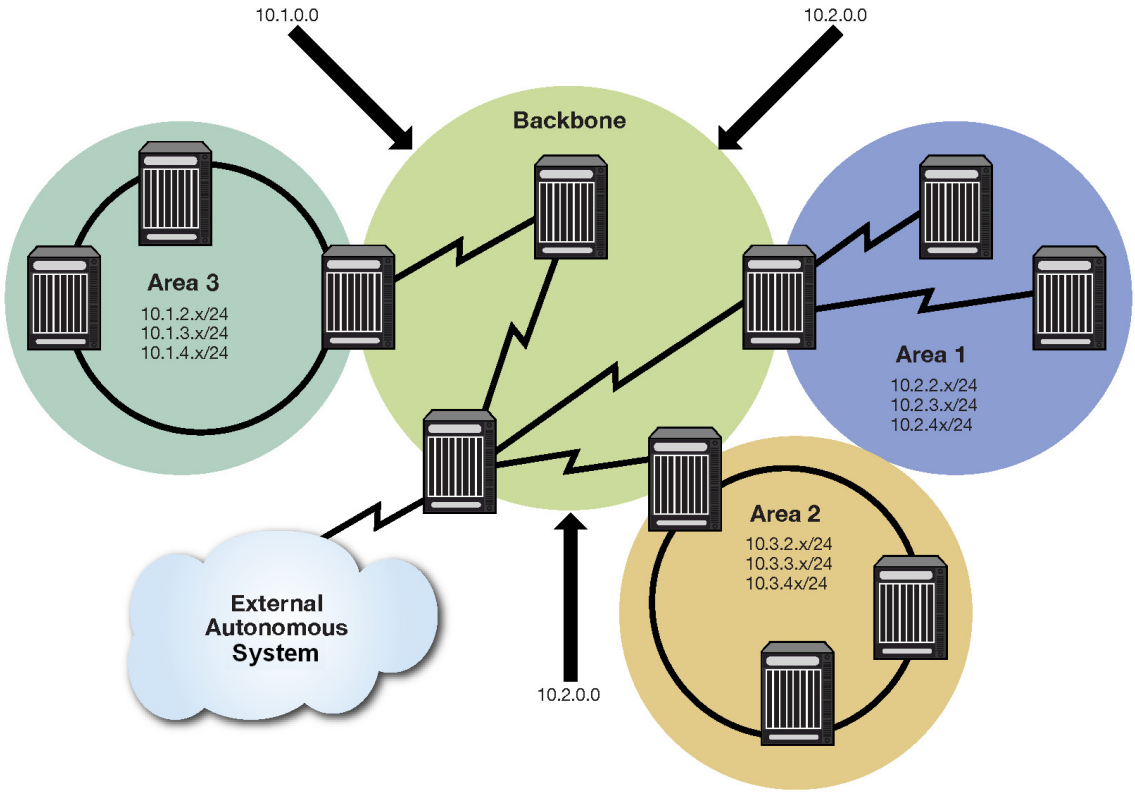
Area 2

```
ABR2(rw)->configure
ABR2(rw-config)->router ospf 1
ABR2(rw-config-ospf-1)->area 2 range 10.3.0.0 255.255.0.0
ABR2(rw-config-ospf-1)->area 2 range 10.3.2.0 255.255.255.0 not-advertised
ABR2(rw-config-ospf-1)->exit
ABR2(rw-config)->
```

Area 3

```
ABR3(rw)->configure
ABR3(rw-config)->router ospf 1
ABR3(rw-config-ospf-1)->area 3 range 10.1.0.0 255.255.0.0
ABR3(rw-config-ospf-1)->exit
ABR3(rw-config)->
```

Figure 41-4 OSPF Summarization Topology



Configuring a Stub Area

A stub area is a non-transit area. In other words, an area that does not originate or propagate external routes. AS-external-LSAs are not flooded into the stub area; routing to AS external networks is based on a single per-area default route. This reduces the link-state-database size and memory requirements for routers within stub areas.

Handy for reducing routing table size, a stub area is a “dead-end” in which there is no other way to enter or exit except through an Area Border Router (ABR). No ASE (Autonomous System External) or NSSA routes are permitted in a stub area. Each router in a stub area must specify that they are members of the stub area. When specifying that the ABR is a member of the stub area, the ABR will inject a default route into the area.

Routing to external designations from stub areas is based on a default route injected by a stub area’s ABR. A default route is automatically created by the stub area’s ABR. This default route is injected into the stub area to enable other stub routers within the stub area to reach any external routes that are no longer inserted into the stub area.

A stub area can be configured such that the ABR is prevented from sending type 3 summary LSAs into the stub area using the **no-summary** option. In this case, all destinations outside of the stub area are represented by means of a default route.

There are a couple of restrictions on the use of stub areas. Virtual-links cannot be configured through stub areas, and AS boundary routers cannot be placed internal to stub areas.

Use the **area stub** command in OSPF router configuration command mode to configure an area as a stub.

Stub Area Default Route Cost

A cost value can be set for the default route that is sent into a stub area by an ABR. Configuration of the stub area default route cost is restricted to the ABR attached to this stub area.

Use the **area default-cost** command in OSPF router configuration command mode on the ABR attached to this stub area to configure the stub area default route cost.

Example

Every router in Areas 1 and 2 are configured for a stub area (Routers 1, 2, and 3 for Area 1 and Routers 5, 6, 7, and 8 for Area 2). Additionally, ABR routers 3, 5, and 6 are also configured with a default-cost to be assigned to the stub area. Router 5 has a lower metric cost when compared to Router 6, so Router 5 will be the preferred router for packets to access the area, with Router 6 employed as a backup in case Router 5 fails. The following example provides the input required to configure the stub topology as displayed in Figure 41-5 on page 41-13:

Router 1

```
Router1(rw-config)->router ospf 1
Router1(rw-config-ospf-1)->area 1 stub
```

Router 2

```
Router2(rw-config)->router ospf 1
Router2(rw-config-ospf-1)->area 1 stub
```

Router 3

```
Router3(rw-config)->router ospf 1
Router3(rw-config-ospf-1)->area 1 stub
Router3(rw-config-ospf-1)->area 1 default-cost 15
```

Router 5

```
Router5(rw-config)->router ospf 1
Router5(rw-config-ospf-1)->area 2 stub
Router3(rw-config-ospf-1)->area 2 default-cost 15
```

Router 6

```
Router6(rw-config)->router ospf 1
Router6(rw-config-ospf-1)->area 2 stub
Router6(rw-config-ospf-1)->area 2 default-cost 20
```

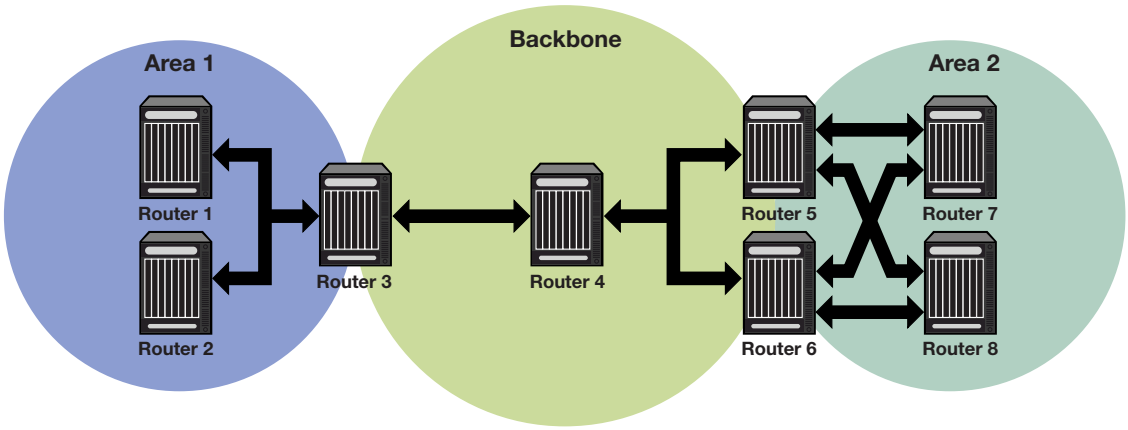
Router 7

```
Router7(rw-config)->router ospf 1
Router7(rw-config-ospf-1)->area 2 stub
```

Router 8

```
Router8(rw-config)->router ospf 1
Router8(rw-config-ospf-1)->area 2 stub
```

Figure 41-5 OSPF Stub Area Topology



Configuring a Not So Stubby Area (NSSA)

A Not So Stubby Area (NSSA) is a hybrid area using an Autonomous System Border Router (ASBR) to connect two disparate organizations. External routes are advertised as Type 7 LSAs and are converted to Type 5 LSAs before flooding to the backbone by the NSSA's ABR. Also, summary routes are allowed into the NSSA while external routes from other networks are still filtered from insertion into the NSSA.

External routes that are not imported into an NSSA can be represented by a default route. If the router is an ABR and has the highest router ID of all ABRs in the area, and no other ABR in the area is configured to translate always, it will translate Type 7 LSAs into Type 5 LSAs. Configuring the identity of the translator can be used to bias the routing to aggregated destinations. When translator role is set to Always, Type-7 LSAs are always translated regardless of the translator state of other NSSA border routers.

When a translating ABR loses a translator election, it will stop translating, and after a number of seconds (set by the **transstabilityint** option), it will flush any Type 5 LSAs resulting from aggregation. Any Type 5 LSAs resulting from direct translation of Type 7 LSAs will be allowed to age out. An ABR will always originate a default route into any attached NSSAs.

If the **no-summary** option is specified, the ABR does not send type 3 summary LSAs into the NSSA area, therefore all destinations outside of the NSSA area are represented by means of a default route.

Use the **area nssa** command to configure an area as a Not-So-Stubby-Area.

Example

Routers 2 and 6 are configured as the ABRs between Area 1 and 0, and Router 4 as the ASBR. Router 2 is configured to set Area 1 as an NSSA, and Type 7 routes from the connected domain will be translated to Type 5 routes into the backbone.

ABR Router 2 will only translate Type 7 LSAs; static routes redistributed by router 4. Also, Router 2 will always translate, since it is configured to do so; Router 6 will not, since only one ABR will perform the translation for a given area.

Router 4 will be configured to redistribute static routes.

The following example provides the input required to configure the NSSA topology as displayed in Figure 41-6 on page 41-15:

Router 6 (ABR)

```
Router 6(rw)->configure
Router 6(rw-config)->interface vlan 1
Router 6(rw-config-intf-vlan.0.1)->ip address 11.1.1.6 255.255.255.252
Router 6(rw-config-intf-vlan.0.1)->no shutdown
Router 6(rw-config-intf-vlan.0.1)->exit
Router 6(rw-config)->interface vlan 2
Router 6(rw-config-intf-vlan.0.2)->ip address 23.1.1.6 255.255.255.252
Router 6(rw-config-intf-vlan.0.2)->no shutdown
Router 6(rw-config-intf-vlan.0.2)->exit
Router 6(rw-config)->router ospf 1
Router 6(rw-config-ospf-1)->router-id 6.6.6.6
Router 6(rw-config-ospf-1)->area 1 nssa
Router 6(rw-config-ospf-1)->network 11.1.1.0 0.0.0.3 area 0
Router 6(rw-config-ospf-1)->network 23.1.1.0 0.0.0.3 area 1
Router 6(rw-config-ospf-1)->exit
```

Router 2(ABR)

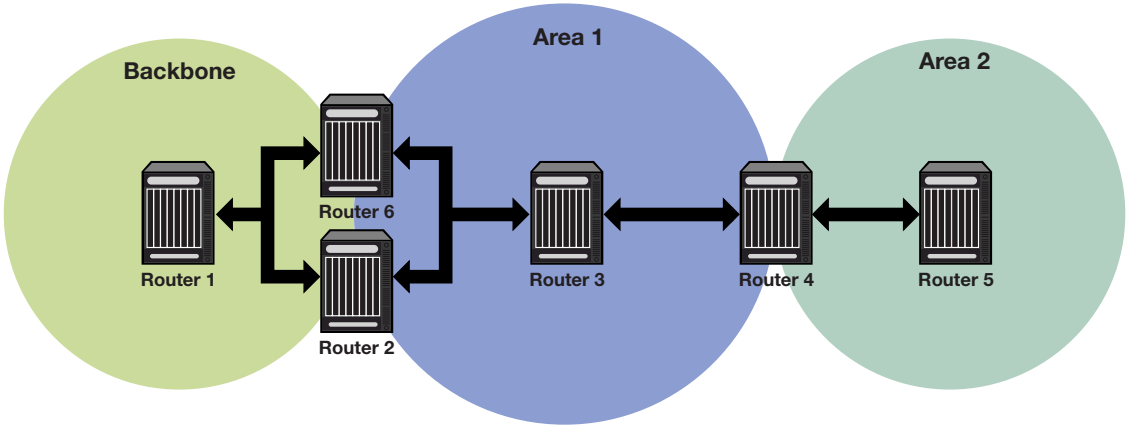
```
Router 2(rw)->configure
Router 2(rw-config)->interface vlan 1
Router 2(rw-config-intf-vlan.0.1)->ip address 11.1.1.2 255.255.255.252
Router 2(rw-config-intf-vlan.0.1)->no shutdown
Router 2(rw-config-intf-vlan.0.1)->exit
Router 2(rw-config)->interface vlan 2
Router 2(rw-config-intf-vlan.0.2)->ip address 23.1.1.1 255.255.255.252
Router 2(rw-config-intf-vlan.0.2)->no shutdown
Router 2(rw-config-intf-vlan.0.2)->exit
Router 2(rw-config)->router ospf 1
Router 2(rw-config-ospf-1)->router-id 2.2.2.2
Router 2(rw-config-ospf-1)->network 11.1.1.0 0.0.0.3 area 0
Router 2(rw-config-ospf-1)->network 23.1.1.0 0.0.0.3 area 1
Router 2(rw-config-ospf-1)->area 1 nssa
Router 2(rw-config-ospf-1)->area 1 nssa transrole always
Router 2(rw-config-ospf-1)->area 1 nssa-range 10.2.0.0 255.255.0.0
Router 2(rw-config-ospf-1)->exit
```

Router 4 (ASBR)

```
Router 4(rw)->configure
Router 4(rw-config)->interface vlan 2
Router 4(rw-config-intf-vlan.0.1)->ip address 23.1.1.2 255.255.255.252
Router 4(rw-config-intf-vlan.0.1)->no shutdown
Router 4(rw-config-intf-vlan.0.1)->exit
Router 4(rw-config)->interface vlan 3
Router 4(rw-config-intf-vlan.0.2)->ip address 30.1.1.1 255.255.255.252
Router 4(rw-config-intf-vlan.0.2)->no shutdown
```

```
Router 4(rw-config-intf-vlan.0.2)->exit
Router 4(rw-config)->router ospf 1
Router 4(rw-config-ospf-1)->router-id 4.4.4.4
Router 4(rw-config-ospf-1)->network 23.1.1.0 0.0.0.3 area 1
Router 4(rw-config-ospf-1)->redistribute static metric-type 1
Router 4(rw-config-ospf-1)->exit
```

Figure 41-6 OSPF NSSA Topology



Configuring Area Virtual-Links

The backbone area 0 cannot be disconnected from any other areas in the AS. Disconnected areas will become unreachable. To establish and maintain backbone connectivity, virtual-links can be configured through non-backbone areas for the purpose of connecting a disconnected area with the backbone through a backbone connected area. The two endpoints of a virtual link are ABRs, both of which belong to the backbone connected area (also referred to as the transit area); one of which belongs to the area disconnected from the backbone. Virtual links cannot be configured through stub areas (see "Configuring a Stub Area" on page 41-11 for stub area configuration information).

The virtual-link is treated as if it were an unnumbered point-to-point network belonging to the backbone and joining the two ABRs. The cost of a virtual link is not configured. It is auto configured with the cost of the intra-area path between the two ABRs that make up the virtual-link.

Use the **area virtual-link** command in OSPF router configuration command mode, providing the transit area ID and the ABRs IP address, to configure an area virtual-link.

Figure 41-7 on page 41-16 displays a typical virtual-link topology. Area 3 does not share an ABR with the backbone area, and is therefore disconnected from the backbone. Area 3 shares an ABR (router 2) with area 1. Area 1 has a second ABR (router 1) that it shares with the backbone. Area 1 is the transit area because it contains an ABR that it shares with the disconnected area and a second ABR that it shares with the backbone. By configuring an area virtual-link between router 2 and router 1, Area 3 will gain connectivity with the backbone and be able to learn routes for this AS.

Example

The following example presents the configuration required to configure the virtual-link displayed in Figure 41-7 on page 41-16:

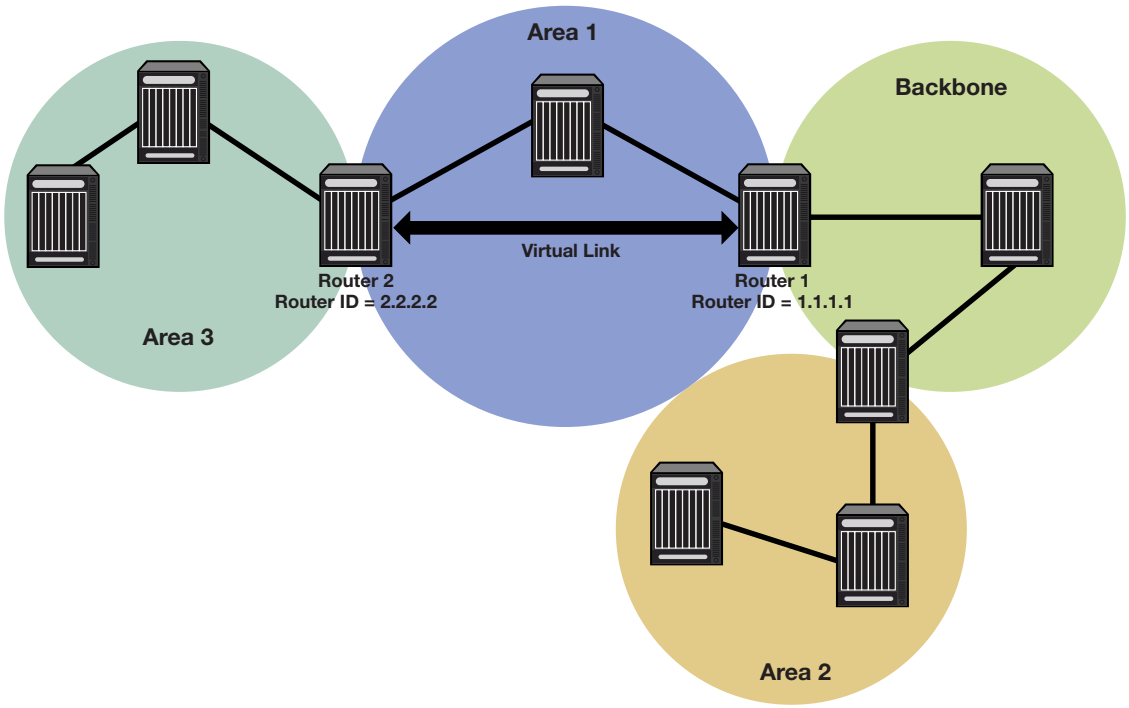
Router 1

```
Router 1(rw-config)->router ospf 1
Router 1(rw-config-ospf-1)->area 0.0.0.1 virtual-link 2.2.2.2
Router 1(rw-config-ospf-1)->exit
Router 1(rw-config)->
```

Router 2

```
Router 2(rw-config)->router ospf 2
Router 2(rw-config-ospf-2)->area 0.0.0.1 virtual-link 1.1.1.1
Router 2(rw-config-ospf-2)->exit
Router 2(rw-config)->
```

Figure 41-7 Virtual Link Topology



Configuring Area Virtual-Link Authentication

An area virtual-link can be configured for either simple or MD5 authentication.

Use the **area virtual-link authentication-key** command in OSPF router configuration command mode to configure simple authentication on this area virtual-link.

Use the **area virtual-link message-digest-key** command in OSPF router configuration command mode to configure MD5 authentication on this area virtual-link.

Configuring Area Virtual-Link Timers

The following timers can be configured for an area virtual-link:

- Dead-interval using the **area virtual-link dead-interval** command
- Hello-interval using the **area virtual-link hello-interval** command
- Retransmit-interval using the **area virtual-link retransmit-interval** command

- Transmit-delay using the **area virtual-link transmit-delay** command

See "Configuring OSPF Timers" on page 41-21 for an OSPF timers discussion.

Note: RFC 2328 specifies that the retransmit-interval should be greater than the expected round-trip delay between the two routers. This may be hard to estimate for a virtual link; it is better to err on the side of making it too large.

Configuring Route Redistribution

Redistribution permits the importation of other routing protocols into OSPF such as RIP, as well as static and directly connected routes. Alternately, you can specify a route-map for redistribution into OSPF. Be aware that if the referenced route map has not yet been configured, then an empty route map is created with the specified name. See "Configuring a Not So Stubby Area (NSSA)" on page 41-13 for an example of redistribution of static routes by an ASBR in an NSSA context.

Use the **redistribute** command in OSPF router configuration command mode to permit the redistributions of OSPF, RIP, static, or connected routes by this router.

Filtering Routes from the OSPF Route Table

Routes can be filtered from the OSPF route table by creating an OSPF filter route route-map and assigning it to the distribute-list for this OSPF router.

For example, the 10.1.1.0/24 network is advertised via OSPF from Area 0. However, private networks exist in 10.0.0.0/8. Various routers will learn the 10.1.1.0/24 route via OSPF, but they should not route packets to the 10.0.0.0/8 network. The solution is to not allow the 10.1.1.0/24 route to be installed in the forwarding tables by filtering it from the routing table with a route-map based upon its network address.

Use the **route-map filter** command as described in the "Route-Map Manager" section of the *Extreme Networks S-Series CLI Reference* to create an OSPF filter route route-map.

Use the **distribute-list route-map in** command to assign the filter route-map to the OSPF distribute-list.

Configuring Passive Interfaces

Passive interfaces explicitly allows the network to be advertised, but prevents it from forming neighbor relationships on that interface. Passive interfaces are included in the OSPF route table. They do not send or receive hello packets. OSPF adjacencies can not be formed on a passive interface.

An option exists to default all interfaces to passive mode.

Use the **passive-interface** command in router configuration command mode to configure an interface as passive.

Graceful Restart

OSPF graceful restart, sometimes referred to as non-stop forwarding, provides for an OSPF router to remain on the forwarding path during a restart of its OSPF software. Graceful-restart has three elements to its configuration: enabling, helper router, and restart interval.

Enabling graceful restart instructs the firmware to perform a graceful restart, rather than a standard OSPF restart. Restart is only initiated by a fail-over. Grace LSAs are sent when OSPF is restarted on another module. Whether the failover is intentional or not, the failed router protocol

is restarted on another module, and upon startup, OSPF sends grace LSAs out to its neighbors using existing link aggregation groups. Use the **graceful-restart enable** command to enable the graceful restart ability on this router.

The helper relationship with the restarting router is on a per network segment basis. The helper monitors the network for topology changes. If no changes occur, the helper router continues to advertise its LSAs as though no restart was occurring. If the restarting router was the designated router, the helper continues to treat it as such. If a topology change does occur, graceful restart is terminated on the restarting router and a standard restart occurs. Helper mode can be disabled on a restarting router neighbor using the **ip ospf helper-disable** command in interface command mode. If the restarting router receives an LSA indicating a disabled helper, the graceful restart terminates and a standard restart occurs.

A restart interval provides for a maximum time in seconds after which the graceful restart will terminate should it not complete or terminate for other reasons within the interval. Use the **graceful-restart restart-interval** command to change the restart interval setting.

View the router OSPF section of the **show running-config** display to verify any non-default graceful restart settings.

Graceful Restart and High Availability

The S-Series module supports single router high availability failover using the following components:

- OSPF graceful restart
- Non-stop router frame forwarding on each module
- Single router configuration
- Router protocol process failover to another module
- Link Aggregate Group (LAG) connectivity to neighboring routers

In a stable network, the route and rule information is fairly constant. If the router protocol process was to suddenly fail, forwarding information current at the time of the failure in all probability is usable for the short time after the failure until recovery occurs. During this recovery period, existing connections (that were not directly using the failed module) remain in effect. New connections continue to be installed using the last known “good” forwarding information. The router protocol process that failed is dynamically restarted. The user does not configure where the router process is running. The router forwarding process remains active on every module. The protocol process exchanges protocol and maintains state that it distributes to the other modules and does not have to run on any specific module. One exception to this rule is that the module must have 256M of memory to be router protocol process eligible.

Upon failure of a module running the router protocol process, the protocol process is started on a recovery module. One of the first messages it sends to its OSPF neighbors is a grace LSA. High availability failover will successfully occur if the following is true:

- The router is enabled for graceful restart
- The neighbors are enabled to participate as graceful restart helper
- The OSPF dead interval is configured for a sufficient period such that the grace LSA is received by its neighbors before the configured OSPF dead interval expires
- And each neighbor is a member of a LAG common to the failed router, allowing the neighbor to remain up

Figure 41-8 Physical and Logical Single Router HA Failover Configuration

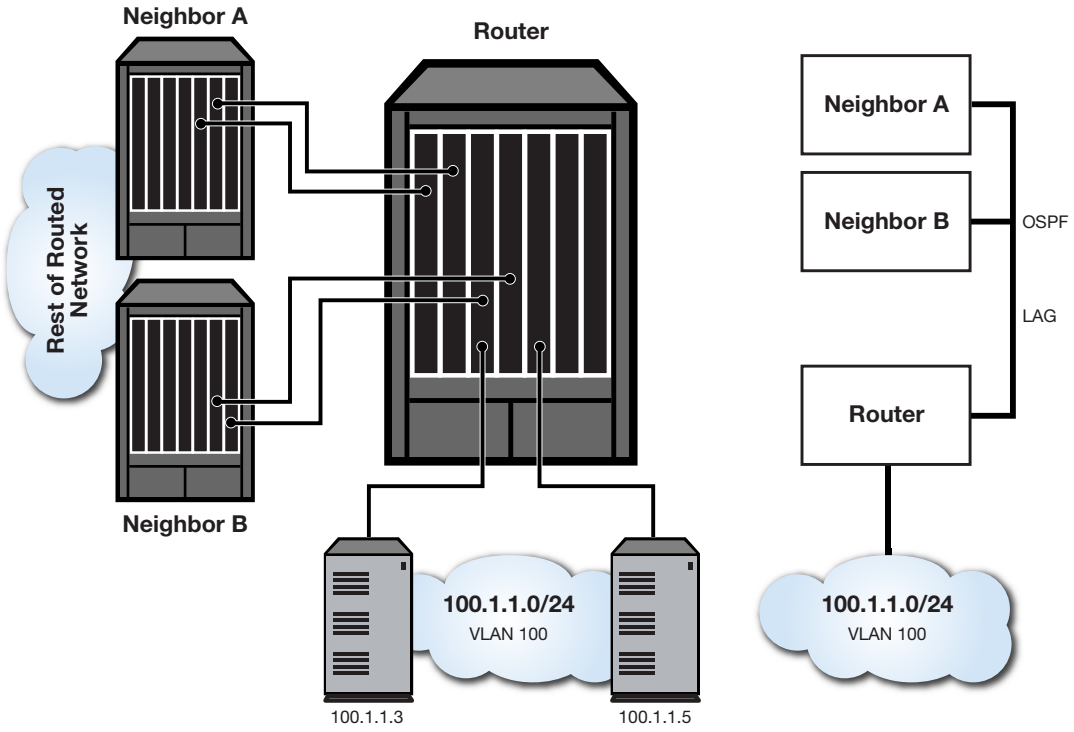


Figure 41-8 depicts the physical and logical configurations of the single router high availability failover mechanism. The neighbor to router lines display direct neighbor connections to the router enabled for OSPF graceful restart and members of LAGs common to the failing router. The server to router lines display VLAN connections common to both the failing and recovery routers.

Configuring Interface Cost

Each interface has an outbound cost associated with it. The lower the cost, the more likely the interface will be used to forward data traffic. Should several equal-cost routes to a destination exist, traffic is distributed equally among them.

The formula for calculating the OSPF interface cost metric is the reference bandwidth divided by the interface bandwidth. By default the reference bandwidth is set to 100 Mbps. For 10 Mbps links, the resulting cost is 10. For 100, 1000, or 10000 Mbps links, the resulting cost is 1. The reference bandwidth can be modified using the **auto-cost reference-bandwidth** command in OSPF configuration mode. The ability to re-center the reference bandwidth to a higher value, allows for OSPF interface costs to default to a value greater than 1 for 100, 1000, or 10000 Mbps links and greater than 10 for 10 Mbps links.

It is recommended that the auto cost reference bandwidth be the same value for all OSPF routers in the domain.

Use the **ip ospf cost** command in interface configuration command mode to statically specify the outbound cost of this interface. A statically configured OSPF interface cost overrides all other interface cost methods.

For logical interfaces containing multiple physical interfaces, such as a LAG, the aggregate interface speed is not readily available. A tracked object configured with the ports belonging to the logical interface can return the physical interface speed of each physical port specified in the tracked object. OSPF will sum the returned interface speeds and use that aggregate value when calculating OSPF interface cost. Because the tracked object will report when a physical interface is

up or down, OSPF will dynamically adjust the aggregate speed when an interface becomes active or goes down and adjust the OSPF interface cost accordingly. This method should be used in LAG and ECMP logical interface contexts.

Note: The speed used in the cost calculation is sum of all ports capabilities in the tracked object. Setting the speed manually will not change the tracked interface speed. A 1GB capable port has a 1 GB speed regardless of the manual speed setting. The same holds true for ports that auto-negotiate to a lower speed. The expectation is that both sides of the link are using the same ports and SFP connectors and should result in the same speed.

Use the **ip ospf cost track** command in interface configuration mode to calculate the OSPF interface cost based upon summing physical interface speeds that belong to a logical interface.

When adding an additional physical port to a logical interface that uses the interface summation method to determine OSPF interface cost, you must also add the physical port to the associated tracked object.

See "Tracked Object Manager Configuration" on page 13-1 for tracked object configuration details.

Configuring OSPF with Authentication at the Interface

Authentication helps ensure that routing information is processed only from trusted routers. This section describes OSPF authentication at the interface level. Two authentication schemes can be used, simple using the **ip ospf authentication-key** command or MD5 using the **ip ospf message digest key md5** command, but a single scheme must be configured for each network. The use of different schemes enables some interfaces to use much stricter authentication than others. When you wish to bar routers from exchanging OSPF packets, use simple authentication. The interfaces that the packets will be sent on still must be trusted because the authentication key will be placed in the packets and are visible to anyone on the network. An adjacency with another router will not occur unless the simple authentication is configured the same on both ends of the interface.

If you do not trust other routers on your network, use MD5 authentication. The system works by using shared secret keys. Because keys are used to sign the packets with an MD5 checksum through a one-way hash function, they cannot be forged or tampered with. Also, because the keys are not included in the packet, snooping the key is impossible. Network users can still snoop the contents of packets, though, because packets are not encrypted.

S-Series device MD5 authentication is compliant with OSPF RFC 2328. This specification uses the MD5 algorithm and an authentication key of up to 16 characters.

Configuring Bidirectional Forwarding Detection (BFD) on Interfaces

BFD is used to detect a communications failure with an OSPF forwarding plane next-hop. BFD detects failures in under one second. BFD augments the OSPF Hello mechanism. The OSPF Hello interval defaults to 10 seconds. With high speed data rates, a failure requiring multiple seconds to detect can result in significant data loss. The OSPF implementation of the BFD protocol uses the following non-configurable parameters:

Transmit Interval – The period of time between the transmission of BFD control packets, set for 100ms.

Receive Interval – The period of time between received BFD control packets, set for 100ms.

Detection Multiplier – The Number of consecutive control packets that can be missed before the BFD session transitions to down, set to 3.

Use the **bfd all-intfs-on** command in OSPF router configuration mode to enable BFD on all OSPF interfaces.

Configuring OSPF Timers

There are five OSPF timers:

- Hello-Interval
- Dead-Interval
- Retransmit-Interval
- Transmit-Delay
- SPF-Delay

To ensure efficient adjacency between OSPF neighbors, the S-Series device provides hello-interval and dead-interval commands. The hello interval is the period between transmissions of hello packet advertisements. The dead interval is the period that can elapse without receiving a router's hello packets before its neighbors will declare it down.

Use the **ip ospf hello-interval** command in interface configuration command mode to configure the period between transmissions of hello packet advertisements.

Use the **ip ospf dead-interval** in interface configuration command mode to configure the period between receiving hello packets before the associated neighbor is declared down.

In order to ensure that flooding is reliable, LSAs are retransmitted until they are acknowledged. The period between retransmissions is the retransmit-interval. If this interval is set too low for an interface, needless retransmissions will take place. If the value is set too high, the speed of the flooding, during the period of lost packets, may be affected.

Use the **ip ospf retransmit-interval** command in interface configuration command mode to configure the retransmit-interval.

The transmit-delay is an estimation of the number of seconds it takes to transmit a link state update packet over this interface. This value should take into account transmission and propagation delays.

Use the **ip ospf transmit-delay** command in interface configuration command mode to configure the transmit-delay.

The SPF-delay is the amount of time that transpires between the receipt of an OSPF update and the SPF calculation.

Use the **timers spf** command in OSPF router configuration command mode to specify the amount of time between receiving an OSPF update and an SPF calculation occurring.

The OSPF timers can also be configured for an area virtual-link. See "Configuring Area Virtual-Links" on page 41-15.

Configuring the PE-CE Protocol

The PE-CE protocol allows a service provider offering Virtual Private Network (VPN) services to their customers to peer Customer Edge (CE) routers with their Provider Edge (PE) routers . RFC 4577 defines how the PE-CE protocol is implemented using the OSPF routing protocol.

When the PE router becomes a routing peer of the CE router, the PE router learns the routes that lead to the CE's site and can redistribute those routes to other PE routers that attach to the same VPN.

Enabling PE-CE enables the following functionality:

- DN Bit
- Sham link

- Domain tag
- Domain ID

Use the **enable-pe-ce** command in OSPF configuration mode to enable the PE-CE protocol on the router.

The OSPF VRF Domain Tag

The configuration and inclusion of the OSPF VRF domain tag is required for PE-CE protocol enabled systems to be backward compatible with systems that do not set the PE-CE protocol DN bit in type 5 LSAs. When a prefix is received from a BGP speaker and redistributed into the PE-CE protocol enabled OSPF instance, the OSPF process for the VRF is given a domain tag. In the event that the customer site attempts to re-advertise the prefix to another PE using the same domain tag, the domain tag will be matched and the prefix will not be accepted by the second PE for redistribution into BGP. Setting the same domain tag for all backbone PE routers on the same VPN prevents routing loops.

Setting the OSPF VRF domain tag is optional when the PE-CE protocol is enabled for all PE backbone routers for a given VRF. If legacy PE routers that do not support the PE-CE protocol are present in the VRF backbone, set the domain tag for this router to agree with the domain tag of the legacy router.

The PE-CE protocol must be enabled using the **enable-pe-ce** command to set the OSPF VRF domain tag.

Use the **domain-tag** command in OSPF configuration mode to set the domain-tag for this PE router.

The OSPF VRF Domain ID

If the OSPF instances of an OSPF domain are given one or more domain IDs, OSPF can determine whether an OSPF-originated VPN-IPv4 route belongs to the same domain as a given OSPF instance and whether the route should be redistributed to that OSPF instance as an inter-area route or as an OSPF AS-external route.

If two OSPF instances with a domain ID configured are in the same OSPF domain, the PE-CE protocol requires that the primary domain ID of the second instance must be one of the domain IDs of the first instance (either primary or secondary). If two OSPF instances with a domain ID configured are not in the same OSPF domain, the primary domain ID of each instance must not be configured as a domain ID of the other OSPF instance.

The PE-CE protocol must be enabled using the **enable-pe-ce** command to set the OSPF VRF domain ID.

Use the domain-id command in OSPF configuration mode to set the domain ID for this PE router.

Redistribute BGP into OSPF

OSPF supports the redistribution of BGP discovered routes into OSPF. A **global** option is available for the redistribution of BGP prefixes from the global router. When using the **global** option, VPN4 address prefixes are appropriately translated.

Use the **redistribute bgp global** command in OSPF configuration mode to redistribute BGP learned routes to other PE routers in this VPN.

OSPF Sham Link

If a VRF contains both an OSPF-distributed route and a VPN-IPv4 route for the same IPv4 prefix, then the backdoor OSPF-distributed route is preferred over the VPN backbone route, unless the

next hop interface for an installed (OSPF distributed) route is the sham link, in which case, the VPN backbone VPN-IPv4 route is used.

If it is desired to have OSPF prefer the routes through the VPN backbone over the routes through the OSPF backdoor link, then the routes through the backbone must appear to be intra-area routes. The sham link provides this appearance of an intra-area link connecting the two PE routers.

Use the **area sham-link** command in OSPF configuration mode, specifying both the source and destination link addresses to configure an OSPF sham link between two VPN PE routers.

Configuring OSPF

This section provides details for the configuration of OSPF on S-Series platforms.

Default Settings

Table 41-1 lists OSPF parameters and their default values.

Table 41-1 Default OSPF Parameters

Parameter	Description	Default Value
router ID	Provides for the identification of one router to another and helps establish adjacencies among OSPF routers.	highest IP address of configured routing interfaces
interface cost	An outbound interface value used in determining which routing interface should forward when more than one routing interface is available.	10
interface priority	A value placed on the interface that helps in determining which router will be elected designated router.	1
interface network type	Specifies the type of network an interface is connecting to.	broadcast
LSA Thresholds	Specifies : • The number of LSA updates that force a full routing calculation • The number of LSA updates that interrupt and restart a full routing calculation • The number of LSA inter-area/external updates that force a full routing calculation • the number of intra updates that force a full routing calculation	4294967295 Update starts 4294967295 Update restarts 50 Inter-area/external updates 0 Intra updates
LSA Pause Frequency	Specifies the number of units SPF calculation runs before pausing.	10000
SPF delay timer	Specifies the amount of time between receiving an OSPF update and the start of an SPF calculation.	5 seconds
retransmit interval	A timer that determines the retransmission of LSAs in order to ensure reliable flooding.	5 seconds

Table 41-1 Default OSPF Parameters (continued)

Parameter	Description	Default Value
transmit delay	Specifies the number of seconds it takes to transmit a link state update packet over this interface.	1 second
hello interval	The period between transmissions of hello packet advertisements.	10 seconds for broadcast and point-to-point networks; 30 seconds for non-broadcast and point-to-multipoint networks
dead interval	The period that can elapse without receiving a router's hello packets before its neighbors will declare it down.	40 seconds
distance	Specifies the administrative distance for OSPF routes. The available protocol with the lowest administrative distance is chosen for this route.	connected = 0 static = 1 OSPF = 110 RIP = 120
graceful-restart	Provides for an OSPF router to remain on the forwarding path during a restart of its OSPF software.	disabled
graceful-restart restart interval	Specifies the maximum time in seconds after which the graceful restart will terminate should it not complete or terminate for other reasons within the interval.	120 seconds
PE-CE Protocol	Enables the Customer Edge (CE) router as a Provider Edge (PE) router peers.	disabled

Procedure 41-1 describes how to configure basic OSPF parameters. All commands in this procedure are entered in OSPF router configuration command mode, except where indicated.

Procedure 41-1 Configuring Basic OSPF Parameters

Step	Task	Command(s)
1.	Configure an IP address for all routing interfaces in the AS. • **primary** - (Optional) Specifies that the configured IP address is a primary address. • **secondary** - (Optional) Specifies that the configured IP address is a secondary address.	**ip address** {*ip-address* | *ip-address*/*prefixLength*} *ip-mask* [**primary** | **secondary**]
2.	Create an OSPF routing instance.	**router ospf** *process-id*
3.	Configure the network addresses, masks, and areas for each subnet on this AS. • **area** - Specifies the *area-id* to be associated with the OSPF address range. Valid values are decimal values between **0** - **4294967295** or an IP address. A subnet address can be specified as the area-id to associate areas with IP subnets.	**network** *ip-address wildcard-mask* **area** *area-id*

Procedure 41-2 describes how to configure basic OSPF parameters.

Procedure 41-2 Configuring OSPF General Optional Parameters

Task	Command(s)
Optionally, change the OSPF router ID for this device.	**router-id** *ip-address*
Optionally, enable the OSPF PE-CE protocol.	**enable-pe-ce**
Optionally, configure the OSPF VRF domain tag.	**domain-tag** *tag*
Optionally, configure the OSPF VRF domain ID.	**domain-id** [**secondary**] **type** *type* **value** *value*
Optionally, configure the OSPF router neighbors for this router.	**neighbor** *ip-address* [**priority** *priority*]
Optionally, change the SPF LSA thresholds for this router.	**spf lsa-thresholds** *num-start num-restart num-intra-full num-ia-ext-full*
Optionally, change the SPF pause frequency to specify the number of units SPF calculation runs before pausing.	**spf pause-frequency** *units*
Optionally, change the delay, in milliseconds, between the receipt of an update and the beginning of the SPF execution.	**timers spf** *spf-delay*
Optionally, enable BFD on all OSPF interfaces.	**bfd all-intfs-on**
Optionally, change the administrative distance for OSPF routes.	**distance** [**ospf** {**external** | **intra-area**}] *weight*
Optionally, define the range of addresses used by this Area Border Router (ABR) when communicating routes to other areas.	**area** *area-id* **range** *ip-address ip-mask* [**not-advertised**]
Optionally, configure an area as a stub area.	**area** *area-id* **stub** [**no-summary**]
Optionally, set the cost for the default route that is sent into a stub area by an ABR.	**area** *area-id* **default-cost** *cost*
Optionally, configure an area as a not so stubby area.	**area** {*area-id* | *ip-address*} **nssa** [**no-summary**] [**transstabilityint** *seconds*] [**transrole always**]
Optionally, configure an Autonomous System Border Router (ASBR) to summarize Type 7 to Type 5 routes matching the specified address and mask.	**area** {*area-id* | *ip-address*} **nssa-range** *ip-address mask*
Optionally, configure an OSPF sham link between two PE routers.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address*
Optionally, configure an OSPF sham link authentication key password.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **authentication-key** *password*
Optionally, modify the OSPF sham link dead interval.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **dead-interval** *seconds*
Optionally, modify the OSPF sham link hello interval.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **hello-interval** *seconds*
Optionally, configure the OSPF sham link keychain.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **keychain** *name*

Procedure 41-2 Configuring OSPF General Optional Parameters (continued)

Task	Command(s)
Optionally, specify an OSPF sham link message digest key and MD5 authentication key.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **message-digest-key** *digest-key* **md5** *auth-key*
Optionally, modify the OSPF sham link retransmit interval.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **retransmit-interval** *seconds*
Optionally, modify the OSPF sham link transmit delay period.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **transmit-delay** *seconds*
Optionally, modify the OSPF sham link cost.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **cost** *cost*
Optionally, configure an OSPF virtual-link, which represents a logical connection between the backbone and a non-backbone OSPF area.	**area** *area-id* **virtual-link** *ip-address* **area** *area-id* **virtual-link** *ip-address* **authentication-key** *key* **area** *area-id* **virtual-link** *ip-address* **dead-interval** *seconds* **area** *area-id* **virtual-link** *ip-address* **hello-interval** *seconds*
	area *area-id* **virtual-link** *ip-address* **message-digest-key** *digest-key* **md5** *format* **line** *auth-key* **area** *area-id* **virtual-link** *ip-address* **retransmit-interval** *seconds* **area** *area-id* **virtual-link** *ip-address* **transmit-delay** *seconds*
Optionally, change the bandwidth reference setting used for calculating interface cost for this OSPF instance.	**auto-cost reference-bandwidth** *bandwidth-multiplier*
Optionally, enable passive OSPF on the specified interface.	**passive-interface** {*vlan-id* | *interface-name* | **default**}
Optionally, allow routing information discovered through non-OSPF protocols to be distributed in OSPF update messages.	**redistribute** {**rip** | **static** | **connected**} [**route-map** *id-number*] [**metric** *metric value*] [**metric-type** *type-value*] [**tag** *tag*]
Optionally, assign an OSPF route filter route-map to the OSPF distribute-list.	**distribute-list route-map** *name* **in**
Optionally, enable the graceful-restart feature on this router.	**graceful-restart enable**
Optionally, change the graceful-restart restart interval for this router.	**graceful-restart restart-interval** *interval*
Optionally, in system command mode, reset the specified OSPF process ID or the OSPF process.	**clear ip ospf process** [*process-id*]
Optionally, in global configuration command mode, enable OSPF protocol debugging output for the specified subsystem.	**debug ip ospf** {*subsystem*}
Optionally, enable this OSPF router for RFC 1583 compatibility.	**rfc1583compatible**

Procedure 41-3 describes how to configure optional OSPF interface parameters. All commands in this procedure are entered in interface configuration command mode.

Procedure 41-3 Configuring OSPF Optional Interface Parameters

Step	Task	Command(s)
1.	Optionally, change the cost of sending an OSPF packet on this router interface. This setting overrides all other interface cost methods.	**ip ospf cost** *cost*
2.	Optionally, sum the interface speeds contained in the specified tracked object when calculating the OSPF interface cost.	**ip ospf cost track** *trackobject-name*
3.	Optionally, change the OSPF priority value for this router interface.	**ip ospf priority** *number*
4.	Optionally, change the OSPF poll-interval value for this non-broadcast neighbor.	**ip ospf poll-interval** *seconds*
5.	Optionally, change the amount of time between retransmissions of LSAs for adjacencies that belong to this interface.	**ip ospf retransmit-interval** *seconds*
6.	Optionally, change the amount of time required to transmit a link state update packet on this interface.	**ip ospf transmit-delay** *seconds*
7.	Optionally, enable the ignore MTU advertisement feature for the neighbor of this interface.	**ip ospf ignore-mtu**
8.	Optionally, change the number of seconds this router must wait before sending a hello packet to neighbor routers on this interface.	**ip ospf hello-interval** *seconds*
9.	Optionally, change the number of seconds this router must wait to receive a hello packet from its neighbor before determining that the neighbor is out of service.	**ip ospf dead-interval** {*seconds* | **minimal hello-multiplier** *number*}
10.	Optionally, assign a password on this interface to be used by neighboring routers using OSPF's simple password authentication.	**ip ospf authentication-key** *password*
11.	Optionally, enable OSPF MD5 authentication on this interface.	**ip ospf message-digest-key** *keyid* **md5** *key*
12.	Optionally, disable the graceful restart helper feature on this interface.	**ip ospf helper-disable**
13.	Optionally, specify the network type that this interface is connected to.	**ip ospf network** {**non-broadcast** | **broadcast** | **point-to-point** | **point-to-multipoint**}

Table 41-2 describes how to display OSPF configuration and statistics.

Table 41-2 Displaying OSPF Configuration and Statistics

Task	Command(s)
Displaying OSPF configuration.	**show ip ospf**
Displaying OSPF link state database information.	**show ip ospf database** [*link-state-id*]

Table 41-2 Displaying OSPF Configuration and Statistics

Task	Command(s)
Displaying information about OSPF internal entries to area border routers and autonomous system boundary routers.	**show ip ospf border-routers**
Displaying OSPF interface configuration information.	**show ip ospf interface** [**vlan** *vlan-id*]
Displaying OSPF neighbor information.	**show ip ospf neighbor** [**detail**] [*ip-address*] [**vlan** *vlan-id*]
Displaying OSPFv3 sham link information.	**show ipv6 ospf sham-link**
Displaying OSPF virtual-links configuration information.	**show ip ospf virtual-links**

42

Open Shortest Path First Version 3 (OSPFv3) Configuration

This chapter provides the following information about configuring and monitoring OSPFv3 on Extreme Networks S-Series devices:

Using the OSPFv3 Protocol in Your Network

Open Shortest Path First Version 3 (OSPFv3) is the OSPF routing protocol for IPv6. OSPFv3 is considered a TCP/IP internet routing Interior Gateway Protocol (IGP). OSPFv3 distributes routing information between routers belonging to a single Autonomous System (AS). The OSPF protocol is based on link-state or SPF technology. The advantages associated with a link-state routing protocol are:

- Rapid convergence
- Reduced routing update traffic over traditional distance-vector protocols

This OSPFv3 implementation supports RFC 2740 *OSPF for IPv6*.

The OSPFv3 protocol is designed expressly for the TCP/IP internet environment. OSPFv3 utilizes IP multicast when sending and receiving routing updates. Routing updates are optionally authenticated using IPsec for OSPFv3.

OSPFv3 routes IP packets based solely on the destination IP address found in the IP packet header. IP packets are not encapsulated in any further protocol headers as they transit the AS. OSPFv3 is a dynamic routing protocol in that it quickly detects topological changes in the AS, such as router interface failures, and calculates new loop-free routes after a period of convergence. This period of convergence is short and involves a minimum of routing traffic. In a link-state routing protocol, each router maintains a database describing the AS's topology. This database is referred to as the link-state database. Each participating router has an identical database. Each individual database entry is a particular router's local state made up of such information as the router's usable interfaces and reachable neighbors. The router distributes its local state throughout the AS by flooding.

Each network that has at least two attached routers has a designated router. The designated router generates an LSA for the network and has other special responsibilities in the running of the

protocol, enabling a reduction in the number of adjacencies required on a network. This in turn reduces the amount of routing protocol traffic and the size of the link-state database.

All routers run the exact same algorithm, in parallel. From the link-state database, each router constructs a tree of shortest paths with itself as root. This shortest-path tree provides the route to each destination in the AS. Externally derived routing information appears on the tree as leaves. When several equal-cost routes to a destination exist, traffic is distributed equally among them. The cost of a route is described by a single dimensionless metric.

OSPF allows sets of networks to be grouped together. Such a grouping is called an area. The topology of an area is hidden from the rest of the AS. This information hiding enables a significant reduction in routing traffic. Also, routing within the area is determined only by the area's own topology, lending the area protection against bad routing data. An area is a generalization of an IP subnetted network. OSPF enables the flexible configuration of IP subnets. Each route distributed by OSPF has a destination. Two different subnets of the same IP network number may have different masks providing a different range of addresses for that subnet. This is commonly referred to as Variable Length Subnet Masking (VLSM). A packet is routed to the longest or most specific match. Host routes are considered to be subnets whose masks are "all ones" (0xffffffff).

If IPsec for OSPFv3 is enabled on the interface, OSPFv3 protocol exchanges are authenticated. This means that only trusted routers can participate in the AS's routing. The S-Series platform supports IPsec for OSPFv3. See "IPsec for OSPFv3" on page 42-4 for a listing of supported authentication and encapsulation algorithms.

Route redistribution is supported for RIP, connected, BGP, and static routes.

The Bidirectional Forwarding Detection (BFD) protocol providing sub-second failure detection on OSPF forwarding interfaces is enabled by default on all OSPF interfaces.

An OSPF Customer Edge (CE) router can be configured as a peer to a Provider Edge (PE) router by enabling the PE-CE protocol on the PE-CE associated routers.

OSPFv3 is similar to OSPFv2 in its usage of the SPF algorithm, flooding, Designated Router (DR) election, timers, metrics, concept of a link-state database, intra/inter area and AS external routes and virtual-links. OSPFv3 differs with OSPFv2 in many respects, as outlined in "OSPFv3 and OSPFv2 Differences" on page 42-2.

OSPFv3 is not backward compatible with OSPFv2. If you need to route both IPv4 and IPv6 using OSPF, enable both OSPFv3 and OSPFv2 on the device.

OSPFv3 and OSPFv2 Differences

OSPFv3 differs from OSPFv2 in a number of respects:

- OSPFv3 processing is per link. OSPFv3 is link rather than subnet centric. An OSPFv3 interface connects to a link, not a subnet. This change has both functional and efficiency advantages. Multiple IP subnets can be assigned to a single link. Two nodes can talk directly over a single link. In OSPFv3, terms such as network and subnet should generally be replaced with the term link in order to understand OSPFv3 processing.
- With the exception of the new link LSA, OSPFv3 LSAs do not carry IPv6 addresses. The removal of addressing from the LSAs has scaling advantages. Router and network LSAs now only contain topology information necessary for SPF processing and no longer contain network addresses. OSPFv3 LSAs do maintain 32-bit RIDs and LSA IDs. Because OSPFv3 IDs are still expressed in dotted-quad notation, OSPFv3 networks can be easily overlayed on an OSPFv2 network.
- All OSPFv3 neighbors are identified by the neighbor router ID. The OSPFv2 behavior of identifying neighbors on broadcast and Non-Broadcast-Multi-Access (NBMA) links by their interface address has been removed.

- OSPFv3 uses link-local addresses, which begin with FF80::/10, as source and next-hop addresses. Link-local addresses are for use on a single link for purposes including neighbor discovery and auto-configuration. On all interfaces, except virtual, OSPF packets are sent using the interface's link-local unicast address as the source. Because all OSPF traffic transits the network on a link basis, IPv6 does not forward (route) IPv6 datagrams having link-local source addresses.
- OSPF specific authentication has been removed and replaced by optionally configuring IPsec for OSPFv3 as defined in RFC 4552. If IPsec for OSPFv3 is not enabled on the interface, OSPFv3 authentication does not take place for OSPF packets.
- RFC 1583 compatibility does not apply to OSPFv3.
- To take advantage of IPv6's link-local scope, OSPFv3 adds a link-local flooding scope to the domain and area flooding scopes present in OSPFv2. The link LSA, which has link-local flooding scope and can not be flooded beyond any attached router, has been added for neighbors on a single link.
- Two new LSAs have been introduced: the link LSA and the intra-area LSA. Point-to-point links are supported in order to enable operation over tunnels. OSPFv3 views IPv6-over-IPv4 tunnels as a point-to-point interface with a link-local address and possibly a global unicast address. OSPFv3 uses the reported MTU for tunnel interfaces.

  The prefix advertisement for OSPFv3 is now in the new intra-area prefix LSA. When information is only relevant to the connected neighbor, OSPFv3 puts it in the link LSA, not in the router or network LSA, in both cases avoiding flooding information beyond the relevant information scope.

Table 42-1 details the supported LSA types by LS ID and name for OSPFv3 and OSPF v2.

Table 42-1 OSPFv3 and OSPFv2 LSA Cross-Reference

OSPFv3 LSAs	OSPFv2 LSAs
0x2001 – Router LSA	1 – Router LSA
0x2002 – Network LSA	2 – Network LSA
0x2003 – Inter-Area Prefix LSA	3 – Network Summary LSA
0x2004 – Inter-Area Router LSA	4 – ASBR Summary LSA
0x2005 – AS-External LSA	5 – AS-External LSA
0x2006 – Group Membership LSA	6 – Group Membership LSA
0x2007 – Type-7 LSA	7 – NSSA External LSA
0x2008 – Link LSA	No Corresponding LSA for OSPFv2
0x2009 – Intra-Area Prefix LSA	No Corresponding LSA for OSPFv2

- Unlike for OSPFv2, the router and network LSAs for OSPFv3 do not advertise prefixes. In OSPFv3, the router and network LSAs only represent the router's node information for SPF and are only flooded if information relevant to the SPF algorithm changes. This behavior avoids the flooding of prefix changes that are not relevant to SPF.

  Inter-area prefix, inter-area router, and type-7 LSAs have the same function as their OSPFv2 counterparts listed in Table 42-1.
- OSPFv3 specifies the processing of unsupported LSAs. Unsupported LSAs are maintained in the database and flooded according to scope. In OSPFv3, routers with 100 or more interfaces generate more than one router LSA. A new link LSA has been created. Addresses in LSAs are specified as [prefix, prefix length].

- OSPFv2 supports multiple OSPF processes on a device. The current OSPFv3 implementation supports a single OSPFv3 process.
- OSPFv3 supports multiple OSPFv3 instances on an interface. Multiple OSPFv3 instances provide for the sharing of an interface when more than one physical network segment needs access to an interface. It also provides for the configuration of multiple areas on a single interface. The multiple OSPF instances feature is not supported by OSPFv2.
- The IPv6 all SPF routers multicast address is FF02::5; the all DRouters multicast address is FF02::6. Both have link-local scope.

OSPFv3 and OSPFv2 Similarities

- OSPFv3 uses the same 5 message types, with the same message numbering, as OSPFv2:
  - 1 – Hello
  - 2 – Database Description
  - 3 – Link-State database Request
  - 4 – Link-State Database Update
  - 5 – Link-State Database Acknowledgment

  Keep in mind that OSPFv3 message header fields differ in that there:
  - Are no fields for authentication
  - Is an instance ID field that has local link significance only
- The mechanisms for neighbor discovery and adjacency formation have not changed.
- The supported interface types point-to-point, point-to-mulitpoint, broadcast, NBMA, and virtual have not changed.
- LSA flooding and aging have not changed.
- All of OSPFv2 optional capabilities, including on-demand circuit support, NSSA areas, and the multicast extensions to OSPF are supported in OSPFv3.
- Area ID and Router ID remain 32 bit identifiers. Areas can be configured for Not-So-Stubby-Area (NSSA), Stub Area, and virtual-links.

IPsec for OSPFv3

IPsec authentication and encrypted authentication are supported. The IPsec authentication algorithms supported are:

- Message-Digest algorithm 5 (**MD5**)
- Secure Hash Algorithm 1 (**SHA1**)
- Advanced Encryption Standard Cipher Algorithm in Cipher Block Chaining (**AESCBC**)

The IPsec encryption algorithms supported are:

- Triple Data Encryption Standard (**3DES**)
- **AESCBC** (with 128, 192, or 256 bit keys)

Implementing OSPFv3

To implement OSPFv3 in your network:

- Map out the AS including routers and the areas to which they belong
- Create an OSPFv3 routing instance for this AS
- Configure each router with a router ID
- Configure the area that each router belongs to
- Enable OSPFv3 on each routing interface for the router specifying the OSPFv3 process, area and optional instance
- Optionally enable IPsec and configure IPsec authentication or encrypted authentication on each routing interface
- Optionally determine which router will be the designated router and backup and configure OSPF priority values on each routing interface accordingly
- Optionally configure OSPFv3 timers
- Optionally, configure the protocols and route types that will be redistributed over this AS
- Optionally configure interface cost
- Optionally modify the administrative distance for OSPF routes
- Optionally enable graceful restart
- Optionally enable the BFD protocol on all OSPF interfaces
- Optionally enable the PE-CE protocol on the router
  - With the PE-CE protocol enabled, optionally, configure a primary or secondary domain ID for this router
  - Optionally redistribute BGP discovered routes over OSPF

OSPFv3 Configuration Overview

OSPFv3 is enabled by creating an OSPFv3 process using the **ipv6 router ospf** command in OSPFv3 router configuration mode. Once an process is created, the router's OSPF settings are configured with respect to the process ID and IP interfaces. By default, OSPFv3 is disabled on the S-Series device. Be aware that unspecified parameters use their default values, and any parameters specified at the interface level will override the values specified at the area level for that interface.

Configuring Basic OSPFv3 Parameters

Basic OSPFv3 configuration consists of:

- Creating an OSPFv3 routing instance on the router
- Entering interface configuration mode for the routing interfaces for this device
- Enabling OSPFv3 on each routing interface, specifying the OSPFv3 process, area, and optional instance
- Enabling the interface

Configuring a Routing Instance

OSPFv3 routing configuration takes place within a routing instance. Configure a routing instance using the **ipv6 router ospf** command in global configuration command mode. Executing this command places you in the OSPFv3 router configuration command mode for the specified OSPFv3 router instance.

The following example creates an OSPFv3 routing instance 23 on the router:

```
S Chassis(su)->configure
S Chassis(su-config)->ipv6 router ospf 23
S Chassis(su-config-ospfv3)->
```

The IPv6 Link-Local Address

OSPFv3 uses the IPv6 link-local interface addresses as both the source and next-hop addresses. A single link-local address is supported per interface. IPv6 link-local addresses begin with FF80::/10. An IPv6 link-local address is autogenerated when you enable IPv6 on the interface, based upon the the MAC address associated with the interface. Use the **ipv6 enable** command to enable IPv6 on a non-routing IPv6 interface. Use the **ipv6 forwarding** command to enable IPv6 on a routing interface. Use the **ipv6 address** command **link-local** option to manually configure the IPv6 link-local address. When manually configuring a link-local address, if a link-local address already exists on the interface, a warning displays asking you if you wish to change it.

Enable the interface using the **no shutdown** command.

Configuring OSPFv3 on the Routing Interface

OSPFv3 is enabled on an IPv6 routing interface using the **ipv6 ospf** command, specifying the OSPFv3 process and the OSPF area to which the interface belongs. A basic OSPFv3 routing interface configuration does not require a configured IPv6 address, only the link-local. If you want to configure an IPv6 address on the interface, use the **ipv6 address** command to manually configure an IPv6 address on the interface or the **ipv6 address autoconfig** command to have an IPv6 address autogenerated for the interface. The following OSPFv3 basic routing interface example:

- Creates interface VLAN 1255
- Enables IPv6 forwarding on the interface
- Displays a summary of the the IPv6 interface configuration
- Enables OSPFv3 for process 23 area 0.0.0.1 on the interface

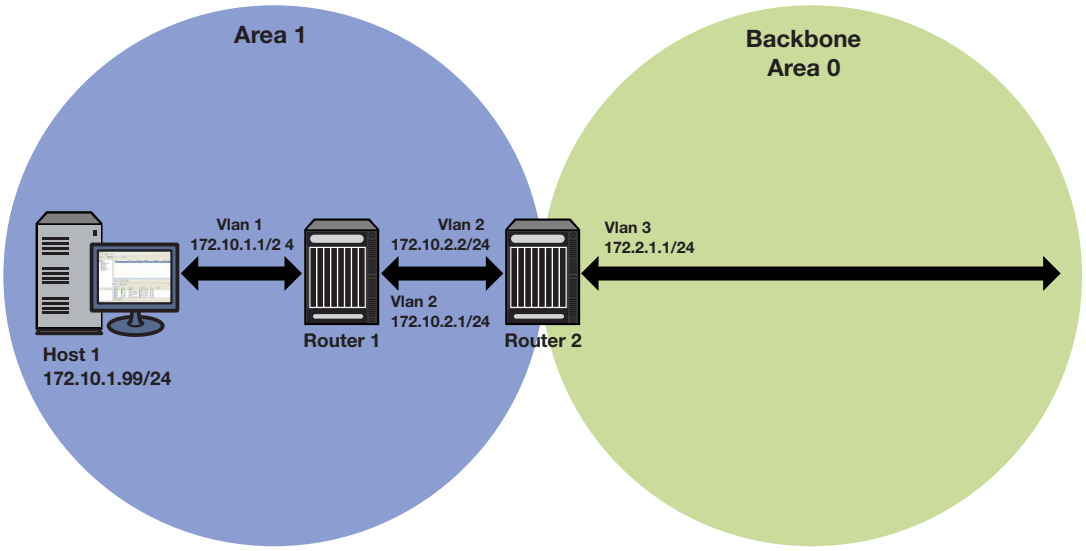
```
S Chassis(su)->configure
S Chassis(su-config)->interface vlan 1255
S Chassis(su-config-intf-vlan.0.1255)->ipv6 forwarding
S Chassis(su-config-intf-vlan.0.1255)->no shutdown
S Chassis(su-config-intf-vlan.0.1255)->show ipv6 interface vlan.0.1255 brief
Oper Status Legend: up = up; dn = down; tn = tentative dp = duplicate
                                                                    --Status--
Interface    IPv6 Address                                        Pfx Adm Op Fwd
------------ ---------------------------------------------------- --- --- -- ---
vlan.0.1255   fe80::21f:45ff:fe9a:6cb7%1255                        10 dn  dn yes

S Chassis(su-config-intf-vlan.0.1255)->ipv6 ospf 23 area 0.0.0.1
S Chassis(su-config-intf-vlan.0.1255)->
```

Basic OSPF Topology

Figure 42-1 provides an overview of a basic OSPFv3 topology. This topology displays two areas: a backbone area which must exist in any OSPF topology and a directly connected area 1. See “Configuring OSPFv3 Areas” on page 42-11 for a full discussion of OSPF area configuration. This

basic configuration requires the configuration of three interfaces and two routers on the OSPFv3 router instance. Because OSPFv3 uses link-local addresses for source and next-hop, an IPv6 address is not required for a minimal OSPF topology. The link-local address is autogenerated on the interface when IPv6 forwarding is enabled. For OSPFv3, the router ID is used as the neighbor ID. For all other OSPFv3 configuration, default values are used.

Figure 42-1 Basic OSPF Topology



1	VLAN 1, link-local address: fe80::20a:45af:fe9a:6ca5%1	**3**	VLAN 2, link-local address: fe80::45f:43ff:fa9b:3bb2%2
2	VLAN 2, link-local address: fe80::38c:53fd:fa8a:3bb2%2	**4**	VLAN 3, link-local address: fe80::a9b:26fa:fc4a:6fa1%3

Example

The following example configures the basic OSPF topology as displayed in Figure 42-1 on page 42-7:

Router 1 CLI Input

```
Router 1(rw)->configure
Router 1(rw-config)->interface vlan 1
Router 1(rw-config-intf-vlan.0.1)->ipv6 forwarding
Router 1(rw-config-intf-vlan.0.1)->ipv6 ospf 1 area 0.0.0.1
Router 1(rw-config-intf-vlan.0.1)->no shutdown
Router 1(rw-config-intf-vlan.0.1)->exit
Router 1(rw-config)->interface vlan 2
Router 1(rw-config-intf-vlan.0.1)->ipv6 forwarding
Router 1(rw-config-intf-vlan.0.1)->ipv6 ospf 1 area 0.0.0.1
Router 1(rw-config-intf-vlan.0.1)->no shutdown
Router 1(rw-config-intf-vlan.0.2)->exit
Router 1(rw-config)->ipv6 router ospf 1
Router 1(rw-101-config-ospfv3)->router-id 1.1.1.1
Router 1(rw-101-config-ospfv3)->exit
Router 1(rw-config)->
```

Router 2 CLI Input

```
Router 2(rw)->configure
Router 2(rw-config)->interface vlan 2
Router 2(rw-config-intf-vlan.0.2)->ipv6 forwarding
Router 2(rw-config-intf-vlan.0.2)->ipv6 ospf 1 area 0.0.0.1
Router 2(rw-config-intf-vlan.0.2)->exit
Router 2(rw-config)->interface vlan 3
Router 2(rw-config-intf-vlan.0.3)->ipv6 forwarding
Router 2(rw-config-intf-vlan.0.3)->ipv6 ospf 1 area 0.0.0.0
Router 2(rw-config-intf-vlan.0.3)->exit
Router 2(rw-config)->ipv6 router ospf 1
Router 2(rw-config-ospfv3)->router-id 2.2.2.2
Router 2(rw-config-ospfv3)->exit
Router 2(rw-config)->
```

Configuring the Router ID

In OSPFv3, all neighbors are identified by their router ID, not the interface address, as is the case for OSPFv2 broadcast and NBMA links. OSPFv3 initially assigns all routers a router ID based on the highest loopback IP address of the interfaces configured for IP routing. If there is no loopback interface configured then it will be the highest VLAN IP address configured. This unique value, which is included in the hello packet transmitted in Link State Advertisements (LSA), identifies one router to another and helps establish adjacencies among OSPFv3 routers. When you specify an interface as the router ID, this value supersedes the default ID.

Use the **router-id** command in OSPFv3 configuration mode to override the default router ID.

The following example sets the router ID to 1.1.1.1, overriding the default router ID value:

```
S-Series(rw)->configure
S-Series(rw-config)->interface loopback 1
S-Series(su-config-intf-loop.0.1)->ip address 2001:a123::1
S-Series(rw-config-intf-loop.0.1)->exit
S-Series(rw-config)->ipv6 router ospf 1
S-Series(rw-config-ospfv3)->router-id 1.1.1.1
S-Series(rw-config-ospfv3)->exit
S-Series(rw-config)->
```

Configuring the Designated Router

In the process of implementing OSPFv3, a large number of multi-access links to routers across the network may cause too many adjacencies to form. To avoid this problem, a Designated Router (DR) is elected per multi-access network to build adjacencies to all other routers on that network. A Backup Designated Router (BDR) is also elected in case the Designated Router (DR) fails, in which case the BDR will become the DR.

Note: A DR is required only for multi-access networks. Point-to-Point links do not need a DR because only a single adjacency is required.

To elect a DR from a host of candidates on the network, each router multicasts an hello packet and examines the priority of hello packets received from other routers. The router with the highest

priority is elected the DR, and the router with the next highest priority is elected the BDR. Any router with a priority of 0 will opt out of the DR election process. See the "Configuring Router Priority" on page 42-9 for details on configuring router priority. If DR candidates all share the same non-zero priorities, OSPF applies the router ID as a tie-breaker where the highest ID is chosen DR and the next highest ID is chosen BDR. If the priorities are not same, router ID values are not used and the highest priority determines the DR and BDR.

Configuring Router Priority

When two routers attached to a network both attempt to become the designated router, the one with the highest router priority takes precedence. A router whose router priority is set to 0 is ineligible to become the designated router on the attached network. Router priority is specified per router interface and is advertised in hello packets sent out by the interface.

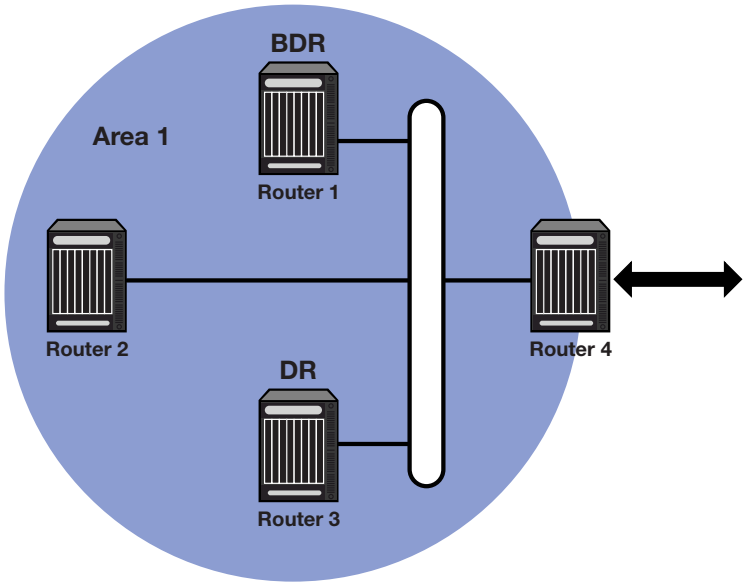
Use the **ipv6 ospf priority** command in interface configuration command mode to specify the router priority that will be included in LSAs going out this interface.

Figure 42-2 on page 42-9 displays a designated router topology example. The example will configure the four displayed routers with the following priorities:

- Router 1 = 25
- Router 2 = 10
- Router 3 = 30
- Router 4 = 0

Router 4 will not take part in the election process at all. Router 3 has the highest priority and therefore will be elected DR. Router 1 has the second highest priority and will be elected BDR.

Figure 42-2 OSPF Designated Router Topology



Example

The following example provides the input required to configure the designated router topology as displayed in Figure 42-2 on page 42-9:

Router 1

```
Router 1(rw)->configure
```

```
Router 1(rw-config)->interface vlan 1
Router 1(rw-config-intf-vlan.0.1)->ipv6 ospf priority 25
Router 1(rw-config-intf-vlan.0.1)->exit
Router 1(rw-config)->
```

Router 2

```
Router 2(rw)->configure
Router 2(rw-config)->interface vlan 1
Router 2(rw-config-intf-vlan.0.1)->ipv6 ospf priority 10
Router 2(rw-config-intf-vlan.0.1)->exit
Router 2(rw-config)->
```

Router 3

```
Router 3(rw)->configure
Router 3(rw-config)->interface vlan 1
Router 3(rw-config-intf-vlan.0.1)->ipv6 ospf priority 30
Router 3(rw-config-intf-vlan.0.1)->exit
Router 3(rw-config)->
```

Router 4

```
Router 4(rw)->configure
Router 4(rw-config)->interface vlan 1
Router 4(rw-config-intf-vlan.0.1)->ipv6 ospf priority 0
Router 4(rw-config-intf-vlan.0.1)->exit
Router 4(rw-config)->
```

Configuring the Administrative Distance for OSPF Routes

If several routes coming from different protocols are presented to the Route Table Manager (RTM), the protocol with the lowest administrative distance will be chosen for route installation. The S-Series platform supports connected, static, OSPF, and RIP routes.

The table below displays the default distance for these routing protocols.

Route Source	Default Distance
Connected	0
Static	1
BGP	20 - Routes external to the AS 200 - Routes internal to the AS
OSPF	110
RIP	120

Use the **distance ospf** command in OSPFv3 router configuration command mode to change the administrative distance assigned to the OSPFv3 protocol. This command provides for the configuration of separate values for OSPFv3 external and intra-area routes.

Configuring OSPFv3 Areas

OSPFv3 allows collections of contiguous networks and hosts to be grouped together. Such a group is called an area. Each area runs a separate copy of the basic link-state routing algorithm. This means that each area has its own link-state database.

The topology of an area is invisible from outside of the area, and routers internal to a given area know nothing of the detailed topology external to the area. This isolation of area detail enables the protocol to effect a marked reduction in routing traffic as compared to treating the entire Autonomous System as a single link-state domain. A router has a separate link-state database for each area it is connected to. Routers connected to multiple areas are called Area Border Routers (ABR). Two routers belonging to the same area have, for that area, identical area link-state databases.

An autonomous system can have one or more areas. A multiple area AS must define one of the areas as the backbone with an area ID of **0**. Area IDs are assigned when enabling OSPFv3 on the interface using the **ipv6 ospf** command (see "Configuring OSPFv3 on the Routing Interface" on page 42-6). All non-backbone areas in a multiple area AS must either be contiguous to the backbone or connected using a virtual-link. The backbone is responsible for distributing routing information between non-backbone areas. The backbone must be contiguous. However, it need not be physically contiguous; backbone connectivity can be established and maintained through the configuration of virtual-links.

Virtual-links can be configured between any two backbone routers that have an interface to a common non-backbone area. Such virtual-links belong to the backbone. The protocol treats two routers joined by a virtual-link as if they were connected by an unnumbered point-to-point backbone network.

Inter-area route calculation for OSPFv3 is similar to OSPFv2. See RFC 2328 for inter-area route calculation details. See RFC 2740 *OSPF for IPv6* for details on the differences between calculating inter-area routes between OSPFv2 and OSPFv3.

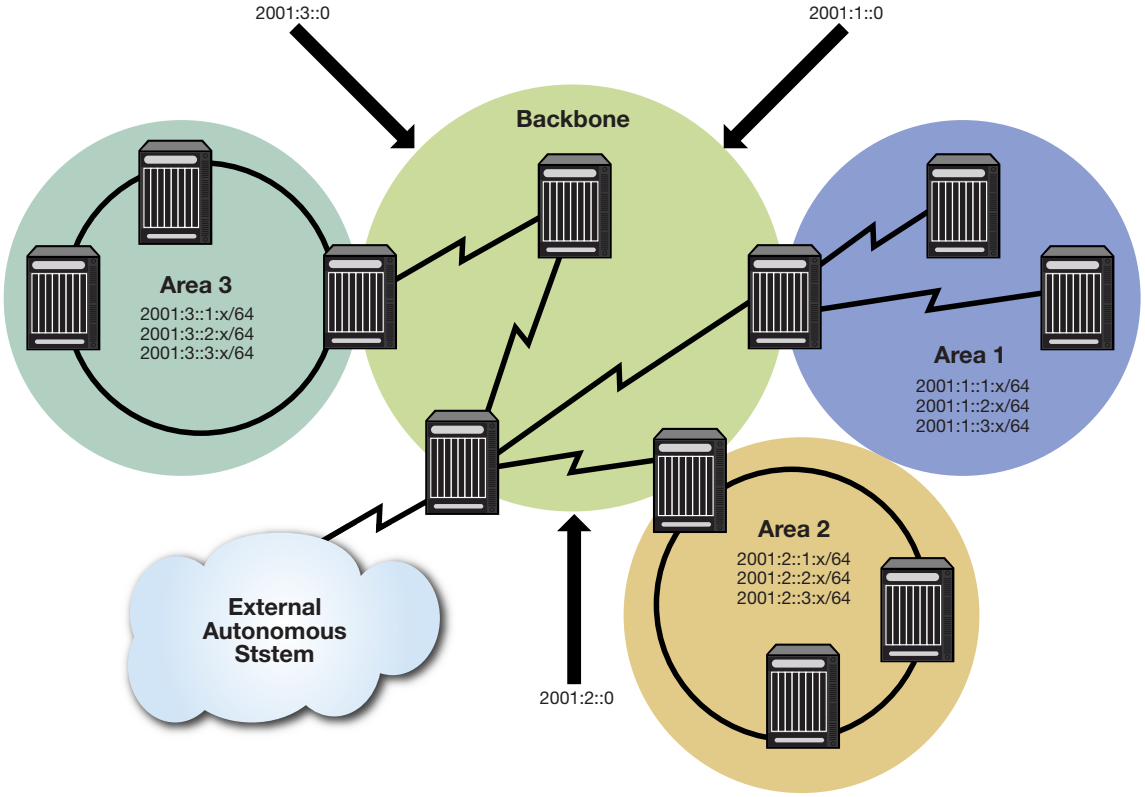
An Area ID can be any value from 0 - 4294967295, but is converted into the 32-bit dotted-quad format (area 50 would be displayed as 0.0.0.50; area 3546 would be displayed as 0.0.13.218)

Configuring Area Range

An area range is a form of address summarization that defines a range of addresses to be used by the backbone ABRs when they communicate routes to other areas. Area range is a critical tool that pares the route tables and update traffic, as well as reduces network recalculation by the Dijkstra algorithm. Area range configuration summarizes by aggregating an areas' internal addresses to advertise a single network. Backbone routers see only one update, representing an entire range of subnets. Area ranges can be configured for purposes of advertisement as well as summarization of addresses that should not be advertised.

Use the **area range** command in OSPFv3 configuration command mode to configure an area address summarization.

Figure 42-3 OSPF Summarization Topology



Example

The following example provides the input required to configure summarization of the three area topology as displayed in Figure 42-3 on page 42-12:

Area 1

```
ABR1(rw)->configure
ABR1(rw-config)->router ospf 1
ABR1(rw-config-ospf-1)->area 1 range 2001:1::0/64
ABR1(rw-config-ospf-1)->exit
ABR1(rw-config)->
```

Area 2

```
ABR2(rw)->configure
ABR2(rw-config)->router ospf 1
ABR2(rw-config-ospf-1)->area 2 range 2001:2::0/64
ABR2(rw-config-ospf-1)->area 2 range 2001:2:4::0/64 not-advertised
ABR2(rw-config-ospf-1)->exit
ABR2(rw-config)->
```

Area 3

```
ABR3(rw)->configure
ABR3(rw-config)->router ospf 1
ABR3(rw-config-ospf-1)->area 3 range 2001:3::0/64
```

```
ABR3(rw-config-ospf-1)->exit
ABR3(rw-config)->
```

Configuring a Stub Area

A stub area is a non-transit area. In other words, an area that does not originate or propagate external routes. AS-external-LSAs are not flooded into the stub area; routing to AS external networks is based on a single per-area default route. This reduces the link-state-database size and memory requirements for routers within stub areas.

Handy for reducing routing table size, a stub area is a “dead-end” in which there is no other way to enter or exit except through an Area Border Router (ABR). No ASE (Autonomous System External) or NSSA routes are permitted in a stub area. Each router in a stub area must specify that they are members of the stub area. When specifying that the ABR is a member of the stub area, the ABR will inject a default route into the area.

Routing to external designations from stub areas is based on a default route injected by a stub area’s ABR. A default route is automatically created by the stub area’s ABR. This default route is injected into the stub area to enable other stub routers within the stub area to reach any external routes that are no longer inserted into the stub area.

A stub area can be configured such that the ABR is prevented from sending type 3 summary LSAs into the stub area using the **no-summary** option. In this case, all destinations outside of the stub area are represented by means of a default route.

There are a couple of restrictions on the use of stub areas. Virtual-links cannot be configured through stub areas, and AS boundary routers cannot be placed internal to stub areas.

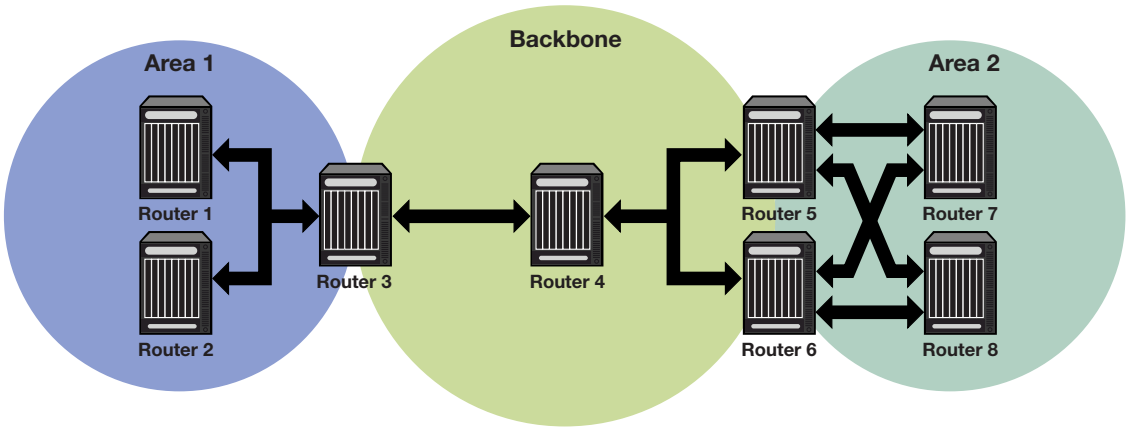
Use the **area stub** command in OSPF router configuration command mode to configure an area as a stub.

Stub Area Default Route Cost

A cost value can be set for the default route that is sent into a stub area by an ABR. Configuration of the stub area default route cost is restricted to the ABR attached to this stub area.

Use the **area default-cost** command in OSPFv3 router configuration command mode on the ABR attached to this stub area to configure the stub area default route cost.

Figure 42-4 OSPF Stub Area Topology



Example

Every router in Areas 1 and 2 are configured for a stub area (Routers 1, 2, and 3 for Area 1 and Routers 5, 6, 7, and 8 for Area 2). Additionally, ABR routers 3, 5, and 6 are also configured with a default-cost to be assigned to the stub area. Router 5 has a lower metric cost when compared to Router 6, so Router 5 will be the preferred router for packets to access the area, with Router 6 employed as a backup in case Router 5 fails. The following example provides the input required to configure the stub topology as displayed in Figure 42-4 on page 42-14:

Router 1

```
Router1(rw-config)->router ospf 1
Router1(rw-config-ospf-1)->area 1 stub
```

Router 2

```
Router2(rw-config)->router ospf 1
Router2(rw-config-ospf-1)->area 1 stub
```

Router 3

```
Router3(rw-config)->router ospf 1
Router3(rw-config-ospf-1)->area 1 stub
Router3(rw-config-ospf-1)->area 1 default-cost 15
```

Router 5

```
Router5(rw-config)->router ospf 1
Router5(rw-config-ospf-1)->area 2 stub
Router3(rw-config-ospf-1)->area 2 default-cost 15
```

Router 6

```
Router6(rw-config)->router ospf 1
Router6(rw-config-ospf-1)->area 2 stub
Router6(rw-config-ospf-1)->area 2 default-cost 20
```

Router 7

```
Router7(rw-config)->router ospf 1
Router7(rw-config-ospf-1)->area 2 stub
```

Router 8

```
Router8(rw-config)->router ospf 1
Router8(rw-config-ospf-1)->area 2 stub
```

Configuring a Not So Stubby Area (NSSA)

A Not So Stubby Area (NSSA) is a hybrid area using an Autonomous System Border Router (ASBR) to connect two disparate organizations. External routes are advertised as Type 7 LSAs and are converted to Type 5 LSAs before flooding to the backbone by the NSSA's ABR. Also, summary routes are allowed into the NSSA while external routes from other networks are still filtered from insertion into the NSSA.

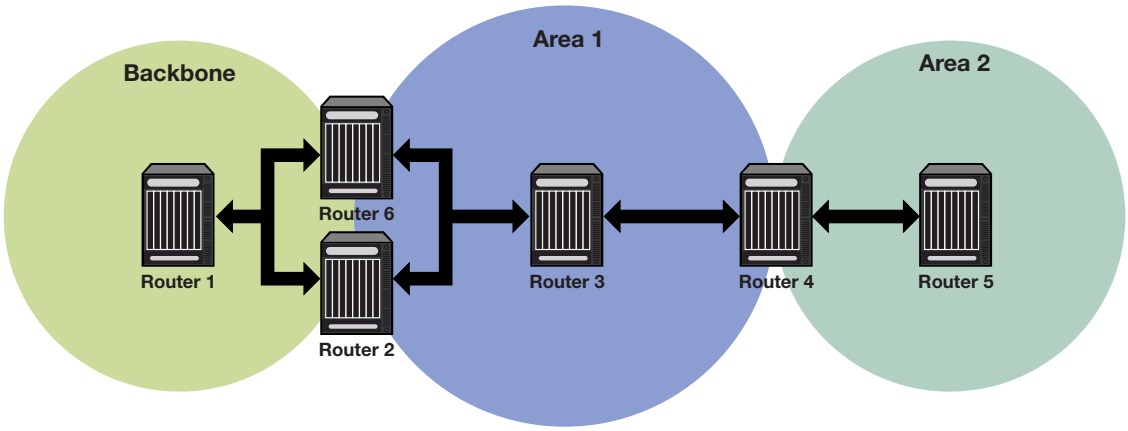
External routes that are not imported into an NSSA can be represented by a default route. If the router is an ABR and has the highest router ID of all ABRs in the area, and no other ABR in the area is configured to translate always, it will translate Type 7 LSAs into Type 5 LSAs. Configuring the identity of the translator can be used to bias the routing to aggregated destinations. When translator role is set to Always, Type-7 LSAs are always translated regardless of the translator state of other NSSA border routers.

When a translating ABR loses a translator election, it will stop translating, and after a number of seconds (set by the **transstabilityint** option), it will flush any Type 5 LSAs resulting from aggregation. Any Type 5 LSAs resulting from direct translation of Type 7 LSAs will be allowed to age out. An ABR will always originate a default route into any attached NSSAs.

If the **no-summary** option is specified, the ABR does not send type 3 summary LSAs into the NSSA area, therefore all destinations outside of the NSSA area are represented by means of a default route.

Use the **area nssa** command to configure an area as a Not-So-Stubby-Area.

Figure 42-5 OSPF NSSA Topology



Example

Routers 2 and 6 are configured as the ABRs between Area 1 and 0, and Router 4 as the ASBR. Router 2 is configured to set Area 1 as an NSSA, and Type 7 routes from the connected domain will be translated to Type 5 routes into the backbone.

ABR Router 2 will only translate Type 7 LSAs; static routes redistributed by router 4. Also, Router 2 will always translate, since it is configured to do so; Router 6 will not, since only one ABR will perform the translation for a given area.

Router 4 will be configured to redistribute static routes.

The following example provides the input required to configure the NSSA topology as displayed in Figure 42-5 on page 42-15:

Router 6 (ABR)

```
Router 6(rw)->configure
```

```
Router 6(rw-config)->interface vlan 1
Router 6(rw-config-intf-vlan.0.1)->ipv6 address 2001:0:1:::1:1/64
Router 6(rw-config-intf-vlan.0.1)->ipv6 forwarding
Router 6(rw-config-intf-vlan.0.1)->ipv6 ospf 1 area 0.0.0.0
Router 6(rw-config-intf-vlan.0.1)->no shutdown
Router 6(rw-config-intf-vlan.0.1)->exit
Router 6(rw-config)->interface vlan 2
Router 6(rw-config-intf-vlan.0.2)->ipv6 address 2001:1:1::1:1/64
Router 6(rw-config-intf-vlan.0.2)->ipv6 forwarding
Router 6(rw-config-intf-vlan.0.2)->ipv6 ospf 1 area 0.0.0.1
Router 6(rw-config-intf-vlan.0.2)->no shutdown
Router 6(rw-config-intf-vlan.0.2)->exit
Router 6(rw-config)->ipv6 router ospf 1
Router 6(rw-config-ospfv3)->router-id 6.6.6.6
Router 6(rw-config-ospfv3)->area 1 nssa
Router 6(rw-config-ospfv3)->exit
```

Router 2(ABR)

```
Router 2(rw)->configure
Router 2(rw-config)->interface vlan 1
Router 2(rw-config-intf-vlan.0.1)->ipv6 address 2001:0:1::2:1/64
Router 2(rw-config-intf-vlan.0.1)->ipv6 forwarding
Router 2(rw-config-intf-vlan.0.1)->ipv6 ospf 1 area 0.0.0.0
Router 2(rw-config-intf-vlan.0.1)->no shutdown
Router 2(rw-config-intf-vlan.0.1)->exit
Router 2(rw-config)->interface vlan 2
Router 2(rw-config-intf-vlan.0.2)->ipv6 address 2001:1:1::2:1/64
Router 2(rw-config-intf-vlan.0.2)->ipv6 forwarding
Router 2(rw-config-intf-vlan.0.2)->ipv6 ospf 1 area 0.0.0.1
Router 2(rw-config-intf-vlan.0.2)->no shutdown
Router 2(rw-config-intf-vlan.0.2)->exit
Router 2(rw-config)->ipv6 router ospf 1
Router 2(rw-config-ospfv3)->router-id 2.2.2.2
Router 2(rw-config-ospfv3)->area 1 nssa
Router 2(rw-config-ospfv3)->area 1 nssa transrole always
Router 2(rw-config-ospfv3)->area 1 nssa-range 2002:1:1::1/64
Router 2(rw-config-ospfv3)->exit
```

Router 4 (ASBR)

```
Router 4(rw)->configure
Router 4(rw-config)->interface vlan 2
Router 4(rw-config-intf-vlan.0.2)->ipv6 address 2001:1:1::2:2/64
Router 4(rw-config-intf-vlan.0.2)->ipv6 forwarding
Router 4(rw-config-intf-vlan.0.2)->ipv6 ospf 1 area 0.0.0.1
Router 4(rw-config-intf-vlan.0.1)->no shutdown
Router 4(rw-config-intf-vlan.0.1)->exit
```

```
Router 4(rw-config)->interface vlan 3
Router 4(rw-config-intf-vlan.0.3)->ipv6 address 2001:3:1::1:1/64
Router 4(rw-config-intf-vlan.0.3)->ipv6 forwarding
Router 4(rw-config-intf-vlan.0.3)->ipv6 ospf 1 area 0.0.0.1
Router 4(rw-config-intf-vlan.0.3)->no shutdown
Router 4(rw-config-intf-vlan.0.3)->exit
Router 4(rw-config)->ipv6 router ospf 1
Router 4(rw-config-ospfv3)->router-id 4.4.4.4
Router 4(rw-config-ospfv3)->redistribute static metric-type 1
Router 4(rw-config-ospfv3)->exit
```

Configuring Area Virtual-Links

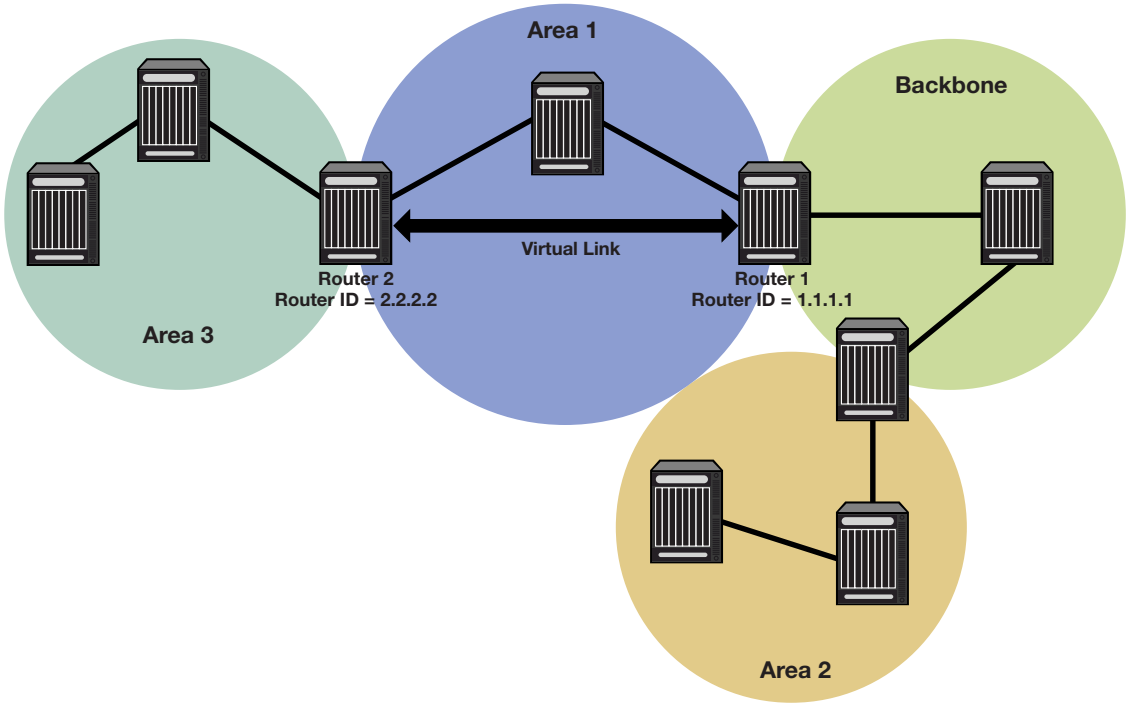
The backbone area 0 cannot be disconnected from any other areas in the AS. Disconnected areas will become unreachable. To establish and maintain backbone connectivity, virtual-links can be configured through non-backbone areas for the purpose of connecting a disconnected area with the backbone through a backbone connected area. The two endpoints of a virtual-link are ABRs, both of which belong to the backbone connected area (also referred to as the transit area); one of which belongs to the area disconnected from the backbone. Virtual-links cannot be configured through stub areas (see “Configuring a Stub Area” on page 42-13 for stub area configuration information).

The virtual-link is treated as if it were an unnumbered point-to-point network belonging to the backbone and joining the two ABRs. The cost of a virtual-link is not configured. It is auto configured with the cost of the intra-area path between the two ABRs that make up the virtual-link.

Use the **area virtual-link** command in OSPF router configuration command mode, providing the transit area ID and the ABRs IP address, to configure an area virtual-link.

Figure 42-6 on page 42-18 displays a typical virtual-link topology. Area 3 does not share an ABR with the backbone area, and is therefore disconnected from the backbone. Area 3 shares an ABR (router 2) with area 1. Area 1 has a second ABR (router 1) that it shares with the backbone. Area 1 is the transit area because it contains an ABR that it shares with the disconnected area and a second ABR that it shares with the backbone. By configuring an area virtual-link between router 2 and router 1, Area 3 will gain connectivity with the backbone and be able to learn routes for this AS.

Figure 42-6 Virtual-Link Topology



Example

The following example presents the configuration required to configure the virtual-link displayed in Figure 42-6 on page 42-18:

Router 1

```
Router 1(rw-config)->router ospf 1
Router 1(rw-config-ospf-1)->area 0.0.0.1 virtual-link 2.2.2.2
Router 1(rw-config-ospf-1)->exit
Router 1(rw-config)->
```

Router 2

```
Router 2(rw-config)->router ospf 2
Router 2(rw-config-ospf-2)->area 0.0.0.1 virtual-link 1.1.1.1
Router 2(rw-config-ospf-2)->exit
Router 2(rw-config)->
```

Configuring Area Virtual-Link Timers

The following timers can be configured for an area virtual-link:

- Dead-interval using the **area virtual-link dead-interval** command
- Hello-interval using the **area virtual-link hello-interval** command
- Retransmit-interval using the **area virtual-link retransmit-interval** command
- Transmit-delay using the **area virtual-link transmit-delay** command

See "Configuring OSPFv3 Timers" on page 42-23 for an OSPF timers discussion.

Note: RFC 2328 specifies that the retransmit-interval should be greater than the expected round-trip delay between the two routers. This may be hard to estimate for a virtual-link; it is better to err on the side of making it too large.

Configuring IPsec Authentication for OSPFv3

Internet Protocol Security (IPsec) is an internet layer, end-to-end security scheme that provides for the securing of IP communications by authentication and encrypted authentication of each communication session IP packet. IPsec for OSPFv3 is configured on the interface. The IPsec for OSPFv3 implementation supports both authentication only and encrypted authentication. IPsec must first be enabled on the interface. Supported IPsec authentication algorithms are:

- **MD5** – Message-Digest algorithm 5
- **SHA1** – Secure Hash Algorithm 1
- **AESCBC** – Advanced Encryption Standard (AES) Cipher Algorithm in Cipher Block Chaining (CBC)

Supported IPsec encryption ciphers are:

- **3DES** – Triple Data Encryption Standard cipher algorithm
- **AESCBC** – AES (Cipher Block Chaining) cipher algorithm

Each IPsec configuration must have a Security Parameters Index (SPI) with a value between **256** - **4294967295** assigned to it and a security key. The key can be specified as a hex key.

IPsec must be enabled in global VRF router configuration mode using the **crypto ipsec enable** command before using IPsec for OSPFv3 authentication or encrypted authentication.

Configure IPsec for OSPFv3 on an interface for authentication only by specifying the SPI and authentication algorithm using the **ipv6 ospf authentication** command in interface configuration mode.

This example shows how to configure VLAN 1 for IPsec SPI entry 256 for MD5 authentication with a hex key of **1234567890abcdef**:

```
S Chassis(rw-config)->crypto ipsec enable
S Chassis(rw-config)->interface vlan 1
S Chassis(rw-config-intf-vlan.0.1)->ipv6 ospf authentication spi 256 md5
1234567890abcdef hex
```

Configure IPsec for OSPFv3 on an interface for encrypted authentication by specifying the SPI, authentication algorithm and encryption cipher using the **ipv6 ospf encryption** command in interface configuration mode.

This example shows how to configure VLAN 1 for IPsec SPI entry 256 for the 128-bit aescbc encryption with a key of **1234567890abcdef**, and for MD5 authentication with a hex key of **1234567890abcdef**:

```
S Chassis(rw-config)->crypto ipsec enable
S Chassis(rw-config)->interface vlan 1
S Chassis(rw-config-intf-vlan.0.1)->ipv6 ospf encryption ipsec spi 256 esp aescbc
128 1234567890abcedf hex auth md5 1234567890abcdef hex
```

Configuring Route Redistribution

Redistribution permits the importation of other routing protocols into OSPF such as RIP, as well as static and directly connected routes. Alternately, you can specify a route-map for redistribution into OSPF. Be aware that if the referenced route map has not yet been configured, then an empty

route map is created with the specified name. See "Configuring a Not So Stubby Area (NSSA)" on page 42-15 for an example of redistribution of static routes by an ASBR in an NSSA context.

Use the **redistribute** command in OSPF router configuration command mode to permit the redistributions of OSPF, RIP, static, or connected routes by this router.

Filtering Routes from the OSPF Route Table

Routes can be filtered from the OSPF route table by creating an OSPF filter route route-map and assigning it to the distribute-list for this OSPFv3 router.

Use the **route-map filter** command as described in the "Route-Map Manager" section of the *Extreme Networks S-Series CLI Reference* to create an OSPF filter route route-map.

Use the **distribute-list route-map in** command to assign the filter route-map to the OSPF distribute-list.

Configuring Passive Interfaces

Passive interfaces explicitly allow the network to be advertised, but the router prevents the forming of neighbor relationships on that interface. Passive interfaces are included in the OSPF route table. Passive interfaces do not send or receive hello packets. OSPF adjacencies can not be formed on a passive interface.

An option exists to default all interfaces to passive mode.

Use the **passive-interface** command in router configuration command mode to configure an interface as passive.

Graceful Restart

OSPF graceful restart, sometimes referred to as non-stop forwarding, provides for an OSPF router to remain on the forwarding path during a restart of its OSPF software. Graceful restart has three elements to its configuration: enabling, helper router, and restart interval.

Enabling graceful restart instructs the firmware to perform a graceful restart, rather than a standard OSPF restart. Restart is only initiated by a fail-over. Grace LSAs are sent when OSPF is restarted on another module. Whether the failover is intentional or not, the failed router protocol is restarted on another module, and upon startup, OSPF sends grace LSAs out to its neighbors using existing link aggregation groups. Use the **graceful-restart enable** command to enable the graceful restart ability on this router.

The helper relationship with the restarting router is on a per network segment basis. The helper monitors the network for topology changes. If no changes occur, the helper router continues to advertise its LSAs as though no restart was occurring. If the restarting router was the designated router, the helper continues to treat it as such. If a topology change does occur, graceful restart is terminated on the restarting router and a standard restart occurs. Helper mode is enabled by default. Helper mode can be disabled on a restarting router neighbor using the **ipv6 ospf helper-disable** command in interface command mode. If the restarting router receives an LSA indicating a disabled helper, the graceful restart terminates and a standard restart occurs.

A restart interval provides for a maximum time in seconds after which the graceful restart will terminate should it not complete or terminate for other reasons within the interval. Use the **graceful-restart restart-interval** command to change the restart interval setting.

View the router OSPF section of the **show running-config** display to verify any non-default graceful restart settings.

Graceful Restart and High Availability

S-Series modules support single router high availability failover using the following components:

- OSPF graceful restart
- Non-stop router frame forwarding on each module
- Single router configuration
- Router protocol process failover to another module
- Link Aggregate Group (LAG) connectivity to neighboring routers

In a stable network, the route and rule information is fairly constant. If the router protocol process was to suddenly fail, forwarding information current at the time of the failure in all probability is usable for the short time after the failure until recovery occurs. During this recovery period, existing connections (that were not directly using the failed module) remain in effect. New connections continue to be installed using the last known “good” forwarding information. The router protocol process that failed is dynamically restarted. You do not configure where the router process is running. The router forwarding process remains active on every module. The protocol process exchanges protocol and maintains state that it distributes to the other modules and does not have to run on any specific module. One exception to this rule is that the module must have 256M of memory to be router protocol process eligible.

Upon failure of a module running the router protocol process, the protocol process is started on a recovery module. One of the first messages it sends to its OSPF neighbors is a grace LSA. High availability failover will successfully occur if the following is true:

- The router is enabled for graceful restart
- The neighbors are enabled to participate as graceful restart helpers
- The OSPF dead interval is configured for a sufficient period such that the grace LSA is received by its neighbors before the configured OSPF dead interval expires
- And each neighbor is a member of a LAG common to the failed router, allowing the neighbor to remain up

Figure 42-7 Physical and Logical Single Router HA Failover Configuration

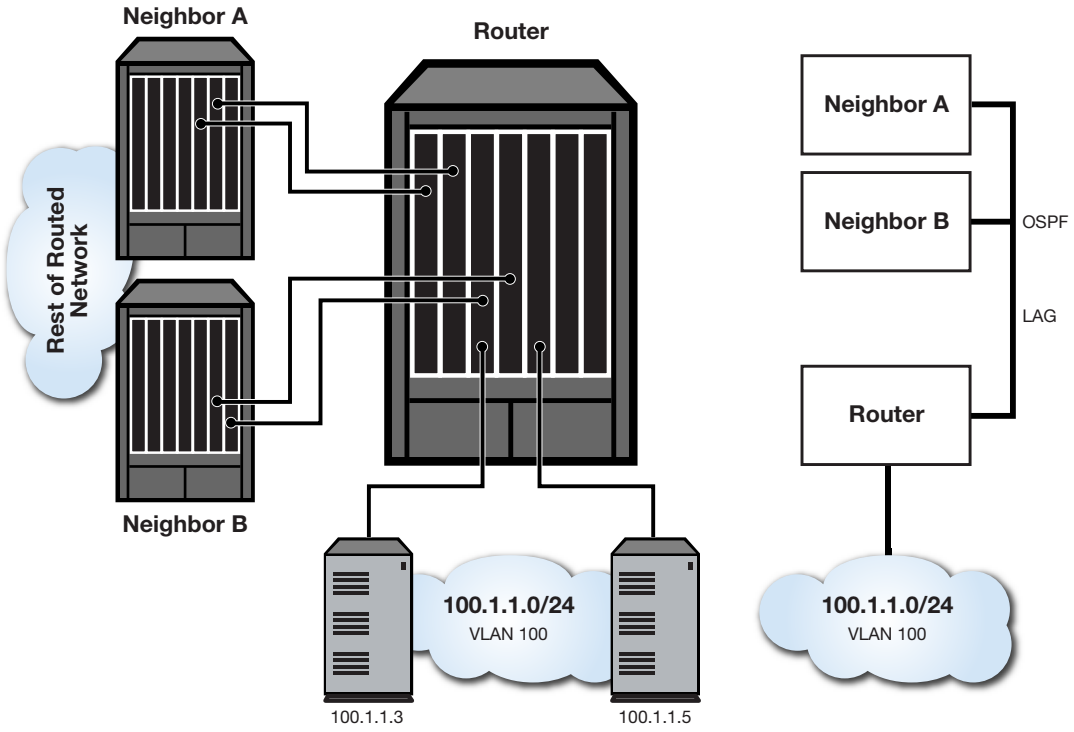


Figure 42-7 depicts the physical and logical configurations of the single router high availability failover mechanism. The neighbor to router lines display direct neighbor connections to the router enabled for OSPF graceful restart and members of LAGs common to the failing router. See Chapter 25, **Link Aggregation Control Protocol (LACP) Configuration** for LAG configuration details. The server 100.1.1.3 and 100.1.1.5 to router lines display VLAN connections common to both the failing and recovery routers. Helper mode on each neighbor is enabled by default. Enable graceful restart on the router using the **graceful-restart enable** command.

```
S Chassis(rw-config)->router ospf 1
S Chassis(rw-config-ospf-1)->graceful-restart enable
S Chassis(rw-config-ospf-1)->
```

Configuring Interface Cost

Each interface has an outbound cost associated with it. The lower the cost, the more likely the interface will be used to forward data traffic. Should several equal-cost routes to a destination exist, traffic is distributed equally among them.

The formula for calculating the OSPF interface cost metric is the reference bandwidth divided by the interface bandwidth. By default the reference bandwidth is set to 100 Mbps. For 10 Mbps links, the resulting cost is 10. For 100, 1000, or 10000 Mbps links, the resulting cost is 1. The reference bandwidth can be modified using the **auto-cost reference-bandwidth** command in OSPF configuration mode. The ability to re-center the reference bandwidth to a higher value, allows for OSPF interface costs to default to a value greater than 1 for 100, 1000, or 10000 Mbps links and greater than 10 for 10 Mbps links.

It is recommended that the auto cost reference bandwidth be the same value for all OSPF routers in the domain.

Use the **ipv6 ospf cost** command in interface configuration command mode to statically specify the outbound cost of this interface. A statically configured OSPF interface cost overrides all other interface cost methods.

For logical interfaces containing multiple physical interfaces, such as a LAG, the aggregate interface speed is not readily available. A tracked object configured with the ports belonging to the logical interface can return the physical interface speed of each physical port specified in the tracked object. OSPF will sum the returned interface speeds and use that aggregate value when calculating OSPF interface cost. Because the tracked object will report when a physical interface is up or down, OSPF will dynamically adjust the aggregate speed when an interface becomes active or goes down and adjust the OSPF interface cost accordingly. This method should be used in LAG and ECMP logical interface contexts.

Note: The speed used in the cost calculation is sum of all ports capabilities in the tracked object. Setting the speed manually will not change the tracked interface speed. A 1GB capable port has a 1 GB speed regardless of the manual speed setting. The same holds true for ports that auto-negotiate to a lower speed. The expectation is that both sides of the link are using the same ports and SFP connectors and should result in the same speed.

Use the **ipv6 ospf cost track** command in interface configuration mode to calculate the OSPF interface cost based upon summing physical interface speeds that belong to a logical interface.

When adding an additional physical port to a logical interface that uses the interface summation method to determine OSPF interface cost, you must also add the physical port to the associated tracked object.

See "Tracked Object Manager Configuration" on page 13-1 for tracked object configuration details.

Configuring Bidirectional Forwarding Detection (BFD) on Interfaces

BFD is used to detect a communications failure with an OSPF forwarding plane next-hop. BFD detects failures in under one second. BFD augments the OSPF Hello mechanism. The OSPF Hello interval defaults to 10 seconds. With high speed data rates, a failure requiring multiple seconds to detect can result in significant data loss. The OSPF implementation of the BFD protocol uses the following non-configurable parameters:

Transmit Interval – The period of time between the transmission of BFD control packets, set for 100ms.

Receive Interval – The period of time between received BFD control packets, set for 100ms.

Detection Multiplier – The Number of consecutive control packets that can be missed before the BFD session transitions to down, set to 3.

Use the **bfd all-intfs-on** command in OSPF router configuration mode to enable BFD on all OSPF interfaces.

Configuring OSPFv3 Timers

There are five OSPF timers:

- Hello-Interval
- Dead-Interval
- Retransmit-Interval
- Transmit-Delay
- SPF-Delay

To ensure efficient adjacency between OSPF neighbors, the S-Series device provides hello-interval and dead-interval commands. The hello interval is the period between transmissions of hello packet advertisements. The dead interval is the period that can elapse without receiving a router's hello packets before its neighbors will declare it down.

Use the **ipv6 ospf hello-interval** command in interface configuration command mode to configure the period between transmissions of hello packet advertisements.

Use the **ipv6 ospf dead-interval** in interface configuration command mode to configure the period between receiving hello packets before the associated neighbor is declared down.

In order to ensure that flooding is reliable, LSAs are retransmitted until they are acknowledged. The period between retransmissions is the retransmit-interval. If this interval is set too low for an interface, needless retransmissions will take place. If the value is set too high, the speed of the flooding, during the period of lost packets, may be affected.

Use the **ipv6 ospf retransmit-interval** command in interface configuration command mode to configure the retransmit-interval.

The transmit-delay is an estimation of the number of seconds it takes to transmit a link state update packet over this interface. This value should take into account transmission and propagation delays.

Use the **ipv6 ospf transmit-delay** command in interface configuration command mode to configure the transmit-delay.

The SPF-delay is the amount of time that transpires between the receipt of an OSPF update and the SPF calculation.

Use the **timers spf** command in OSPFv3 router configuration command mode to specify the amount of time between receiving an OSPF update and an SPF calculation occurring.

The OSPF timers can also be configured for an area virtual-link. See "Configuring Area Virtual-Links" on page 42-17.

Configuring the PE-CE Protocol

The PE-CE protocol allows a service provider offering Virtual Private Network (VPN) services to their customers to peer Customer Edge (CE) routers with their Provider Edge (PE) routers . RFC 6565 defines how the PE-CE protocol is implemented using the OSPF routing protocol.

When the PE router becomes a routing peer of the CE router, the PE router learns the routes that lead to the CE's site and can redistribute those routes to other PE routers that attach to the same VPN.

Use the **enable-pe-ce** command in OSPFv3 configuration mode to enable the PE-CE protocol on the router.

The OSPF VRF Domain ID

If the OSPF instances of an OSPF domain are given one or more domain IDs, OSPF can determine whether an OSPF-originated VPN-IPv6 route belongs to the same domain as a given OSPF instance and whether the route should be redistributed to that OSPF instance as an inter-area route or as an OSPF AS-external route.

If two OSPF instances with a domain ID configured are in the same OSPF domain, the PE-CE protocol requires that the primary domain ID of the second instance must be one of the domain IDs of the first instance (either primary or secondary). If two OSPF instances with a domain ID configured are not in the same OSPF domain, the primary domain ID of each instance must not be configured as a domain ID of the other OSPF instance.

The PE-CE protocol must be enabled using the **enable-pe-ce** command to set the OSPF VRF domain ID.

Use the domain-id command in OSPFv3 configuration mode to set the domain ID for this PE router.

Redistribute BGP into OSPF

OSPF supports the redistribution of BGP discovered routes into OSPF. A **global** option is available for the redistribution of BGP prefixes from the global router. When using the **global** option, VPN-IPv6 address prefixes are appropriately translated.

Use the **redistribute bgp** command in OSPFv3 configuration mode to redistribute BGP learned routes to other PE routers in this VPN.

OSPF Sham Link

If a VRF contains both an OSPF-distributed route and a VPN-IPv6 route for the same IPv6 prefix, then the backdoor OSPF-distributed route is preferred over the VPN backbone route, unless the next hop interface for an installed (OSPF distributed) route is the sham link, in which case, the VPN backbone VPN-IPv6 route is used.

If it is desired to have OSPF prefer the routes through the VPN backbone over the routes through the OSPF backdoor link, then the routes through the backbone must appear to be intra-area routes. The sham link provides this appearance of an intra-area link connecting the two PE routers.

Use the **area sham-link** command in OSPFv3 configuration mode, specifying both the source and destination link addresses to configure an OSPF sham link between two VPN PE routers.

OSPFv3 Configuration Details

This section provides details for the configuration of OSPFv3 on S-Series platforms.

Default Settings

Table 42-2 lists OSPF parameters and their default values.

Table 42-2 Default OSPF Parameters

Parameter	Description	Default Value
router ID	Provides for the identification of one router to another and helps establish adjacencies among OSPF routers.	highest loopback IP address or highest configured VLAN IP address if no loopback configuration exists
interface cost	An outbound interface value used in determining which routing interface should forward when more than one routing interface is available.	10
interface priority	A value placed on the interface that helps in determining which router will be elected designated router.	1
interface network type	Specifies the type of network an interface is connecting to.	broadcast

Table 42-2 Default OSPF Parameters (continued)

Parameter	Description	Default Value
LSA Thresholds	Specifies:	
	The number of LSA updates that force a full routing calculation	• 4294967295 Update starts
	The number of LSA updates that interrupt and restart a full routing calculation	• 4294967295 Update restarts
	The number of LSA inter-area/external updates that force a full routing calculation	• 50 Inter-area/external updates
	The number of intra updates that force a full routing calculation	• 0 Intra updates
LSA Pause Frequency	Specifies the number of cpu units SPF calculation runs before pausing.	10000
SPF delay timer	Specifies the amount of time between receiving an OSPF update and the start of an SPF calculation.	5 seconds
retransmit interval	A timer that determines the retransmission of LSAs in order to ensure reliable flooding.	5 seconds
transmit delay	Specifies the number of seconds it takes to transmit a link state update packet over this interface.	1 second
hello interval	The period between transmissions of hello packet advertisements.	• 10 seconds for broadcast and point-to-point networks • 30 seconds for non-broadcast and point-to-multipoint networks
dead interval	The period that can elapse without receiving a router's hello packets before its neighbors will declare it down.	40 seconds
distance	Specifies the administrative distance for OSPF routes. The available protocol with the lowest administrative distance is chosen for this route.	connected: 0 static: 1 BGP: • 20 - Routes external to the AS • 200 - Routes internal to the AS OSPF = 110 RIP = 120
graceful restart	Provides for an OSPF router to remain on the forwarding path during a restart of its OSPF software.	disabled
graceful restart restart interval	Specifies the maximum time in seconds after which the graceful restart will terminate should it not complete or terminate for other reasons within the interval.	120 seconds

Table 42-2 Default OSPF Parameters (continued)

Parameter	Description	Default Value
PE-CE Protocol	Enables the Customer Edge (CE) router as a Provider Edge (PE) router peers.	disabled

Procedure 42-1 describes how to configure basic OSPF parameters.

Procedure 42-1 Configuring Basic OSPFv3 Parameters

Step	Task	Command(s)
1.	Create an OSPF routing instance in router configuration mode.	**ipv6 router ospf** *process-id*
2.	In interface configuration mode, enable OSPFv3 on each routing interface, specifying the OSPFv3 process, area, and optional instance	**ipv6 ospf** *process* **area** *area* [**instance** *instance-id*]
3.	Optionally, in interface configuration mode, configure an IPv6 address for all routing interfaces in the AS.	**ipv6 address** {*link-local-address* **link-local** | *ipv6-address/length* | *ipv6-prefix/length* **eui-64** | **autoconfig** | *general-prefix sub-bits/length*}

Table 42-3 describes how to configure basic OSPFv3 parameters.

Table 42-3 Configuring OSPFv3 General Optional Parameters

Task	Command(s)
Optionally, change the OSPFv3 router ID for this device.	**router-id** *router-id*
Optionally, enable the OSPF PE-CE protocol.	**enable-pe-ce**
Optionally, configure the OSPF VRF domain ID.	**domain-id** [**secondary**] **type** *type* **value** *value*
Optionally, change the SPF LSA thresholds for this router.	**spf lsa-thresholds** *num-start num-restart num-intra-full num-ia-ext-full*
Optionally, change the SPF pause frequency to specify the number of units SPF calculation runs before pausing.	**spf pause-frequency** *units*
Optionally, change the delay, in milliseconds, between the receipt of an update and the beginning of the SPF execution.	**timers spf** *spf-delay*
Optionally, enable BFD on all OSPF interfaces.	**bfd all-intfs-on**
Optionally, change the administrative distance for OSPFv3 routes.	**distance** [**ospf** {**external** | **intra-area**}] *weight*
Optionally, define the range of addresses used by this Area Border Router (ABR) when communicating routes to other areas.	**area** *area-id* **range** *ipv6-address* [**not-advertise**]
Optionally, configure an area as a stub area.	**area** *area-id* **stub** [**no-summary**]
Optionally, set the cost for the default route that is sent into a stub area by an ABR.	**area** *area-id* **default-cost** *cost*
Optionally, configure an area as a not so stubby area.	**area** {*area-id* | *A.B.C.D*} **nssa** [**no-summary**] [**transstabilityint** *seconds*] [**transrole always**]

Table 42-3 Configuring OSPFv3 General Optional Parameters (continued)

Task	Command(s)
Optionally, configure an Autonomous System Border Router (ASBR) to summarize Type 7 to Type 5 routes matching the specified address and mask.	**area** {*area-id* | *A.B.C.D*} **nssa-range** *ipv6-address* [**not-advertise**]
Optionally, configure an OSPF sham link between two PE routers.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address*
Optionally, configure an OSPF sham link authentication key password.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **authentication-key** *password*
Optionally, modify the OSPF sham link dead interval.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **dead-interval** *seconds*
Optionally, modify the OSPF sham link hello interval.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **hello-interval** *seconds*
Optionally, configure the OSPF sham link keychain.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **keychain** *name*
Optionally, specify an OSPF sham link message digest key and MD5 authentication key.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **message-digest-key** *digest-key* **md5** *auth-key*
Optionally, modify the OSPF sham link retransmit interval.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **retransmit-interval** *seconds*
Optionally, modify the OSPF sham link transmit delay period.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **transmit-delay** *seconds*
Optionally, modify the OSPF sham link cost.	**area** *area-id* **sham-link** *source-ip-address destination-ip-address* **cost** *cost*
Optionally, configure an OSPF virtual-link, which represents a logical connection between the backbone and a non-backbone connected OSPF area.	**area** *area-id* **virtual-link** *router-id* **area** *area-id* **virtual-link** *router-id* **dead-interval** *seconds* **area** *area-id* **virtual-link** *router-id* **hello-interval** *seconds* **area** *area-id* **virtual-link** *router-id* **retransmit-interval** *seconds* **area** *area-id* **virtual-link** *router-id* **transmit-delay** *seconds*
Optionally, change the bandwidth reference setting used for calculating interface cost for this OSPF instance.	**auto-cost reference-bandwidth** *bandwidth-multiplier*
Optionally, enable passive OSPF on the specified interface.	**passive-interface** {**vlan** *vlan-id* | *interface-name* | **default**}
Optionally, allow routing information discovered through non-OSPF protocols to be distributed in OSPF update messages.	**redistribute** {**bgp** [**global**] | **connected** | **rip** | **static** | **blackhole**} [**route-map** *name*] [**metric** *metric-value*] [**metric-type** *type-value*] [**tag** *tag*]
Optionally, assign an OSPF route filter route-map to the OSPF distribute-list.	**distribute-list route-map** *name* **in**
Optionally, enable adjacency logging on this OSPFv3 router.	**log-adjacency**

Table 42-3 Configuring OSPFv3 General Optional Parameters (continued)

Task	Command(s)
Optionally, enable the graceful restart feature on this router.	**graceful-restart enable**
Optionally, change the graceful restart restart interval for this router.	**graceful-restart restart-interval** *interval*
Optionally, in system command mode, reset the specified OSPFv3 process ID or the OSPFv3 process.	**clear ipv6 ospf process** [*process-id*]
Optionally, in global configuration command mode, enable OSPFv3 protocol debugging output for the specified subsystem.	**debug ipv6 ospf** {*subsystem*}

Table 42-4 describes how to configure optional OSPF interface parameters. All commands in this procedure are entered in interface configuration command mode.

Table 42-4 Configuring OSPF Optional Interface Parameters

Task	Command(s)
Optionally, configure the OSPFv3 router neighbors for this router.	**ipv6 ospf neighbor** *ipv6-address*
Optionally, assign a cost for the specified neighbor on the interface.	**ipv6 ospf neighbor** *ipv6-address* **cost** *number*
Optionally, set the OSPFv3 priority value for the specified neighbor on the interface.	**ipv6 ospf neighbor** *ipv6-address* **priority** *number*
Optionally, set a non-broadcast neighbor polling interval.	**ipv6 ospf neighbor** *ipv6-address* **poll-interval** *seconds*
Optionally, filter outgoing link-state advertisements to an OSPFv3 neighbor on this interface.	**ipv6 ospf neighbor** *ipv6-address* **database-filter-all-out**
Optionally, set the OSPFv3 priority value for the interface.	**ipv6 ospf priority** *number*
Optionally, change the cost of sending an OSPF packet on this router interface. This setting overrides all other interface cost methods.	**ipv6 ospf cost** *cost*
Optionally, sum the interface speeds contained in the specified tracked object when calculating the OSPF interface cost.	**ipv6 ospf cost track** *trackobject-name*
Optionally, set the amount of time between retransmissions of link state advertisements (LSAs) for adjacencies that belong to the interface.	**ipv6 ospf retransmit-interval** *seconds*
Optionally, set the amount of delay before transmitting a link state update packet on an interface.	**ipv6 ospf transmit-delay** *seconds*
Optionally, enable the ignore MTU advertisement feature for the neighbor of this interface.	**ipv6 ospf ignore-mtu**
Optionally, change the number of seconds this router must wait before sending a hello packet to neighbor routers on this interface.	**ipv6 ospf hello-interval** *seconds*
Optionally, change the number of seconds this router must wait to receive a hello packet from its neighbor before determining that the neighbor is out of service.	**ipv6 ospf dead-interval** {*seconds* | **minimal** [**hello-multiplier** *number*]}

Table 42-4 Configuring OSPF Optional Interface Parameters (continued)

Task	Command(s)
Optionally, configure IPsec authentication on an interface.	**ipv6 ospf authentication spi** *spi* {**md5** *key* | **sha1** *key* | **aescbc** *key*} [hex]
Optionally, configure IPsec encrypted authentication on an interface.	**ipv6 ospf encryption ipsec spi** *spi* {**none** | **3des** *key* | **aescbc** {**128** | **192** | **256**} *key*} [hex] **auth** {**md5** *key* | **sha1** *key* | **aescbc** *key* | **no-auth**}
Optionally, disable the graceful restart helper feature on this interface.	**ipv6 ospf helper-disable**
Optionally, specify the network type that this interface is connected to.	**ipv6 ospf network** {**non-broadcast** | **broadcast** | **point-to-point** | **point-to-multipoint**}

Table 42-5 describes how to display OSPFv3 configuration and statistics.

Table 42-5 Displaying OSPFv3 Configuration and Statistics

Task	Command(s)
Displaying OSPFv3 configuration.	**show ipv6 ospf**
Displaying OSPFv3 link-state database information.	**show ipv6 ospf database** *type* [*link-state-id*]
Displaying information about OSPFv3 internal entries to area border routers and autonomous system boundary routers.	**show ipv6 ospf border-routers**
Displaying OSPFv3 interface configuration information.	**show ipv6 ospf interface** [**vlan** *vlan-id*]
Displaying OSPFv3 neighbor information.	**show ipv6 ospf neighbor** [*router-id*] [**detail**] [**vlan** *vlan-id*]
Displaying OSPFv3 sham link information.	**show ipv6 ospf sham-link**
Displaying OSPFv3 virtual-links configuration information.	**show ipv6 ospf virtual-links**

43

Intermediate System To Intermediate System (IS-IS) Configuration

This chapter provides information about configuring and monitoring Intermediate System To Intermediate System (IS-IS) on S-Series devices.

Using IS-IS in Your Network

IS-IS is an interior gateway link-state routing protocol, defined in ISO 10589, and operates by reliably flooding link state information throughout a network of routers within an administrative domain.

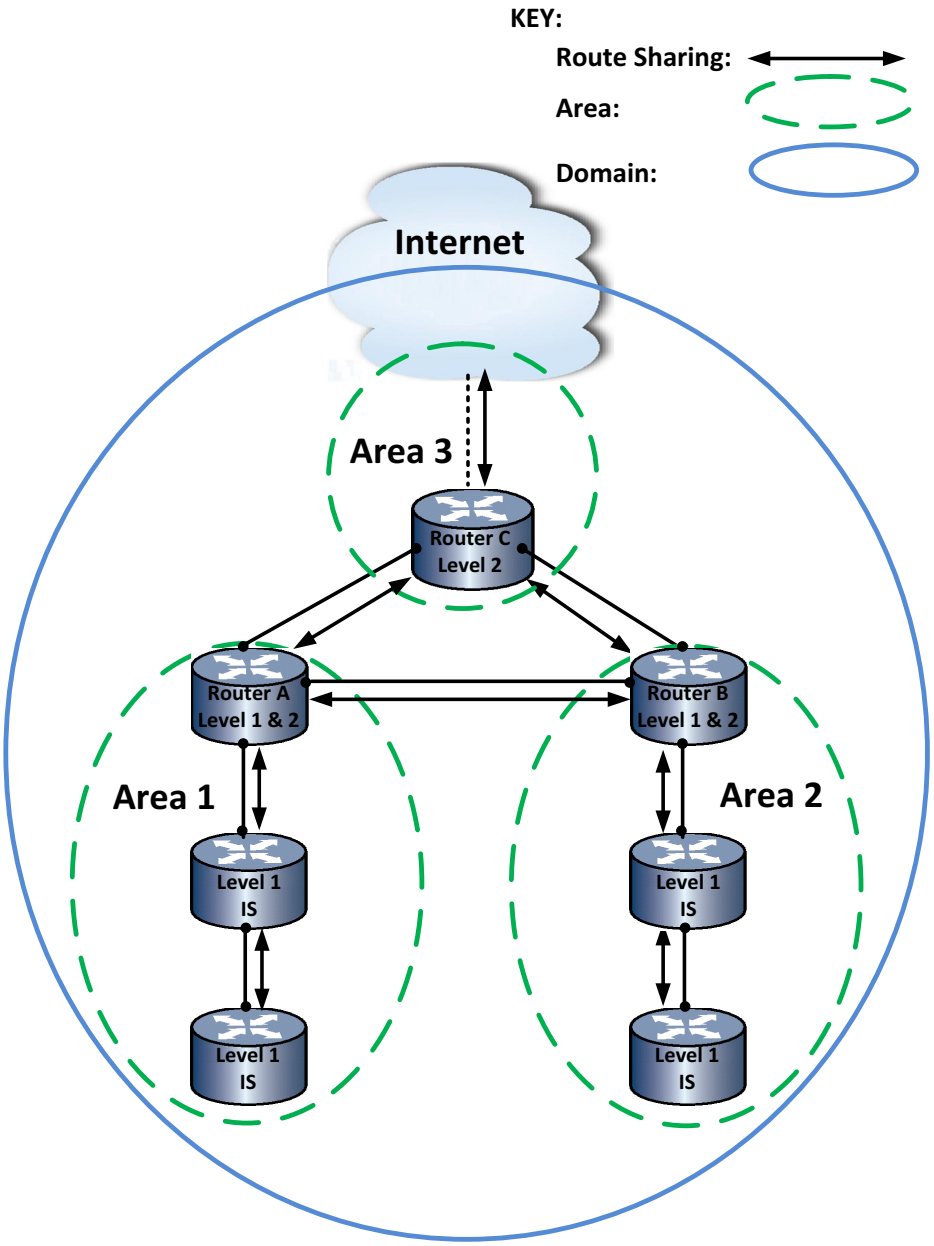
IS-IS uses the term domain to refer to any group of routers that are administered by a single organization. For example, the network owned and operated by a single carrier would be a domain. The concept of a domain is analogous to the autonomous system in OSPF.

A domain is subdivided into areas. Areas simplify management by breaking a domain into smaller entities that are easier to manage. Routing protocols that operate within the domain are known as intra-area routing protocols (Interior Gateway Protocols (IGP) in the IP world). Routing protocols that operate between domains are known as inter-area routing protocols. IS-IS handles routing information within a domain and is therefore known as an intra-area routing protocol or IGP.

Routers that handle traffic within an area are known as Level 1 routers. Routers that handle traffic between areas are known as Level 2 routers. Routers that handle traffic both within and between areas are known as Level 1 and 2 routers. Level 1 and 2 routers run two copies of the routing algorithm.

Figure 43-1 on page 43-2 displays a three area domain. Routers A, B, and C all share routes with each other. The Level 1 routers belonging to areas 1 and 2 share routes with Level 1 routers within their area. If they need to reach routers outside of their respective areas, they depend upon the Level 1 and Level 2 routers (Router A and Router B) to forward frames outside of their respective areas. Area 3 intermediate systems beyond Router C are not displayed. Routers A and B forward frames to Router C and depend upon Router C to forward frames to other intermediate systems in Area 3.

Figure 43-1 IS-IS Network Overview



IS-IS is a link-state routing protocol. That is, each intermediate system (router) in a domain is represented as being in a particular state at any given time. Depending on the state of the intermediate system, different messages are expected from neighboring intermediate systems or from that intermediate system before the intermediate system can transition to the next state and ultimately exchange routing information and process its routing tables. The packets used in the IS-IS routing protocol fall into the following classes:

- Hello packets
- Link State PDUs (LSPs),
- Sequence Number PDUs (SNP)s.

Hello packets are used to initialize and maintain adjacencies between neighboring intermediate systems. There are three types of IS-IS Hello packets:

- Level 1 LAN IS-IS Hello PDUs are used by Level 1 intermediate systems on broadcast LANs.
- Level 2 LAN IS-IS Hello PDUs are used by Level 2 intermediate systems on broadcast LANs.
- Point-to-Point IS-IS Hello PDUs are used on non-broadcast media, such as point-to-point links.

Link State PDUs (LSPs) contain link state information. There are two types of LSPs:

- Level 1 intermediate systems transmit Level 1 LSPs
- Level 2 intermediate systems transmit Level 1 and Level 2 LSPs

Sequence number PDUs are used to ensure that neighboring intermediate systems are aware of the most recent LSP from every other intermediate system. The sequence number PDUs serve a similar function to acknowledgement packets. There are two types of sequence number packets for both Level 1 and Level 2 intermediate systems:

- Partial sequence-number PDUs are used to request LSPs in broadcast networks and to acknowledge LSP receipt on point-to-point networks.
- A complete sequence number PDU contains a description of all LSPs in the database. A complete sequence number packet is used to ensure synchronization of the database between adjacent intermediate systems either periodically, or when an adjacency first comes up.

Intermediate systems running IS-IS establish an adjacency by passing hello packets to each other. Based on the responses to those packets, an intermediate system determines whether it can establish this adjacency and whether the adjacency is a Level 1 adjacency, a Level 2 adjacency or a Level 1 and 2 adjacency.

Because IS-IS was originally designed for OSI and then evolved to include IP routing protocols, it represents intermediate systems using OSI addressing. Each node (router or end system) in OSI is known by a unique identifier known as the Network Service Access Point (NSAP). See “Network Layer Addresses (NSAP)” on page 43-4 for an NSAP graphic presentation.

This address is divided into two equal parts:

- The initial domain part (IDP)
- The domain specific part (DSP).

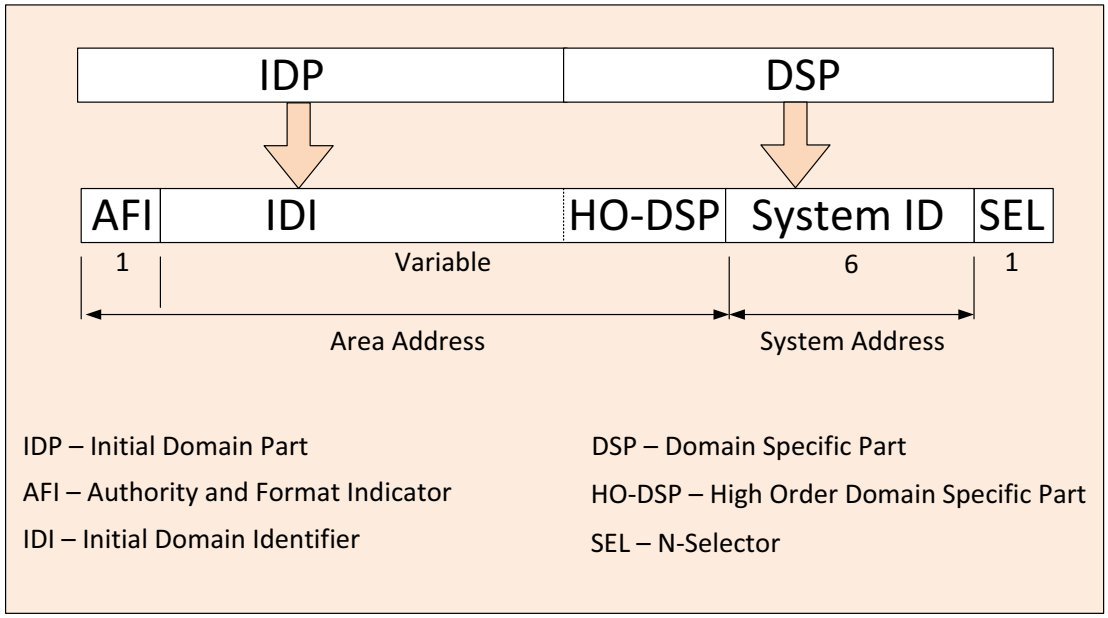
The initial domain part is further broken into the authority and format indicator (AFI), which specifies the format of the IDP, and the initial domain identifier (IDI). The AFI is always one octet in length.

The DSP is further broken into the High Order Domain Specific part (HO-DPS) and the system ID and selector (SEL). The system ID is always 6 octets. The SEL is one octet in length and is always 00.

The AFI, IDI and HO-DSP make up the area address which is variable in length, depending on the value of the AFI.

The network entity title (NET) of an IS-IS intermediate system is the six-byte NSAP and a single SEL byte set to 0.

Figure 43-2 Network Layer Addresses (NSAP)



Implementing IS-IS

To implement IS-IS:

1. Enable IS-IS on the intermediate point.
2. Enable IS-S on each interface that will use IS-IS routing to initiate the forming of an adjacency with the IS-IS neighbor.
3. Configure a NET for each area (up to three) the router will route to using IS-IS.
4. Optionally, modify IS-IS optional global parameters.
5. Optionally, modify IS-IS optional interface parameters.

IS-IS Configuration Overview

For information about...	Refer to page...
Configuring the TLV Metric Style	43-12
Configuring IS-IS Priority	43-12
Configuring the IS-IS Intermediate System as Overloaded	43-13

Enabling IS-IS Globally

IS-IS must be enabled globally on the device.

Use the **router isis** command in global configuration mode to globally enable IS-IS on the intermediate system.

The following example shows how to globally enable IS-IS routing on the device:

```
S Chassis(rw)->configure
S Chassis(rw-config)->router isis
S Chassis(rw-config-isis)->
```

The IS-IS type for the intermediate system defaults to Level 1 and 2. You can change the IS-IS type for the intermediate system to IS-IS type Level 1 or IS-IS type Level 2.

Use the **is-type** command to configure the IS-IS type for the intermediate system.

Enabling IS-IS on the Interface

IS-IS must be enabled on each interface with an IS-IS adjacency. Not all interfaces on an intermediate system participate in IS-IS routing. Because IS-IS is a link-state routing protocol, routing information is not exchanged unless at least one adjacency is formed. In IS-IS, enabling of the IS-IS protocol on an interface begins the adjacency formation process. Therefore, no routing information is exchanged until IS-IS is enabled on at least one interface.

Use the **ip router isis** command in interface configuration mode to enable IS-IS on an IPv4 interface

The following example shows how to enable IPv4 IS-IS routing on VLAN 100:

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 100
S Chassis(rw-config-intf-vlan.0.100)->ip router isis
S Chassis(rw-config-isis)->
```

Use the **ipv6 router isis** command in interface configuration mode to enable IS-IS on an IPv6 interface.

The following example shows how to enable IPv6 IS-IS routing on VLAN 100:

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 100
S Chassis(rw-config-intf-vlan.0.100)->ipv6 router isis
S Chassis(rw-config-isis)->
```

The IS-IS circuit type for the interface defaults to Level 1 and 2. You can change the IS-IS circuit type for the intermediate system to IS-IS type Level 1 or IS-IS type Level 2.

Use the **isis circuit-type** command to configure the IS-IS circuit type for the interface.

The cost of using an interface can be configured. The cost of using an interface defaults to **10**. Use the **isis metric** command in interface configuration mode to configure the interface metric.

Configuring a Network Entity Title (NET)

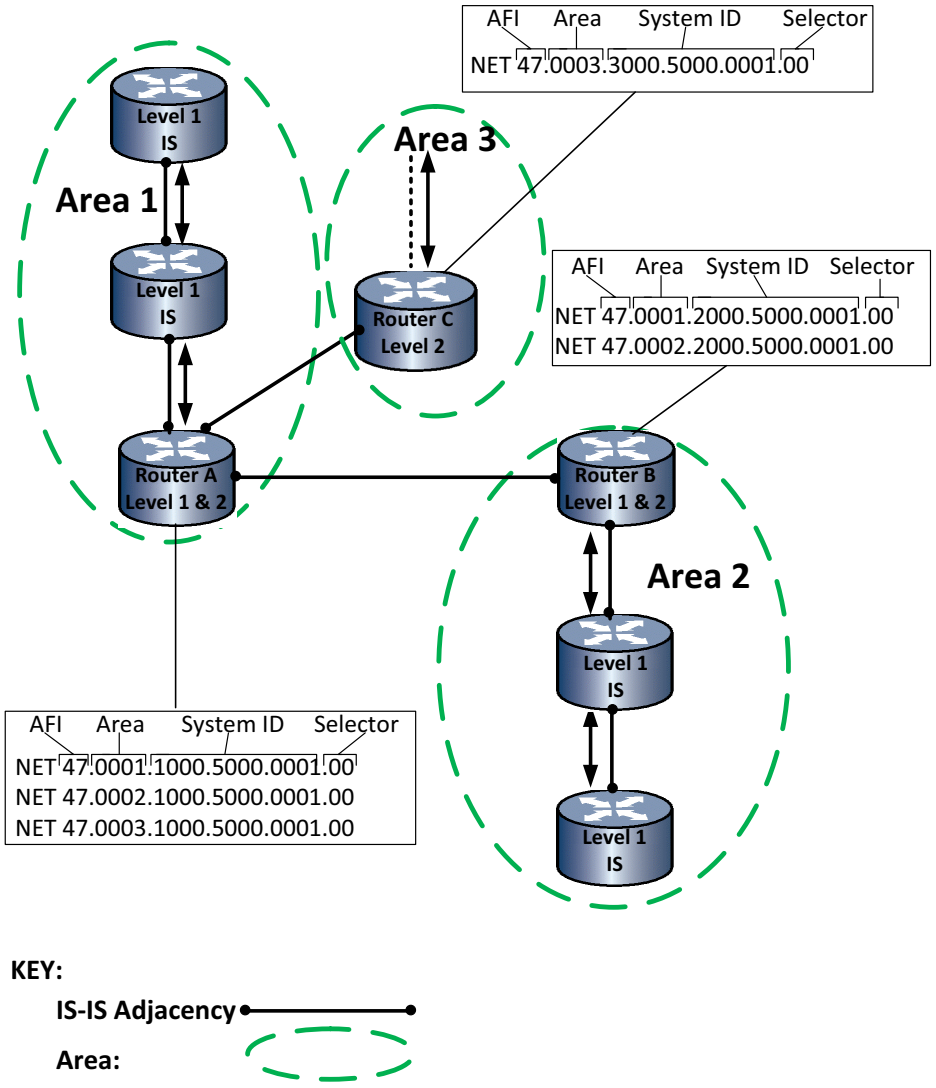
A NET is a Network Service Access Point (NSAP) address of varying length where the last byte (the NSAP-selector) is always zero. All intermediate systems within an IS-IS domain must use the same length NET. The first variable number of bytes identify the area, followed by seven fixed bytes that are divided between six bytes identifying the system ID and a single selector byte. Each intermediate system has a unique system identifier. To configure separate areas for the intermediate system, enter each area number, followed by the unique system ID for this intermediate system, followed by 00 (the NSAP-selector octet). For example: NET address **12.3333.4444.5555.6666.00** has an

- Area of **12.3333**
- System identifier of **4444.5555.6666**
- NSAP-selector of **00**

The IS-IS process does not start until at least one NET is configured.

In Figure 43-3, three areas are defined: 47.0001, 47.0002, and 47.0003. Each intermediate system can belong to up to three areas. Router A is a Level 1-2 intermediate system with adjacencies in all three areas. Router B is a Level 1-2 intermediate system with adjacencies in areas 47.0001 and 47.0002. Router C is a Level 2 intermediate system with adjacencies only in area 47.0003.

Figure 43-3 IS-IS NET Configuration



Router A is configured for:

NET **47.0001.1000.5000.0001.00**

NET **47.0002.1000.5000.0001.00**

NET **47.0003.1000.5000.0001.00**

The Level 1 adjacencies to Router A are configured with area **47.0001** and unique system IDs.

The following example configures the NETs for Router A:

```
S Chassis(rw)->configure
S Chassis(rw-config)->router isis
S Chassis(rw-config-isis)->net 47.0001.1000.5000.0001.00
S Chassis(rw-config-isis)->net 47.0002.1000.5000.0001.00
S Chassis(rw-config-isis)->net 47.0003.1000.5000.0001.00
S Chassis(rw-config-isis)->
```

Router B is configured for:

NET **47.0001.2000.5000.0001.00**

NET **47.0002.2000.5000.0001.00**

The Level 1 adjacencies to Router B are configured with area **47.0002** and unique system IDs.

The following example configures the NETs for Router B:

```
S Chassis(rw)->configure
S Chassis(rw-config)->router isis
S Chassis(rw-config-isis)->net 47.0001.2000.5000.0001.00
S Chassis(rw-config-isis)->net 47.0002.2000.5000.0001.00
S Chassis(rw-config-isis)->
```

Router C is configured for:

NET **47.0003.3000.5000.0001.00**

Router A participates in a single-area domain. The area is area 47.0001 and the intermediate system identifier is 1000.5000.0001.

The following example configures the NETs for Router C:

```
S Chassis(rw)->configure
S Chassis(rw-config)->router isis
S Chassis(rw-config-isis)->net 47.0003.3000.5000.0001.00
S Chassis(rw-config-isis)->
```

Configuring Administrative Distance

Administrative distance configures the preference given to a route learned from one protocol over the same route learned from another protocol. The route with the lowest administrative distance is installed in the IP routing table and propagated to neighbors.

Routes with distance values of 255 are not installed in the routing table.

Use the **distance** command in IS-IS configuration command mode to set the distance of IS-IS routes to the specified value.

IS-IS has a default administrative distance of 115.

The following example shows how to change the administrative distance for external IS-IS routes to **100**:

```
S Chassis(rw-config)->router isis
S Chassis(rw-config-isis)->distance isis external 100
```

Configuring IS-IS Authentication

Authentication can be set both globally to authenticate between IS-IS domains and areas. Authentication can be set in interface configuration mode for authentication between intermediate systems within an area. Authentication is enabled in global configuration mode by assigning an area password. Authentication is enabled in interface configuration mode by assigning an IS-IS password.

Use the **domain-password** command to enable domain authentication at the intermediate system global level.

Use the **area-password** command to enable area authentication at the intermediate system global level.

Use the **isis password** command to enable authentication in interface configuration mode.

The authentication mode can be set at either the global or interface mode to either MD5 or text. Specify the IS-IS level of the intermediate system or interface when configuring authentication mode.

Use the **authentication-mode** command in global configuration mode to set the authentication mode at the intermediate system global level.

Use the **isis authentication-mode** command in interface configuration mode to set the authentication mode at the interface level.

A configured key-chain can be applied to IS-IS authentication in either a global or interface context.

Use the **authentication key-chain** command in IS-IS configuration mode to apply a key chain to IS-IS authentication for the intermediate system.

Use the **isis authentication key-chain** command in interface configuration mode to apply a key chain to IS-IS authentication in an interface context.

Authentication can be configured for send frames only at both the global and interface level. When configured, no authentication will be performed on received frames for the configured context.

Use the **authentication send-only** command to configure IS-IS to only include authentication on frames sent by the intermediate system.

Use the **isis authentication send-only** command in interface configuration mode to configure IS-IS to only include authentication on frames sent by the interface.

Configuring Multiple Parallel Routes

You can set multiple parallel routes installed in the IP routing table. During packet switching, load balancing is performed among the multiple paths. A maximum of 32 paths is supported. By default, 8 paths are installed to the IP routing table.

During load balancing, a source/destination pair always uses the same path. Use the no option to reset the number of parallel paths to 8.

Use the **maximum-paths** command in IS-IS configuration command mode to configure the maximum number of multiple parallel routes.

Enabling Route Summarization

Summarizing addresses reduces the number of LSPs and the size of the link state database. Multiple addresses can be summarized for a given IS-IS instance.

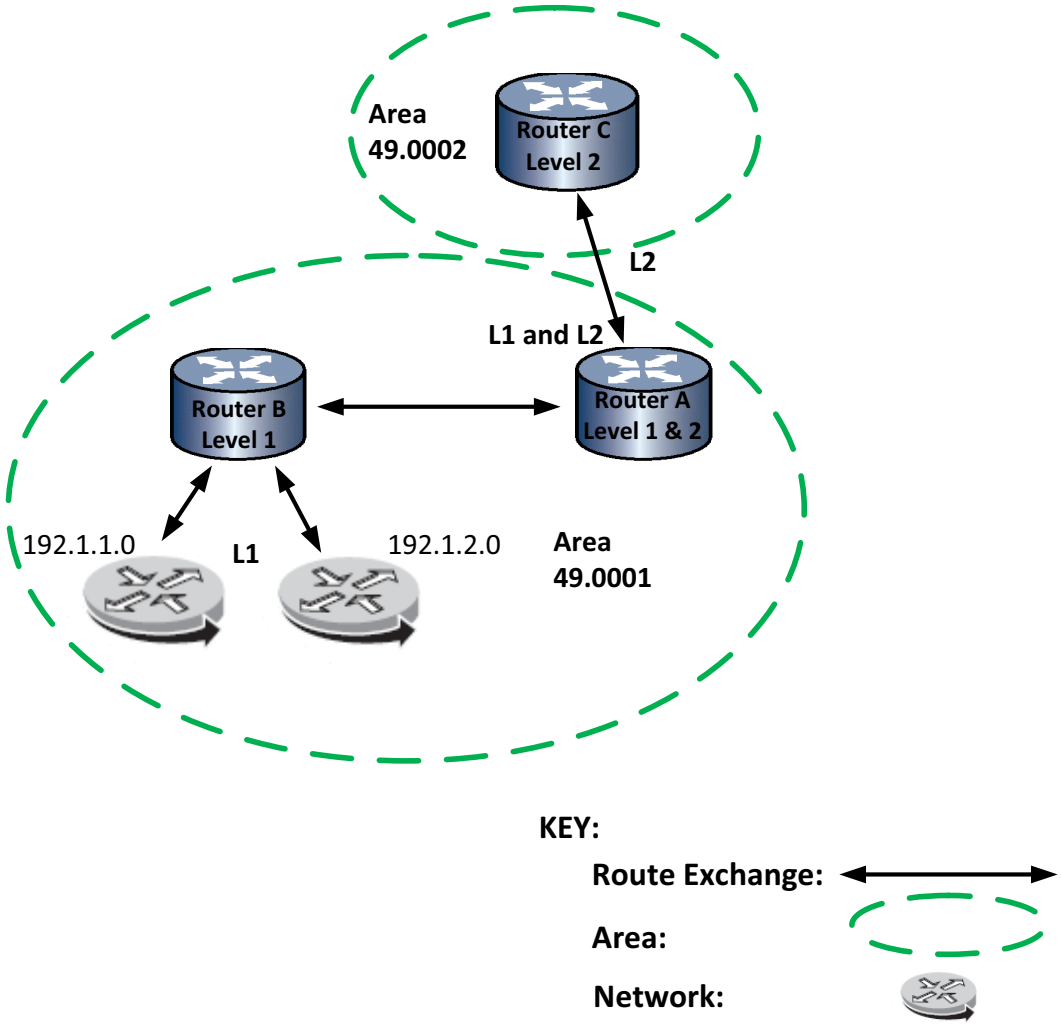
To summarize a unicast IPv6 address, you must be in the IPv6 unicast family address configuration mode. See “Configuring the IPv6 Unicast Address Family” on page 43-13 to enter IPv6 unicast family address configuration mode.

In IS-IS, routes are leaked from Level 1 to Level 2, on intermediate systems running both Level 1 and Level 2. The summary address command aggregates the addresses that are leaked from Level 1 and Level 2.

In Figure 43-4, Router A is running Level 1 and Level 2. Router B is running Level 1 and Router C is running Level 2. Router A learns networks 192.1.1.0 and 192.1.2.0 through Level 1 IS-IS from

Router B. Without a route summary, they are leaked to Level 2 and flooded to Router C as 192.1.1.0 and 192.1.2.0.

Figure 43-4 IS-IS Route Summarization



The following lines are configured on Router A:

```
S Chassis(rw)->configure
S Chassis(rw-config)->router isis
S Chassis(rw-config-isis)->summary-address 192.1.0.0/16
S Chassis(rw-config-isis)->
```

Now Router A will leak summary 192.1.0.0 to Router B but will not flood routes 192.1.1.0 and 192.1.2.0. A summary is only flooded if there is at least one Level 1 route that falls into the configured summary address range.

Configuring Route Redistribution

You can redistribute routes into IS-IS for route types and protocols: BGP, connected, OSPF, RIP, Static and blackhole. You can also redistribute IS-IS Level 2 routes into Level 1.

A route map can be applied to a redistribution configuration to filter the routes that will be redistributed for the specified route type or protocol. You can also specify a protocol metric value for BGP, connected, RIP, or static redistribution routes.

You must be in IS-IS IPv6 unicast address family configuration mode to configure redistribution on an IPv6 IS-IS intermediate system instance. See "Configuring the IPv6 Unicast Address Family" on page 43-13 for details on entering IS-IS IPv6 unicast address family configuration mode.

If you do not specify a distribute list when redistributing IS-IS level 2 routes into IS-IS level 1, all layer 2 addresses are redistributed into layer 1.

Use the **redistribute** command in IS-IS configuration or IS-IS IPv6 unicast address family configuration mode to specify the protocol or route type to redistribute into IS-IS.

Configuring IS-IS Timers

There are three global level and four interface level IS-IS timers. See Table 43-1 for IS-IS timer details.

Table 43-1 IS-IS Timers

Timer	Description	Command
LSP Generation Interval	The minimum interval between the generation of LSPs, configured in the IS-IS configuration mode.	**lsp-gen-interval**
Maximum LSP Lifetime	The maximum time the LSPs persist without being refreshed.	**max-lsp-lifetime**
SFP Interval	The minimum amount of time between Shortest Path First (SFP) processing on an IS-IS instance. When a topology change occurs the SPF calculation is run. The SPF calculation is not run when external routes change. The SPF calculation is CPU intensive. For a network with a large area and frequent topology changes you may want to increase the minimum time between SPF calculations. Increasing the SPF interval reduces the processor load, but potentially slows the rate of convergence.	**sfp-interval**
Complete Sequence Number PDU Interval	The IS-IS Complete Sequence Number PDU (CSNP) interval for the interface. Designated Routers (DRs) send out CSNP packets on the interface to maintain database synchronization.	**isis csnp-interval**

Table 43-1 IS-IS Timers (continued)

Timer	Description	Command
Hello Interval	The IS-IS Hello PDU interval for the interface. The advertised holdtime in the hello packet is set to three times the hello interval seconds. Topological changes are detected faster with a smaller hello interval, but there is more routing traffic. The number of hello packets a neighbor must miss before the intermediate system declares the adjacency down on the interface is the holdtime multiplier which can be changed using the **isis hello-multiplier** command.	**isis hello-interval**
LSP transmission interval	The minimum interval between the transmission of Link-State Packets (LSP)s on the interface.	**isis lsp-throttle**
LSP retransmission interval	The minimum interval between the retransmission of the same LSP. The retransmit interval should be greater than the expected round-trip delay between any two intermediate systems on the attached network. Retransmissions occur when LSPs are dropped. Higher retransmission values have little effect on reconvergence. The more neighbors intermediate systems have, and the more paths over which LSPs can be flooded, the higher this value can be made.	**isis retransmission-interval**

Configuring the TLV Metric Style

By default the IS-IS metric style defaults to both wide and narrow. The narrow metric is a 6-bit metric as defined in ISO 10589. The wide metric is a 4-byte metric and configures a intermediate system to generate and accept type, length, and value (TLV) object 135 for IP addresses. The TLV metric style can be configured for either both wide and narrow or wide.

Use the **metric-style** command in IS-IS configuration mode to configure the TLV metric style for the IS-IS intermediate system.

Configuring IS-IS Priority

The priority is used to determine the designated router. The intermediate system with the highest priority becomes the designated router. IS-IS does not support the concept of a backup designated router. Setting the priority to 0 does not prevent this system from becoming the designated router. If priorities are equal, the interface with the highest MAC address breaks the tie. Priority defaults to **64**.

The following example shows how to configure VLAN **100** for a priority of 80:

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 100
S Chassis(rw-config-intf-vlan.0.100)->isis priority 80
S Chassis(rw-config-intf-vlan.0.100)->
```

Configuring the IS-IS Intermediate System as Overloaded

There are some circumstances in which it is advantageous to have an IS-IS intermediate system not fully participate in forwarding traffic. For example:

- During startup this intermediate system may be temporarily too busy with the tasks associated with convergence to forward traffic.
- The intermediate system is in a test network that has connections to a production network. The overload bit prevents traffic from moving between the two networks.

You can configure the intermediate system to not forward traffic by enabling the overload bit.

When enabled, the overload bit instructs other intermediate systems not to use this intermediate system as an intermediate hop in their SPF computations. No paths through this intermediate system are visible to other intermediate systems in the domain. IP and CLNS prefixes directly connected to this intermediate system are reachable.

Use the **set-overload-bit** command in IS-IS configuration mode to enable overload on this intermediate system.

Configuring the IPv6 Unicast Address Family

By default, IS-IS configuration applies to both IPv4 and IPv6 routes. Use this command to configure IPv6 specific configuration on the device. IPv6 unicast specific address family configuration currently supports:

- Administrative distance – See "Configuring Administrative Distance" on page 43-8
- Redistribution of routes from other protocols into IS-IS – See "Configuring Route Redistribution" on page 43-11
- Route summarization– See "Enabling Route Summarization" on page 43-9

The following example shows how to enter IPv6 unicast IS-IS address family configuration mode and set the IPv6 unicast IS-IS administrative distance to 100:

```
S Chassis(rw)->configure
S Chassis(rw-config)->router isis
S Chassis(rw-config-isis)->address-family ipv6 unicast
S Chassis(rw-config-isis-af)->distance 100
S Chassis(rw-config-isis-af)->
```

Graceful Restart

IS-IS graceful restart, sometimes referred to as non-stop forwarding, provides for an IS-IS router to remain on the forwarding path during a restart of its IS-IS software. Graceful-restart has four elements to its configuration: enabling, helping a peer, database re-synchronization interval, and restart interval.

Enabling graceful restart instructs the firmware to perform a graceful restart, rather than a standard IS-IS restart. Restart is only initiated by a fail-over. Whether the failover is intentional or not, the failed router protocol is restarted on another module, and upon startup, IS-IS informs its neighbors using existing link aggregation groups. Use the **graceful-restart enable** command to enable the graceful restart ability on this router.

When the helper peer is informed that a graceful restart is taking place, it sends the restarting router its database and prevents the rest of the network from being informed there is an issue with the restarting router. The helper also monitors the network for topology changes. If no changes occur, the helper router continues to advertise as though no restart was occurring. If a topology

change does occur, graceful restart is terminated on the restarting router and a standard restart occurs. Helper mode is enabled by default, but can be disabled on an IS-IS router using the **no graceful-restart enable-help-peer** command in IS-IS router command mode.

The length of time to allow database synchronization during a graceful restart can be configured using the **graceful-restart restart-sync-interval** command in IS-IS router configuration mode.

A restart interval provides for a maximum time in seconds after which the graceful restart will terminate should it not complete or terminate for other reasons within the interval. Use the **graceful-restart restart-interval** command to change the restart interval setting.

View the router OSPF section of the **show running-config** display to verify any non-default graceful restart settings.

Configuring IS-IS

This section provides a table of feature default values and a procedure for configuring a feature system.

Table 43-2 lists IS-IS default values.

Table 43-2 IS-IS Parameters

Parameter	Description	Default Value
authentication mode	The IS-IS MD5 authentication mode that provides a cryptographic hash MD5 digest to each IS-IS PDU.	MD5
distance	The administrative distance for IS-IS routes.	115
graceful restart state	Specifies whether graceful restart is enabled or disabled on a router.	disabled
graceful restart help peer	A function on the graceful restarting router peer that specifies whether the peer will help in the graceful restarting process.	enabled
graceful restart adjacency interval	Specifies the length of time graceful restart waits for the adjacency to form.	10 seconds
graceful restart interval	Specifies the length of time a graceful restart is allowed to continue without completing before graceful restart is terminated.	65535 seconds
graceful restart sync interval	Specifies the length of time graceful restart will wait for the IS-IS database to synchronize.	60 seconds
ignore LSP errors	Specifies that IS-IS ignores link state packet checksum errors on the device.	disabled
LSP buffer size	Specifies the LSP buffer size, based upon the maximum size of LSPs originated by this IS-IS routing instance.	1492 bytes
LSP generation interval	The minimum interval between generation of LSPs.	1 second
maximum LSP lifetime	The maximum time an LSP can persist without being refreshed.	1200 seconds
maximum paths	The maximum number of parallel routes to be installed into the routing table for this device.	8
metric style	The TLV metric style for this IS-IS instance.	both (narrow and wide)

Table 43-2 IS-IS Parameters (continued)

Parameter	Description	Default Value
overload bit	When set, instructs this intermediate system to tell other intermediate systems not to use it as an intermediate hop in the SPF calculation.	not set
SPF interval	The minimum amount of time between Shortest Path First (SPF) processing for this IS-IS instance.	33 milli-seconds
CSNP interval	The interval between sending sequence number PDUs (CSNP) on the interface.	10 seconds
hello interval	The minimum amount of time between sending hello PDUs on the interface.	10 seconds
hello multiplier	The number of hello packets a neighbor must miss before the intermediate system declares the adjacency down for the interface.	3
LSP MTU	The maximum PDU size for PDUs on the interface.	1490 bytes
LSP throttle	The minimum interval between the transmission of Link-State-Packets on the interface.	30 milli-seconds
metric	The cost of using the interface.	10
priority	The priority used to determine which intermediate system on a LAN is the designated router.	64
retransmit interval	The minimum amount of time between the retransmission of the same LSP on an interface.	5 seconds

Procedure 43-1 describes feature configuration on the Extreme Networks S-Series devices. All commands used to configure IS-IS are entered in the IS-IS configuration mode after enabling IS-IS using the **router isis** command.

Procedure 43-1 Configuring Global IS-IS

Step	Task	Command(s)
1.	In global configuration mode, enable IS-IS globally on the device and enter IS-IS intermediate system configuration mode.	**router isis**
2.	Specify the Network Access Point address for this IS-IS instance in the format *xx.xxxx*.(...)*xxxx.xxxx.xxxx*.**00**.	**net** *net*
3.	Optionally configure the IS-IS type for this IS-IS instance.	**is-type** {**level-1** | **level-1-2** | **level-2**}
4.	Specify the IS-IS domain password for this device.	**domain-password** *password*
5.	Specify the area password.	**area-password** *password*
6.	Optionally specify an IS-IS authentication key chain for the device.	**authentication key-chain** *keychain* [**level-1** | **level-1-2** | **level-2**]

Procedure 43-1 Configuring Global IS-IS (continued)

Step	Task	Command(s)
7.	Optionally set the authentication mode.	**authentication mode** {**md5** | **text**} [**level-1** | **level-1-2** | **level-2**]
8.	Optionally configure IS-IS authentication only on sent IS-IS frames.	**authentication send-only** [**level-1** | **level-1-2** | **level-2**]
9.	In IS-IS configuration mode, optionally modify the IS-IS administrative distance for this IS-IS intermediate system.	**distance** [**isis** {**external** | **internal**}] *weight*
10.	Optionally configure a dynamic hostname.	**hostname dynamic** *hostname*
11.	Optionally enable IS-IS to ignore link state packet checksum errors on the device.	**ignore-lsp-errors**
12.	Optionally configure the LSP buffer size based upon the specified maximum size of the LSP's originated by the IS-IS routing instance.	**lsp-buf-size** *size* [**level-1** | **level-1-2** | **level-2**]
13.	Optionally configure the minimum interval between the generation of LSPs on the device.	**lsp-gen-interval interval** [**level-1** | **level-1-2** | **level-2**]
14.	Optionally configure the maximum time that LSPs persist without being refreshed.	**max-lsp-lifetime** *lifetime*
15.	Optionally configure the maximum number of parallel routes to be installed into the routing table for this device.	**maximum-paths** *num*
16.	Optionally configure the TLV metric style for this IS-IS instance.	**metric-style** {**wide** | **both**}
17.	In IS-IS configuration mode, optionally configure the redistribution of routing protocols into IS-IS.	**redistribute** {**bgp** | **connected** | **ospf** *process_id*| **rip** | **static** | **blackhole** | **isis level-2 into level-1** [**distribute-list** *access-list*]} [**route-map** *name*] [**metric** *metric-value*]
18.	Optionally configure the intermediate system to signal other intermediate systems not to use this intermediate system as an intermediate hop in their SPF calculations.	**set-overload-bit** [**level-1** | **level-1-2** | **level-2**]
19.	Optionally configure the minimum amount of time between Shortest Path First (SPF) processing on an IS-IS instance.	**spf-interval** *interval*
20.	Optionally create an aggregate IS-IS address for summarization of routes.	**summary-address** *ip-address*/*length*
21.	Optionally enable IS-IS graceful restart for this router.	**graceful-restart enable**
22.	Optionally set the length of time graceful restart will allow for the forming of adjacencies.	**graceful-restart restart-adj-interval** *interval*
23.	Optionally set the length of time graceful restart will attempt to complete a restart before terminating.	**graceful-restart restart-interval** *interval*
24.	Optionally specify the amount of time graceful restart will allow for the synchronization of the database.	**graceful-restart restart-sync-interval** {**level-1** | **level-1-2** | **level-2**} *interval*

Table 43-3 describes IS-IS Interface configuration on the Extreme Networks S-Series devices. All IS-IS interface configuration commands are entered in interface configuration mode. All IS-IS interface configuration is optional.

Table 43-3 Configuring IS-IS on the Interface

Task	Command(s)
If IPv6 routing will take place on the interface, enable IPv6 IS-IS routing on the interface. IPv4 IS-IS routing is enabled globally.	**ipv6 router isis**
Configure an IS-IS authentication key chain on an interface	**isis authentication key-chain** *keychain* [**level-1** | **level-2**]
Configure the IS-IS authentication mode on an interface.	**isis authentication mode** {**md5** | **text**} [**level-1** | **level-2**]
Configure IS-IS authentication only on sent IS-IS frames on an interface.	**isis authentication send-only** [**level-1** | **level-2**]
Configure the IS-IS type on an interface	**isis circuit-type** {**level-1** | **level-1-2** | **level-2**}
Configure the IS-IS complete sequence number PDU (CSNP) interval for the interface.	**isis csnp-interval** *seconds* [**level-1** | **level-1-2** | **level-2**]
Configure the IS-IS Hello Protocol Data Units interval for the interface.	**isis hello-interval** {*seconds* | **minimal**} [**level-1** | **level-1-2** | **level-2**]
Configure the number of hello packets a neighbor must miss before the intermediate system declares the adjacency down for the interface.	**isis hello-multiplier** *multiplier* [**level-1** | **level-1-2** | **level-2**]
Configure IS-IS hello padding on an interface.	**isis hello-padding**
Configure the maximum PDU size for PDUs on the interface.	**isis lsp-mtu** *size*
Configure the minimum interval between the transmission of Link-State Packets (LSP)s.	**isis lsp-throttle** *interval*
Configure the cost of using the interface.	**isis metric** *cost* [**level-1** | **level-1-2** | **level-2**]
Configure a two device network that uses broadcast media and IS-IS to function as a point-to-point link.	**isis network-point-to-point**
Configure the suppression of IS-IS packets from being transmitted by the interface and received packets from being processed by the interface.	**isis passive-interface**
Configure an authentication password for the interface.	**isis password** *password* [**level-1** | **level-2**]
Configure the priority used to determine which intermediate system on a LAN is the designated router.	**isis priority** *priority* [**level-1** | **level-1-2** | **level-2**]
Configure the minimum interval between retransmissions of the same LSP.	**isis retransmit-interval** *interval*

Procedure 43-2 describes IS-IS IPv6 unicast address family configuration on the Extreme Networks S-Series devices.

Procedure 43-2 Configuring IS-IS IPv6 Unicast Address Family

Step	Task	Command(s)
1.	From global configuration mode, optionally enter the IPv6 unicast IS-IS address family configuration mode to configure administrative distance, redistribution of routing protocols into IS-IS, and address summarization for IPv6 unicast.	**address-family ipv6 unicast**
2.	In IPv6 unicast address family configuration mode, optionally modify the IS-IS administrative distance for this IS-IS intermediate system.	**distance** [**isis** {**external** | **internal**}] *weight*
3.	In IPv6 unicast address mode, optionally configure the redistribution of routing protocols into IS-IS.	**redistribute** {**bgp** | **connected** | **ospf** *process_id*| **rip** | **static** | **blackhole** | **isis level-2 into level-1** [**distribute-list** *access-list*]} [**route-map** *name*] [**metric** *metric-value*]
4.	In IPv6 unicast address family configuration mode, optionally create an aggregate IS-IS address for summarization of routes.	**summary-address** *ip-address*/*length*

Table 43-4 describes how to display IS-IS information on the Extreme Networks S-Series devices. All IS-IS display commands can be entered from any command mode.

Table 43-4 Displaying IS-IS Information

Task	Command(s)
Display IS-IS database information for the intermediate system.	**show isis database** [**lsp** *lsp*] | [**level-1**] | [**level-2**] | [**detail**] | [**verbose**]
Display the hostname per LSP ID.	**show isis hostname**
Display the frequency and reason for LSP changes on an interface.	**show isis lsp-log**
Display IS-IS intermediate system neighbors.	**show isis neighbors**
Display the IS-IS topology.	**show isis topology**

Refer to the *Extreme Networks S-Series CLI Reference* for more information about each command.

Terms and Definitions

Table 43-5 lists terms and definitions used in this feature configuration discussion.

Table 43-5 Feature Configuration Terms and Definitions

Term	Definition
adjacency	A direct connection between IS-IS routers.
area	An area is a logical segmenting of an IS-IS domain for purposes of simplifying network management.
domain	An IS-IS network administered by a single organization.

Table 43-5 Feature Configuration Terms and Definitions (continued)

Term	Definition
hello packets	An IS-IS packet type used to initialize and maintain adjacency between neighboring intermediate systems.
intermediate system	A router in an IS-IS context.
Intermediate System to Intermediate System (IS-IS)	IS-IS is an interior gateway link-state routing protocol, defined in ISO 10589, operating by reliably flooding link state information throughout a network of routers within an administrative domain.
link state PDU	An IS-IS packet type used to exchange a router's link state information with other routers in the IS-IS network.
Network Entity Title (NET)	The portion of the NSAP that contains the system ID and the selector field.
Network Service Access Point (NSAP)	An addressing scheme to identify IS-IS intermediate points and the area they belong to.
router level	One of three possible levels (L1, L2, and L1 and L2) assigned to a router to determine whether route exchange is limited to an area or can be between areas.
sequence number PDUs	An IS-IS packet type used to ensure that neighboring intermediate systems are aware of the most recent LSP from every other intermediate system.

44

Border Gateway Protocol (BGP) Configuration

This chapter provides the following information about configuring and monitoring BGP on Extreme Networks S-Series devices:

Using BGP in Your Network

The Border Gateway Protocol (BGP), documented in RFC 4271, is the standard protocol for routing between administrative domains. BGP refers to an administrative domain as an Autonomous System (AS). BGP is an exterior gateway routing protocol (EGP). BGP exchanges routing information among neighboring routers in different autonomous systems. An autonomous system is a set of routers under a single administration. AS numbers supported are 1 to 4294967295.

An AS typically uses a single Interior Gateway Protocol (IGP), such as OSPF, to propagate routing information among its routers.

A BGP system establishes sessions with neighboring routers, or peers, and maintains a database of network reachability information that it exchanges with its neighbors via update messages. BGP uses the Transmission Control Protocol (TCP) and port 179 for establishing connections.

Path Attributes

BGP routing updates include the complete route to each destination, as well as other information related to the route. Route information is included in the path attributes. BGP uses path attributes to provide more information about each route. Path attributes can also be used to distinguish between groups of routes to determine administrative preferences, allowing greater flexibility in determining route preference to achieve a variety of administrative ends. Supported BGP attributes include IP next hop, Multi-Exit Discriminator (MED), and local preference. BGP also uses path attributes to maintain the AS path.

The AS path is a path attribute that provides a list of the AS numbers the route traverses. The AS path is used for loop detection. Its length is used as a route selection criteria in the event the same prefix is learned from multiple peers. BGP uses the AS path and the path attributes to determine the network topology. This, in turn, enables BGP to detect and eliminate routing loops and to make routing policy decisions.

Refer to "Using AS-Path Regular Expressions" on page 44-7 for information about using regular expressions when configuring AS path preference in route-maps.

Peers and Peer Groups

A peer is the BGP router's next hop neighbor. Peers can be organized into peer groups. A peer group is a group of neighbors that share the same attributes. You assign neighbors to the peer group, and create attributes for the group. Peer groups enable you to reduce the time and effort needed for configuration. You can override a peer group's policy by configuring unique policies for a specific peer group member.

BGP Sessions

BGP supports two basic types of sessions between neighbors: internal (sometimes referred to as IBGP) and external (EBGP). Internal sessions are run between routers in the same autonomous system. External sessions run between routers in different autonomous systems. When a router routes to an external peer, the local AS number is prepended to the AS path. This means that routes received from an external peer are guaranteed to have the AS number of that peer at the start of the path. In general, routes received from an internal neighbor will not have the local AS number prepended to the AS path. Those routes will have the same AS path that the route had when the first internal neighbor received the route from an external peer. Routes with no AS numbers in the path may be legitimately received from internal neighbors. BGP considers these routes internal to the receiver's own AS.

External BGP sessions may or may not include the Multi-Exit Discriminator (MED) among its path attributes. BGP uses MED to break ties between routes with equal preference from the same neighboring AS.

Internal BGP sessions carry the local preference attribute. The larger the local preference value, the greater the route is preferred within an AS. Internal sessions can optionally include the MED, carried in from external sessions.

Routes

A route consists of a prefix, a prefix length, and a set of information indicating policies and preference to reach the destinations indicated by the prefix. A prefix is made up of a dotted decimal formatted network identifier that includes a length that specifies the number of significant bits in the network. The route prefix is contained in the Network Layer Reachability Information (NLRI) and the BGP next hop path attribute determines where packets matching the prefix should be forwarded. The BGP next hop may be non-directly connected. In this case, for the route to be installed in the routing table, the router must have a route to the BGP Next Hop.

You can redistribute routing information between BGP and another protocol, and use route-maps to control the route updates.

Routing Policy

Routing policies can be used to filter routes both on an import and export basis, based upon its IP prefix, community (RFC 1997), extended community (RFC 4260), AS path, source IP address, and IP next hop. Routing policy is configured in a route-map, which is then applied to the route.

Confederations and Route Reflectors

Confederations enable you to divide a large AS into several smaller ASs, or to create an AS out of members of multiple ASs. Confederations are still fully meshed but require only a single connection to other ASs, reducing the number of peering relationships. From the outside, the confederation of smaller ASs looks like a single AS. Confederations are defined in RFC 3065

Route reflection enables you to configure a BGP router to advertise the routes learned from clients to other clients. This eliminates the full mesh requirement. You can configure one or more routers

in the AS to be reflectors. The other routers are configured as clients. Route reflection is defined in RFC 4456.

BGP Sub-Features

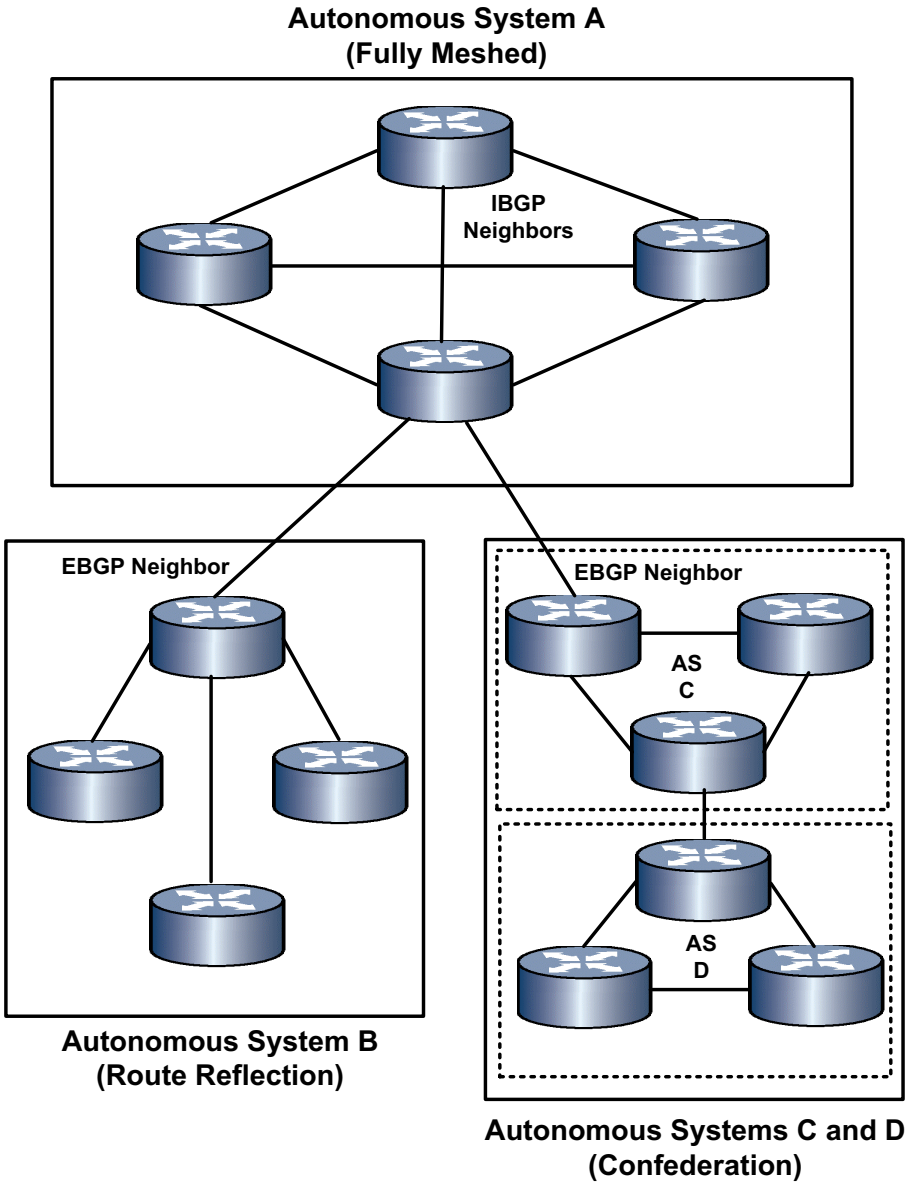
Supported BGP sub-features include:

- Graceful restart – Provides for the continued processing of the data-forwarding plane of a router should the control plane fail (RFC 4724)
- Outbound Route Filtering – Allows a BGP speaker to send to its BGP peer a set of Outbound Route Filters (ORFs), which the peer applies in addition to its locally configured outbound filters (if any), to constrain its outbound routing updates to the speaker (RFC 5291)
- Route Refresh – Allows for the dynamic exchange of route refresh requests between BGP speakers and subsequent re-advertisement of the respective Adj-RIB-Out (RFC 2918)
- Route-Flap Dampening – Treats routes that are being announced and withdrawn at a rapid rate as unreachable, based upon a route penalty for each route withdrawal, and reachable again, based upon a configurable decay over time of that route penalty (RFC 2439)
- Multiprotocol BGP Extensions – Enable BGP to carry routing information for multiple Network Layer protocols such as IPv6 and IPX (RFC 2858)
- 4-Octet AS numbers – Allows for the encoding of 4-octet AS numbers (RFC 4893)
- TCP/MD5 Authentication – Enhances BGP security by defining a TCP option for carrying an MD5 digest in a TCP segment that acts like a signature for that segment, incorporating information known only to the connection end points (RFC 2385)
- Conditional Advertisement – Provides for the sending of BGP announcements, in addition to normal announcements, when a route specified in the configured advertise map does not exist in the configured non-exist map
- Aggregation – Provides for the aggregating of one or more specific routes into a single aggregate route, if a more specific route of the aggregate route exists in the BGP routing table.
- Soft Reconfiguration – Speeds up the route installation process when an inbound policy change occurs by keeping a local copy of the routes for the specified peer or group

Figure 44-1 shows a sample BGP topology with four autonomous systems:

- Autonomous system **A** displays a standard fully meshed AS
- Autonomous system **B** displays a route reflected topology
- Autonomous system **C** and **D** displays a confederation topology with two confederations

Figure 44-1 BGP Topology



Implementing BGP

Before configuring BGP on the routers in your network, map out the network BGP topology including autonomous systems (full mesh, route reflected, and confederation), member routers, router peers, peer policy (route-maps)

Required steps to implement BGP in your network:

- Configure each router specifying the autonomous system the router belongs to
- Configure each router as part of a full mesh, route reflection, or confederation topology
- Configure all IBGP and EBGP neighbors for the router including all optional neighbor parameters specified in "Configuring BGP Neighbor Parameters" on page 44-34

BGP parameters with default values that can be modified:

- Optionally modify the route MED value using an applied route-map and optional MED behaviors using the appropriate BGP commands
- Optionally modify the local preference of advertised routes for the router
- Optionally modify the BGP route selection priority (distance) compared to other protocols for the router
- Optionally modify maximum allowed EBGP and IBGP ECMP routes for the router

BGP features that can be configured on the router:

- Optionally configure aggregate addresses
- Optionally configure soft reset on the router by configuring soft reconfiguration for the neighbor or automatic router refresh for the router
- Optionally configure flap dampening on the router
- Optionally enable graceful restart on the router
- Optionally configure outbound route filtering for the router
- Optionally configure BGP route-maps and apply them to configured neighbors and route redistribution
- Optionally configure Syslog and trap behavior for changes in peer state

BGP Overview

Injecting Routes Into BGP

Routes can be injected into BGP using either redistribution for connected, static, RIP, or OSPF routes or by specifying a network prefix and length using the **network** command. The route or

routes are injected into BGP as long as they are valid routes with resolvable next hops. Once injected into BGP, these routes can be advertised to BGP peers.

Using Redistribution

With redistribution, the user specifies the source protocol in BGP router or address family configuration mode. Redistribution can be configured for all routes of the specified type or routes can be filtered using a route-map.

Redistribution entries are created with a specified source and destination protocol to allow redistribution from the source to the destination protocol. The user can also configure a route-map to specify a set of matching prefixes as well as to set route attributes on matching routes.

In the S-Series implementation for the redistribution route-map, matching is performed only on an IP prefix as specified in an access-list. The redistribution command line allows for setting MED, local-preference, AS-path limit, and origin to matching routes. Filtering on AS-path regular expressions is supported; see "Using AS-Path Regular Expressions" on page 44-7.

BGP route-maps support setting the AS, AS-path limit, community, a number of extended community values, local preference, MED, IP next hop, origin, ORF-association local, weight, and flap table. See Chapter 51, **Route-Map Manager Configuration** for BGP route-map configuration details.

Use the **redistribute connected** command, optionally specifying a route-map, AS-path limit, origin, MED, and local preference for the route, to inject all or filtered connected routes into BGP.

In the following example BGP is configured to redistribute connected routes that match the contents of the **connRoute** route-map:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->redistribute connected route-map connRoute
```

Use the **redistribute rip** command, optionally specifying a route-map, AS-path limit, origin, MED, and local preference for the route, to inject all or filtered RIP routes into BGP.

In the following example BGP is configured to redistribute all RIP routes with the local preference set for **100**.

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->redistribute rip local-pref 100
```

Use the **redistribute static** command, optionally specifying a route-map, AS-path limit, origin, MED, and local preference for the route, to inject all or filtered static routes into BGP.

In the following example BGP is configured to redistribute all static routes with the local preference set for 100.

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->redistribute static local-pref 100
```

Use the **redistribute ospf** command, optionally specifying the route-map, AS-path limit, origin, MED, and local preference attributes for the route, to inject all or filtered OSPF routes into BGP.

In the following example BGP is configured to redistribute OSPF routes that match the contents of the **OSPFroutes** route-map.

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->redistribute ospf route-map OSPFroutes
```

Using the Network Command

With the **network** command, the user explicitly specifies the Network IP prefix to be injected into BGP. The route will be injected into BGP for advertisement to BGP peers as long as the local router

has a route to the prefix with a reachable next-hop. The **network** command supports the injection of the default route (0.0.0.0/0) into BGP, if the route is present in the routing table.

Use the **network** command, specifying the network prefix and length and optionally specifying the route-map, AS-path limit, origin, MED, and local preference attributes for the route, to inject a route into BGP.

The following example imports the network 10.1.0.0 with a mask of 255.255.255.0 into BGP. Additionally, this network range will be advertised to other peers.

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 159.1.1.9
S Chassis(su-config-bgp)->network 10.1.0.0/24
```

The following example imports the prefix 2001::/64 into BGP. This network will be advertised based upon the **routes1** route-map contents with origin set to IGP.

```
S Chassis(su-config)->router bgp 65151
S Chassis(config-router-bgp)-> bgp router-id 1.2.3.4
S Chassis(config-router-bgp)-> network 2001::/64 route-map routes1 origin 0
```

Using AS-Path Regular Expressions

The BGP AS path attribute includes a list of autonomous systems that routing information has passed through to get to a specified router and an indicator of the origin of this route. Each autonomous system through which a route passes prepends its AS number to the beginning of the AS path. The AS path is used to prevent routing loops in BGP.

This routing information can be used to prefer one path to a destination network over another. The route-map supports the configuration of AS path preference when importing and exporting routes. The matching of an AS path packet attribute is configured in a route-map using the **match as-path** command.

The **match as-path** command supports the following regular expressions:

Table 44-1 AS-Path Regular Expressions

Character	Description	Example
^	Start of string.	^200 matches any string starting with “200”.
$	End of string.	200$ matches any string ending with “200”.
.	Matches any character.	2.0 match “200”, “210”, “220”
*****	Matches the preceding element zero or more times.	22* matches “2”,” 22”, “222”
.*	Matches any character any number of times.	This is a match all.
[]	Matches a single character inside the brackets.	[d] matches the character “d”.
[-]	Matches a range.	[0-9] matches any number from 0 to 9.
()	Specifies a subexpression.	(200:500) is treated as a single entity.
[^]	Matches any single character not specified in the brackets.	[^er] matches all characters except for “e” and “r”.
?	Matches the preceding exactly.	200? matches “200” only.

Table 44-1 AS-Path Regular Expressions (continued)

Character	Description	Example
+	Matches the preceding element one or more times.	200? matches “200”, “200200”, “200200200”.
|	Matches either the expression before or the expression after the operator (the choice operator).	100 | 250 matches either “100” or “250”

Note: Regular expressions are also supported by the BGP community and extended community attributes.

This example shows how to match a packet AS path attribute that starts with AS number **20313** and with the next AS number ending with **13**:

```
S Chassis(su)->configure
S Chassis(su-config)->route-map bgp bgprm1 permit
S Chassis(su-config-route-map-bgp)->match as-path ^20313.*13$
S Chassis(su-config-route-map-bgp)->show route-map bgprm1
 route-map bgp bgprm1 permit 10
  match afi ipv4
  match safi unicast
  match as-path "^20313_*13$"
S Chassis(su-config-route-map-bgp)->
```

See Chapter 51, **Route-Map Manager Configuration** for BGP route-map configuration details.

Route Selection Preference

When a route to the same prefix can be reached through two or more paths, the BGP local-preference attribute can be used within an AS to favor a specified path based on highest local-preference value. The local-preference value is only applicable within the local AS.

Use the **bgp local-pref** command, in BGP configuration mode, to set the local-preference for advertised routes.

The following example sets the local-preference for advertised routes to 150:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 151.1.1.9
S Chassis(su-config-bgp)->bgp local-pref 150
S Chassis(su-config-bgp)->
```

Multi-Exit Discriminator (MED)

MEDs are used when an AS has multiple connections to another AS. Routes are advertised on both connections with different MEDs to specify a preferred path, typically for purposes of load balancing. By default, the MED for routes from different Autonomous Systems to the same destination are not compared. When two routes to the same destination are received from peers in different Autonomous Systems, the MED feature allows you to specify whether to compare those MEDs. When choosing between these routes, assuming that nothing else makes one preferable to the other (such as a configured policy), the values of the differing MEDs are used to choose the route to use. In this comparison, the route with the lowest MED is preferred.

By default, BGP sorts paths based on the neighbor AS and MED so that paths are sorted the same way every time. This results in a deterministic best-path selection. BGP can be configured to not take the neighbor AS into consideration when comparing the MED for each path. In this case, BGP compares MEDs when multiple routes with differing MEDs are received from peers in different autonomous systems.

The route MED value is set using a BGP route-map applied to the route.

Routes without a configured MED are treated as having the best possible MED.

Use the **bgp deterministic-med** command, in BGP configuration mode, to enable the deterministic processing of MEDs based upon the neighbor AS and MED. Deterministic MED is the default behavior.

The following example disables BGP deterministic MED for BGP router **65151**:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->no bgp deterministic-med
S Chassis(su-config-bgp)->
```

Use the **bgp always-compare-med** command, in BGP configuration mode, to compare MEDs when multiple routes with differing MEDs are received from peers in different autonomous systems.

The following example enables the comparison of MEDs from different ASs:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp always-compare-med
S Chassis(su-config-bgp)->
```

Route Aggregation

Route aggregation allows you to aggregate one or more specific routes into a single route. Aggregate routes are only created if a more specific route of the aggregate route exists in the BGP routing table. Aggregate route configuration options provide for:

- Creating and advertising the aggregate route while at the same time suppressing the advertisement of all the more specific routes for this aggregate through route summarization.
- Retaining the advertisement of the AS-Path information for the specific routes of the aggregate. The default behavior for an aggregate route is to suppress the AS-Path information for the specific routes of the aggregate. It may be desirable to retain AS-Path information for routes in the aggregate that belong to an AS outside of the AS in which the aggregate is created.
- Enabling both route summarization and advertisement of the AS-Path information for the specific routes of the aggregate.
- Creating and advertising an aggregate while at the same time suppressing only those specific routes that match clauses in the applied route-map. Prefixes contained in the aggregate route that are not specifically matched in the route-map are not suppressed. You can not use this option in conjunction with route summarization.
- Creating and advertising an aggregate, while at the same time allow specifying in a route-map which AS path information is retained in the aggregate. This option is used in conjunction with retaining the advertisement of the AS-Path information for specific routes of the aggregate. You can not use this option in conjunction with route summarization.
- Creating and advertising an aggregate, while at the same time allowing for the modifying of aggregate route attributes specified in the route-map.

The following example creates and advertises aggregate route **200.51.0.0/22** and suppresses the advertisement of all the more specific routes for this aggregate:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->aggregate-address 200.51.0.0/22 summary
S Chassis(su-config-bgp)->
```

The following example sets the MED attribute to **50** for routes in aggregate route **200.51.0.0/22** using route-map **attrmap1**:

```
S Chassis(su-config)->route-map bgp attrmap1 permit 10
S Chassis(su-config-route-map-bgp)->set med 50
S Chassis(su-config-route-map-bgp)->exit
S Chassis(su-config)->show route-map attrmap1
 route-map bgp attrmap1 permit 10
  set med 50
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->aggregate-address 200.51.0.0/22 attribute-map attrmap1
S Chassis(su-config-bgp)->
```

The following example retains AS-path information for routes **200.51.1.0/24** using route-map **advmap1** in aggregate route **200.51.0.0/22**:

```
S Chassis(su-config)->ip prefix-list advlist1 permit seq 1 200.51.1.0/24
S Chassis(su-config)->route-map bgp advmap1
S Chassis(su-config-route-map-bgp)->match prefix-list advlist1
S Chassis(su-config-route-map-bgp)->exit
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->aggregate-address 200.51.0.0/22 advertise-map advmap1
S Chassis(su-config-bgp)->
```

Source IP Address Update to the Peer

By default, BGP sets the source IP address of the BGP message to the outgoing interface. BGP checks the source IP address of the received message against the configured address of the BGP peer. If there is a mismatch, BGP discards the message. For EBGP neighbors that are directly connected and there is not an alternative path to the peer, the default outgoing interface as source address is sufficient. If the connection fails, no alternative route is available anyway.

In the case of an IBGP connected peer with an alternative route, should the connection on the outgoing interface fail, the route is withdrawn. By setting the source address to a virtual interface, such as a loopback interface, because the source address of the route is still available when the connection fails and an alternative route is available, BGP reconverges and installs the alternative route. If multiple paths exist between the BGP routers, using a loopback interface as the neighbor's source address can add stability to the network.

Use the **neighbor update-source** command in BGP router configuration mode to specify an alternative source address instead of the default directly connected interface address as the source address advertised for this IBGP router.

The following example causes the TCP session to peer **168.192.50.5** to be established over the loopback interface **4.3.2.1**:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 1.1.1.1
```

```
S Chassis(su-config-bgp)->neighbor 168.192.50.5 remote-as 5
S Chassis(su-config-bgp)->neighbor 168.192.50.5 update-source 4.3.2.1
```

See "Configuring Source IP Address Update" on page 44-35 for a remote peer source IP address update configuration example.

Scalability and the Peer Full Mesh Requirement

BGP requires that all internal peers must establish a peer relationship with each other, which is called a full mesh. Full mesh networks scale very poorly. This can result in a large routing table and a management nightmare. BGP provides two techniques for reducing the full mesh: confederations and route reflectors.

Confederations

The confederations extension to BGP, defined in RFC 3065, provides for the configuration of AS confederations. An AS confederation is a collection of routers, belonging to one or more autonomous systems, advertised as a single AS number to BGP speakers that are not members of the confederation.

Each AS confederation has a confederation ID. A router member of the confederation advertises itself to non-confederation peers using the AS confederation ID. A router member of the confederation advertises itself to other confederation peers using its AS number. A confederation ID can be any value from **1** to **65535**.

Each router member of a confederation must identify its confederation member peers.

Confederation information in the AS path sent to a neighbor peer is included by default. Inclusion of confederation information in the AS path sent to a neighbor peer can be disabled. It is possible to have AS path segments that do not adhere strictly to the confederation standard. Strict confederation path standards can be enforced. Strict confederation path enforcement is disabled by default.

See Figure 44-1 on page 44-3 for a depiction of a BGP confederation topology.

Use the **bgp confederation-id** command, in BGP configuration mode, to specify the confederation this router belongs to.

The following example configures the BGP router to be a member of BGP confederation **100**:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 151.1.1.9
S Chassis(su-config-bgp)->bgp confederation-id 100
S Chassis(su-config-bgp)->
```

Use the **neighbor confed-member** command, in BGP configuration mode, to configure the neighbor as a member of the router's confederation.

The following example configures neighbor **200.51.1.1** as a member of this router's confederation:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 159.1.1.9
S Chassis(su-config-bgp)->bgp confederation-id 100
S Chassis(su-config-bgp)->neighbor 200.51.1.1 confed-member
S Chassis(su-config-bgp)->
```

Use the **neighbor aggregate-confed** command, in BGP configuration mode, to enable the inclusion of confederation information in the AS path sent to this router's peers.

The following example disables the inclusion of confederation information in the AS paths sent to this router's peers:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 159.1.1.9
S Chassis(su-config-bgp)->no neighbor 200.51.1.1 aggregate-confed
S Chassis(su-config-bgp)->
```

Use the **bgp strict-confeds** command, in BGP configuration mode, to enable BGP to drop AS-Paths with non-standard confederation segments.

The following example enables the strict-confeds feature on this router:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp strict-confeds
S Chassis(su-config-bgp)->
```

See "Configuring BGP Confederations" on page 44-37 for a BGP confederation configuration example.

Route Reflection

Route reflection enables you to configure a BGP speaker as a route reflector which passes internally learned routes to a cluster of linked IBGP neighbors. The route reflector configured router advertises the routes it has learned from each linked client to the other linked clients in the AS. In a route reflection topology, the route reflector is the hub, and each client only peers with the the hub. This eliminates the full mesh requirement. You can configure one or more routers in the AS to be reflectors. Some or all of the other routers for the AS are configured as clients. Route reflection is defined in RFC 4456.

Route reflection clients only peer with the route reflector. Route reflection configuration only occurs on the route reflector, identifying each route reflection client.

Multiple route reflectors can be configured in an AS. Multiple route reflectors can belong to a single route reflection cluster. A route reflection cluster is identified by a unique ID. If only a single route reflector is configured for an AS, the cluster ID defaults to the router ID of the route reflector.

See Autonomous System B of Figure 44-1 on page 44-3 for a depiction of a route reflection topology.

Use the **neighbor route-reflector-client** command, in BGP configuration mode, to identify each client for the route reflection cluster.

The following example specifies that the neighbor **168.192.50.5** is a client of route reflector **1.1.1.1**:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 1.1.1.1
S Chassis(su-config-bgp)->neighbor 168.92.50.5 remote-as 5
S Chassis(su-config-bgp)->neighbor 168.92.50.5 route-reflector-client
```

Use the **bgp cluster-id** command, in BGP configuration mode, to specify a unique route reflection cluster ID the route reflector(s) belong to.

The following example configures a cluster ID of 1.2.3.4 for router 1.1.1.1:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 1.1.1.1
S Chassis(su-config-bgp)->bgp cluster-id 1.2.3.4
S Chassis(su-config-bgp)->
```

See "Configuring Route Reflection" on page 44-40 for a route reflection configuration example.

Outbound Route Filtering (ORF)

The ORF feature allows a BGP speaker to notify a peer, using route-refresh messages, of the prefixes, communities, or extended-communities the router is interested in receiving updates for. Instead of using an inbound route-map to filter a large set of routes for installation into the local-RIB, this feature allows the router to ask for the particular set they are interested in. ORF can be configured to send route-refresh messages to the peer, receive them from the peer, or both.

The ORF capability is configured in BGP router configuration mode. The peer must also support the ORF capability. Configuring the ORF capability results in the advertisement of the ORF capability OPEN message. If the peer recognizes this capability, the peer advertises the capability in its OPEN message, otherwise, the peer sends a notification with an unrecognized capability error code. In this case, the local router will resend its OPEN message without the capability advertised.

Use the **bgp orf comm-filter** command to configure ORF for community filtering on both the local router and the peer router.

This example configures BGP to send the ORF capability for community filtering for IPv4 unicast to the peer:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 151.1.1.9
S Chassis(su-config-bgp)->bgp orf ipv4 unicast comm-filter send
S Chassis(su-config-bgp)->
```

Use the **bgp orf extcomm-filter** command to configure ORF for extended community filtering on both the local router and the peer router.

This example configures BGP to send ORF capability for extended community filtering for IPv4 unicast:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 151.1.1.9
S Chassis(su-config-bgp)->bgp orf ipv4 unicast extcomm-filter send
S Chassis(su-config-bgp)->
```

Use the **bgp orf prefix-filter** command to configure ORF for prefix filtering on this router.

This example configures BGP to send the ORF capability for prefix filtering to the BGP peer for IPv4 unicast:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 151.1.1.9
S Chassis(su-config-bgp)->bgp orf ipv4 unicast prefix-filter send
S Chassis(su-config-bgp)->
```

Conditional Advertisement

The conditional advertisement feature allows a service provider to advertise certain routes to a preferred subnet under normal operational conditions, while maintaining the ability to move its traffic to an alternative subnet should its preferred routes fail. The conditional advertisement feature uses two route-maps to achieve this capability:

- The **non-exist-map** route-map which contains match prefix-list clauses for the preferred route(s) used under normal operational conditions

- The **advertise-map** route-map which contains match prefix-list clauses for the alternative route(s) used only if a preferred route is not available

Should any route specified in the non-exist route-map fail (no longer exist), BGP advertises all the routes specified in the advertise route-map, otherwise routes in the advertise route-map are not advertised to the peer. The conditional advertisement feature can be used to reduce traffic within an AS.

To configure the conditional advertisement feature:

- Create two prefix lists: one used to match prefixes for the advertise route-map; a second to match prefixes for the non-exist route-map
- Create a BGP route-map with a match clause for the advertise prefix-list
- Create a BGP route-map with a match clause for the non-exist prefix-list
- Apply the two route-maps to the **neighbor advertise-map** command

The following example:

- Configures an advertise map prefix-list named **adv-list1** and assigns it to BGP route-map **adv-map1**, specifying **prefix 1.0.0.0/8** as the advertised prefix
- Configures a non-exist map prefix-list named **non-exist-list1** and assigns it to BGP route-map **non-exist-map1**, specifying prefix **2.0.0.0/8** as the non-exist map prefix
- Configures a BGP advertise map for neighbor **192.168.12.112** that assigns **adv-map1** as the advertise map route-map and **non-exist-map1** as the non-exist map route-map

```
S Chassis(su-config)->ip prefix-list adv-list1 permit seq 1 1.0.0.0/8
S Chassis(su-config)->route-map bgp adv-map1
S Chassis(su-config-route-map-bgp)->match adv-list1 adv-map1
S Chassis(su-config-route-map-bgp)->exit
S Chassis(su-config)->ip prefix-list non-exist-list1 permit seq 1 2.0.0.0/8
S Chassis(su-config)->route-map bgp non-exist-map1
S Chassis(su-config-route-map-bgp)->match non-exist-list1 non-exist-map1
S Chassis(su-config-route-map-bgp)->exit
S Chassis(su-config)->router bgp 1
S Chassis(su-config-bgp)->neighbor 192.168.12.112 advertise-map adv-map1
non-exist-map non-exist-map1
S Chassis(su-config-bgp)->
```

See “Configuring Conditional Advertisement” on page 44-43 for a BGP conditional advertisement example.

BGP Soft Reset

BGP soft reset configuration determines BGP behavior when inbound route policy changes. New policy needs to be applied to routes to determine which are admitted into the RIB. BGP soft reset can be applied in three ways.

Internally Stored Route Reconfiguration

The soft reset of BGP routes can be based upon local internal storing of routes. For the soft reconfiguration approach, the received routes are stored in a dedicated table. When the inbound policy changes, the new policy is automatically applied. This approach is memory intensive, due to the storage of all the routes per peer or peer-group in a separate table.

Use the **neighbor soft-reconfiguration** command, in BGP router configuration mode, to enable the soft reconfiguration option on the local router. This command requires that you either specify the peer IP address or the peer group to be enabled for soft reconfiguration.

The following example turns on the route refresh capability for peer **10.10.25.1/24**:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 159.1.1.9
S Chassis(su-config-bgp)->neighbor 10.10.25.1/24 remote-as 5
S Chassis(su-config-bgp)->neighbor 10.10.25.1/24 soft-reconfiguration
```

Route-Refresh

Route refresh, defined in RFC 2918, is a capability that the peer advertises in the OPEN message during session establishment. If both the local router and its peer agree to support this capability, a router can send a route refresh message to its peer whenever the local router's inbound policy changes. The peer will respond by resending its update messages. The local router can then reapply its policy without tearing down the BGP connection and without locally storing the received routes.

If the route refresh capability is enabled, the route refresh message is generated automatically when the inbound policy changes. Route refresh is enabled by default. If the soft reconfiguration soft reset method is enabled (see Internally Stored Route Reconfiguration) route refreshes are not sent.

Use the **bgp automatic-route-refresh** command, in BGP router configuration mode, to enable route refresh on the local router.

The following example disables BGP automatic route refresh for BGP router 65151:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->no bgp automatic-route-refresh
S Chassis(su-config-bgp)->
```

Tear Down the BGP Connection

If neither route refresh nor soft reconfiguration are enabled, the only other method for dealing with soft refresh is to completely tear down the BGP connection. This approach is the most disruptive and requires reinitiation of the TCP and BGP connections and the reexchange of update messages. The reexchange of updates ensures that the new policy is applied to the routes. This approach is considered a last resort.

Use the **clear ip bgp** command, in any command mode, specifying the IP address of the peer to be torn down or an asterisk (*) for all peers on this local router. Specifying the **soft** keyword sends a route refresh, if supported by the effected peers.

The following example clears the BGP peer 1.2.3.4:

```
S Chassis(rw)->clear ip bgp 1.2.3.4
```

The following example clears all BGP peers and sends a route refresh message to each cleared peer.

```
S Chassis(rw)->clear ip bgp * soft
```

Community and Extended Community Attributes

BGP community and extended community attributes are optional, transitive BGP attributes that provide an administrative route labeling capability using route-maps. See Chapter 51, **Route-Map Manager Configuration** for details on configuring BGP route-maps.

Community Attribute

Communities provide a label to a set of prefixes that share one or more common properties. Upstream providers use the community label to apply routing policy using route-maps to the community member prefixes.

Route-maps support the community attribute for both match and set clauses. Use the set community route-map clause to label the route as a member of the specified community.

There are two means of specifying a community name:

- Specify the AS number this route belongs to, followed by a colon (:), followed by the community number
- Specify a predefined community value, as defined in RFC 1997 and RFC 3765, that are supported by the community field such as:
  - **NO_EXPORT** – Routes must not be advertised outside a BGP confederation boundary
  - **NO_ADVERTISE** – Routes must not be advertised to other BGP peers
  - **NO_EXPORT_SUBCONFED** – Routes must not be advertised to external BGP peers
  - **NO_PEER** – Routes must not be advertised across bilateral peer connections

One of the following actions must be specified:

- **remove-all** – Specifies that the action is to remove all communities from the route.
- **remove-specific** – Specifies that the action is to remove all matching communities from the route.
- **set-specific** – Specifies that the action is to append the specified community to the route.
- **remove-all-and-set** – Specifies that the action is to replace any existing communities in the route with the specified community

This example shows how to append the community value **100:100** to BGP routes matching prefix list named permit100:

```
S Chassis(su)->configure
S Chassis(su-config)->route-map bgp bgprm1 permit
S Chassis(su-config-route-map-bgp)->match prefix-list permit100
S Chassis(su-config-route-map-bgp)->set community 100:100 set-specific
S Chassis(su-config-route-map-bgp)->
```

This example shows how to append the well-known **NO_PEER** community (RFC-3765) to BGP routes matching prefix list named **permit200**:

```
S Chassis(su)->configure
S Chassis(su-config)->route-map bgp bgprm1 permit
S Chassis(su-config-route-map-bgp)->match prefix-list permit200
S Chassis(su-config-route-map-bgp)->set community NO_PEER set-specific
S Chassis(su-config-route-map-bgp)->
```

The route-map **match community** clause provides the ability to set route policy for packets that have been set with community name matching the community name specified in the match clause.

This example shows how to match a packet community to community **100** in AS **121**:

```
S Chassis(su)->configure
S Chassis(su-config)->route-map bgp bgprm1 permit
S Chassis(su-config-route-map-bgp)->match community 121:100
```

```
S Chassis(su-config-route-map-bgp)->
```

Extended Community Attribute

The Extended Community Attribute provides a mechanism for labeling information carried in BGP. It provides two important enhancements over the existing BGP Community Attribute:

- An extended range of use, ensuring that communities can be assigned to many different non-overlapping uses.
- The ability to specify a community type, providing structure for the community space. The S-Series supports extended community types:
  - IP, AS, and AS 4-octet route target
  - IP, AS, and AS 4-octet site-of-origin
  - OSPF domain ID, router ID, and route type

Use the set clause for the appropriate extended community type to label a route with the specified extended community. The extended community is labeled with a set value appropriate to the extended community. For example: The IPv4 route-target extended community requires a set value consisting of a valid IPv4 address followed by a colon (:) followed by a number in the range 0 - 65535. The firmware converts this set value into a hex identifier. The hex identifier for each set extended community is displayed in the **show ip bgp** output. When configuring an extended community match clause, use the **show ip bgp** command to determine the appropriate extended community identifier.

This example shows how to remove all matching extended communities from the AS route target **1001:10000** when all match clauses match for route-map **bgprm1**:

```
S Chassis(su)->configure
S Chassis(su-config)->route-map bgp bgprm1 permit
S Chassis(su-config-route-map-bgp)->match prefix-list permit200
S Chassis(su-config-route-map-bgp)->set extended-community as-route-target
1001:10000 remove-specific
S Chassis(su-config-route-map-bgp)->
```

This example shows how to match a packet against the extended community AS route target attribute **000203E9000186A0**:

```
S Chassis(su)->show ip bgp 1.0.0.0/8 detail
Route status codes: > - active

   Network                Next Hop             Rib MED Local-Pref Origin AS Path
>  1.0.0.0/8              192.168.121.112       U   0      100    IGP     121 2013

Community attributes in route:
121:100

Extended Community attributes in route:
Route Target: 1001:10000 (0x000203E9000186A0)
S Chassis(su)->configure
S Chassis(su-config)->route-map bgp bgprm1 permit
S Chassis(su-config-route-map-bgp)->match extended-community 000203E9000186A0
S Chassis(su-config-route-map-bgp)->
```

Route Flap Dampening

BGP route flap dampening is used to suppress routes that have been unstable due to misconfiguration, a rebooting router or module, or link flapping. For each route flap, a penalty is assessed to the route. When this route penalty reaches the cutoff threshold, the route is suppressed. The route penalty decays over time based upon a configured decay half-life value. Once the decaying penalty reaches the reuse threshold, or the hold-time timer has expired, the route is reinstalled into the routing table.

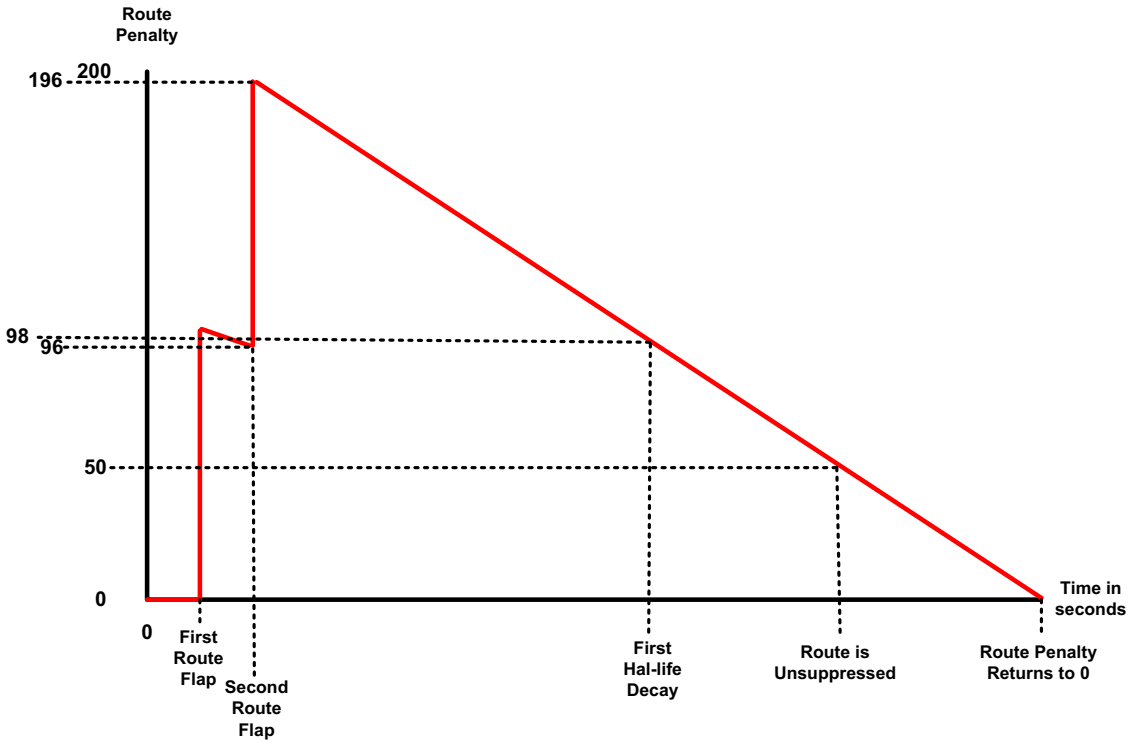
The route penalty starts with a value of **0**. With each route instability, the route penalty is increased by **100**. Route flap dampening provides for the following thresholds and timers that interact with the route penalty:

- **Cutoff threshold** – Specifies the route penalty value beyond which the route is suppressed.
- **Half-life reachable timer** – Specifies the time, in seconds, it takes a route penalty to decay to half of its current value, assuming the route is both reachable and remains stable during that period
- **Half-life unreachable timer** – Specifies the time, in seconds, it takes a route penalty to decay to half of its current value, assuming the route is both unreachable and remains stable during that period
- **Memory limit reachable timer** - Specifies the maximum time, in seconds, the history of a previous instability is retained in memory for a reachable route
- **Memory limit unreachable timer** - Specifies the maximum time, in seconds, the history of a previous instability is retained in memory for an unreachable route
- **Reuse threshold** – Specifies the route penalty below which a suppressed route is reused (unsuppressed).
- **Hold-time timer** – Specifies the maximum time a route can be suppressed regardless of its stability history

Route flap dampening configuration is contained in an administratively named flap table initially configured with default timer and threshold values. The flap table is assigned to a route-map. The route-map is assigned to the neighbor address to be monitored as an inbound route-map. When the first route instability occurs, the route penalty for this flap table is set to 100. This penalty immediately starts to decay at the rate set by the appropriate half-life timer. Upon the occurrence of a second route instability, the current route penalty is increased by 100. If the current penalty is now greater than the cutoff threshold (default penalty of 125) the route is suppressed. If no further instability occurs, the route penalty will eventually decay below the value of the reuse threshold, based upon the appropriate half-life timer setting. When the route penalty falls below the value of the reuse threshold, the route is unsuppressed. If route instability continues to occur, adding penalty points to the route penalty at a greater rate than the penalty can decay, it is possible for a route to stay suppressed until the hold-time timer expires. Once the hold-time timer expires, the route is unsuppressed, regardless of the current route penalty.

Figure 44-2 presents a default route flap dampening timing example. The route penalty starts at 0 and stays there until the first route flap. At the first route flap, route flap dampening sets the route penalty to 100. It immediately starts to decay based upon the default reachable half-life of 300 seconds. Approximately 20.5 seconds later, a second flap occurs. Route flap dampening adds 100 to the current route penalty of 96. The current route penalty is now greater than the cutoff value. The route is suppressed. No further route flaps occur.

Figure 44-2 Route Flap Dampening Timing



The route penalty immediately starts to decay. 300 seconds later the first reachable half-life is reached and the penalty is now 98. Because the route penalty is still greater than the reuse setting, route flap dampening continues to suppress the route. Route flap dampening unsuppresses the route when the route penalty decays to 50, the default reuse setting. Because no further route flaps occur, the route penalty continues to decay until it reaches 0.

The memory limit timers are used by route flap dampening for internal calculations. Half-life timers must be configured to a value less than the corresponding reachable or unreachable memory limit timer.

The flap table flap count or all flap statistics can be cleared. When clearing the flap count on a suppressed route, the route remains suppressed. When clearing all statistics on a suppressed route, the route is unsuppressed, regardless of the current route penalty value.

Use the **dampen-flap** command, in router configuration mode, to name the flap table and enter route flap dampening configuration mode.

The following example enters route flap dampening configuration mode for flap table **flap1**.

```
S Chassis(su-config)-> dampen-flap flap1
S Chassis(su-config-dampen-flap)->
```

Use the **cutoff** command, in route flap dampening configuration mode, to modify the route suppression threshold for this flap table context.

The following example modifies the cutoff threshold from the default value of 125 to **150** for the flap table **flap1**.

```
S Chassis(su-config)->dampen-flap flap1
S Chassis(su-config-dampen-flap)->cutoff 150
S Chassis(su-config-dampen-flap)->
```

Use the **half-life-reach** command, in route flap dampening configuration mode, to specify the time in seconds after which a reachable route's penalty value decays to half of its current value, assuming no further route instability for this route.

The following example configures the half life reachable value to **250** seconds for the **flap1** flap table.

```
S Chassis(su-config)->dampen-flap flap1
S Chassis(su-config-dampen-flap)->half-life-reach 250
S Chassis(su-config-dampen-flap)->
```

Use the **half-life-unreach** command, in route flap dampening configuration mode, to specify the time in seconds after which an unreachable route's penalty value decays to half of its current value, assuming no further route instability for this route.

The following example configures the half life unreachable value to **600** seconds for the **flap1** flap table.

```
S Chassis(su-config)->dampen-flap flap1
S Chassis(su-config-dampen-flap)->half-life-unreach 600
S Chassis(su-config-dampen-flap)->
```

Use the **hold-time** command, in route flap dampening configuration mode, to specify the maximum amount of time, in seconds, a route can be suppressed for this route table context.

The following example configures the hold-time to **1000** seconds for flap table **flap1**:

```
S Chassis(su-config)->dampen-flap flap1
S Chassis(su-config-dampen-flap)->hold-time 1000
```

Use the **reuse** command, in route flap dampening configuration mode, to specify the route penalty threshold under which a suppressed route is reused (unsuppressed).

The following example configures the reuse threshold to the route penalty value of **75** for flap table **flap1**:

```
S Chassis(su-config)->dampen-flap flap1
S Chassis(su-config-dampen-flap)->reuse 75
```

Use the **memory-limit-reach** command, in route flap dampening configuration mode, to specify the maximum time, in seconds, any memory of a previous instability is retained for this route table.

The following example configures the memory limit reachable value to **800** seconds for the **flap1** flap table.

```
S Chassis(su-config)->dampen-flap flap1
S Chassis(su-config-dampen-flap)->memory-limit-reach 800
S Chassis(su-config-dampen-flap)->
```

Use the **memory-limit-unreach** command, in route flap dampening configuration mode, to specify the maximum time, in seconds, any memory of a previous instability is retained for this route table.

The following example configures the memory limit unreachable value to **1600** seconds for the **flap1** flap table.

```
S Chassis(su-config)->dampen-flap flap1
S Chassis(su-config-dampen-flap)->memory-limit-unreach 1600
S Chassis(su-config-dampen-flap)->
```

See "Configuring Flap Dampening" on page 44-46 for a BGP flap dampening configuration example.

Graceful Restart

BGP graceful restart provides for the continued processing and packet forwarding of a router's data-forwarding plane even if the router control plane fails. With both a router and its peer graceful restart enabled, BGP exchanges the graceful restart capability (BGP code 64) in the initial BGP OPEN messages that establish the session.

When a failure takes place and the router restarts its BGP process, normally peer routers clear all routes associated with the restarting router. When graceful restart is enabled on a router, the peer router marks all routes as "stale" and continues to forward packets based on the expectation that the restarting router will reestablish the BGP session within a reasonable period of time. During the period of the restart, the restarting router continues to forward packets based upon routing state at the time of the restart. Peers refresh the restarting router with RIB updates.

When the restarting router opens the new BGP session, it will again send the BGP capability code 64 to its peers with flags set to let the peer router know that the BGP process has restarted. When the restarting router completes its restart and RIB update, it in turn updates its peers with any new updates.

Graceful restart reduces routing flaps, which stabilizes the network and reduces the consumption of control-plane resources.

BGP graceful restart timing is based upon four configurable intervals:

- **Restart defer interval** – Specifies the upper bound (in seconds) on the amount of time route selection will be deferred when BGP is restarting. The value specified should be large enough to provide all peers with enough time to send all their routes. The value must be greater than or equal to the restart timeout setting.
- **Restart timeout interval** – Specifies the interval which BGP advertises to its peers, in the OPEN message exchange, as the estimated time (in seconds) it will take for the BGP session to be reestablished after a restart. This can be used to speed up routing convergence by its peer in case the BGP speaker does not come back after a restart. Following a local restart, BGP will impose the restart timeout value as the upper bound on the length of time permitted for BGP to restart. If BGP fails to restart within the restart timeout period, route selection will commence immediately thereby overriding the restart defer timer.
- **Restart time interval** – Allows the peer to configure the maximum time (in seconds) it will wait for the restarting router to come back after a restart. This value will be used instead of the restart timeout value advertised in the OPEN message exchange, if the OPEN message value exceeds this restart timer value.
- **Stale path interval** – Configures the maximum time following a restart before removing stale routes from the peer. The stale path interval must be greater than or equal to the restart time.

Graceful restart must be enabled for the four configurable graceful restart timers to be relevant.

Use the **bgp graceful-restart** command, in BGP configuration mode, to enable graceful restart on the local router.

The following example enables graceful restart on router 151.1.1.9

```
S Chassis(su-config-bgp)->bgp router 65151
S Chassis(su-config-bgp)->bgp router-id 151.1.1.9
S Chassis(su-config-bgp)->bgp graceful-restart
S Chassis(su-config-bgp)->
```

Use the **bgp restart-defer** command, in BGP configuration mode, to configure the time to defer route selection after gracefully restarting.

The following example configures the defer timer to 150 seconds:

```
S Chassis(su-config)->router bgp 65151
```

```
S Chassis(su-config-bgp)->bgp router-id 151.1.1.9
S Chassis(su-config-bgp)->bgp restart-defer 150
S Chassis(su-config-bgp)->
```

Use the **bgp restart-time** command, in BGP configuration mode, to configure the maximum time to wait for a graceful restarting peer to come back up after a restart.

The following example configures the restart time to be 100 seconds:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 159.1.1.9
S Chassis(su-config-bgp)->bgp restart-time 100
S Chassis(su-config-bgp)->
```

Use the **bgp restart-timeout** command, in BGP configuration mode, to configure the estimated time advertised to peers in the OPEN message for the session to be reestablished after a graceful restart.

The following example configures the restart-timeout to be 150 seconds:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 159.1.1.9
S Chassis(su-config-bgp)->bgp restart-timeout 150
```

Use the **bgp stale-path-time** command, in BGP configuration mode, to configure the maximum time following a restart stale routes are allowed to persist on the peer.

The following example sets the stale-path-time to 150 seconds:

```
S Chassis(su-config)->router bgp 65151
S Chassis(su-config-bgp)->bgp router-id 159.1.1.9
S Chassis(su-config-bgp)->bgp stale-path-time 150
```

Configuring BGP

For information about...	Refer to page...
Configuring Graceful Restart	44-50
BGP Monitoring and Clearing	44-51

This section provides details for BGP configuration on S-Series products.

Table 44-2 lists BGP default values.

Table 44-2 Default BGP Parameters

Parameter	Description	Default Value
advertisement interval	The minimum interval in seconds between sending EBGP routing updates.	30 seconds
AS origination interval	The interval in seconds between successive update messages for route prefixes that originate in the local AS.	15 seconds
AS path limit	The upper limit on the AS path length when configuring a route.	1
connection retry interval	The amount of time between attempts to reestablish a connection to configured peers that are no longer available.	120 seconds
cutoff	The route suppression threshold used by flap dampening to determine when a flapping route should be suppressed.	125
distance (External to AS)	The priority given to external BGP routes relative to other protocols for the local router.	20
distance (Internal to AS)	The priority given to internal BGP routes relative to other protocols for the local router.	200
graceful restart	A BGP extension that provides for the continued processing and forwarding of packets by the data-forwarding plane even if the control plane fails.	disabled
half-life reachable	The time in seconds after which a reachable route's penalty value decays to half of its current value.	300 seconds
half-life unreachable	The time in seconds after which an unreachable route's penalty value decays to half of its current value.	900 seconds
hold-time (flap dampening)	The maximum amount of time a route can be suppressed.	900 seconds
hold-time (peering session negotiation)	The number of seconds to use when negotiating a peering session within a group.	90 seconds

Table 44-2 Default BGP Parameters (continued)

Parameter	Description	Default Value
idle hold interval	The interval in seconds between returning to the idle state and reinitiating a TCP connection for the peer.	15 seconds
keepalive timer	The interval between keepalive messages	30 seconds or one-third of the hold-time setting
local preference	The preference for this route over other possible routes on the local router.	100
maximum EBGP ECMP routes	The maximum number of external BGP ECMP routes on the local router.	1
maximum IBGP ECMP routes	The maximum number of internal BGP ECMP routes on the local router.	1
maximum ORF entries	The maximum number of outbound route filtering entries that will be accepted from the peer.	100000
maximum prefixes	The peak number of prefixes that BGP will accept for installation into the routing information base.	0 – unlimited
MED	The Multi-Exit Discriminator value when configuring a route.	0
memory limit reachable	The maximum time in seconds any memory of a previous instability is retained for a reachable route, given the route state is both unchanged and reachable.	700 seconds
memory limit unreachable	The maximum time in seconds any memory of a previous instability is retained for an unreachable route, given the route state is both unchanged and unreachable.	1800 seconds
open delay	The interval in seconds between the establishment of a TCP connection and the sending of an OPEN message to open a BGP session.	0 – no delay
origin	The value of the origin process attribute when configuring a route	0 – IGP
peer type	The type of peer or peer group	IBGP
peering type	Determines whether updates for prefixes containing the NOPEER community will be accepted by or sent to this neighbor.	unspecified

Table 44-2 Default BGP Parameters (continued)

Parameter	Description	Default Value
restart defer period	The time in seconds that route selection is deferred after a graceful restart.	120 seconds
restart time	The time in seconds to wait for a graceful restart capable peer to come back after a graceful restart.	120 seconds
restart timeout	The estimated time in seconds that is advertised to peers in the OPEN message for the session to be reestablished after a graceful restart.	120
reuse penalty	The route penalty value below which a suppressed route is reused.	50
route withdrawal interval	The interval between the advertisement and subsequent withdrawal of a route.	30 seconds
stale path time	The maximum time in seconds following a restart before removing stale routes from the peer.	120
time-to-live (TTL)	Specifies the number of hops for this neighbors TTL.	64

Configuring Basic BGP Router Parameters

The basic steps for configuring BGP are:

- Entering BGP configuration mode for this router, specifying the AS
- Setting a BGP-specific router ID
- Configuring BGP parameters

Procedure 44-1 describes how to configure Basic BGP.

Procedure 44-1 Configuring Basic BGP

Step	Task	Command(s)
1.	In configuration command mode, enable BGP and enter BGP configuration mode, specifying the autonomous system for this router.	**router bgp** *as-number*
2.	In BGP configuration mode, configure the BGP router-ID.	**bgp router-id** *router-id*
3.	Optionally enter address family mode and configure the address family indicator (AFI) for BGP peers.	**address-family** [**ipv4** | **ipv6**] [**unicast** | **multicast** | **both**] [**vrf** *vrf-name*]
4.	In BGP configuration mode, optionally configure an aggregate by combining the characteristics of multiple routes so that a single route is advertised.	**aggregate-address** *prefix/length* [**summary**] [**as-set**] [**summary-and-as-set**] [**suppress-map** *route-map*] [**advertise-map** *route-map*] [**attribute-map** *route-map*]

Procedure 44-1 Configuring Basic BGP (continued)

Step	Task	Command(s)
5.	In BGP configuration mode, optionally enable aggregation of routes independent of the route MED.	**bgp aggregate-med**
6.	In BGP configuration mode, optionally disable deterministic processing of MEDs.	**no bgp deterministic-med**
7.	In BGP configuration mode, optionally specify whether to compare MEDs when multiple routes with differing MEDs are received from peers in different Autonomous Systems.	**bgp always-compare-med**
8.	In BGP configuration mode, optionally modify the local-preference of advertised routes for the router.	**bgp local-pref** *pref-value*
9.	In BGP configuration mode, optionally modify the route selection priority given to internal or external BGP routes compared to other protocols for the router.	**bgp distance** {**internal** | **external**} *distance*
10.	In BGP configuration mode, optionally modify the maximum number of allowed external BGP ECMP routes.	**bgp max-ebgp-ecmp-routes** *value*
11.	In BGP configuration mode, optionally modify the maximum number of allowed internal BGP ECMP routes.	**bgp max-ibgp-ecmp-routes** *value*
12.	In BGP configuration mode, optionally disable BGP configuration on the router. Administratively disabled BGP configuration can be reenabled using the **enable** command.	**no enable**
13.	In BGP configuration mode, optionally disable message logging via the syslog mechanism whenever a BGP peer enters or leaves the established state.	**no log-up-down**
14.	In BGP configuration mode, optionally enable the sending of BGP traps when a peer transitions to Established or a lower state.	**bgp trap** {**peer-established** | **peer-degraded**}

Configuring BGP Route Injection

Routes can be injected into BGP by route redistribution or by specifying the network prefixes to import into BGP.

Procedure 44-2 describes how to inject routes into BGP.

Procedure 44-2 Configuring BGP Route Injection

Step	Task	Command(s)
1.	In BGP configuration mode, optionally specify network prefixes to be imported into BGP.	**network** *prefix/length* [**route-map** *name*][**aspath-limit** *limit*] [**origin** *code*] [**med** *value*] [**local-pref** *value*]

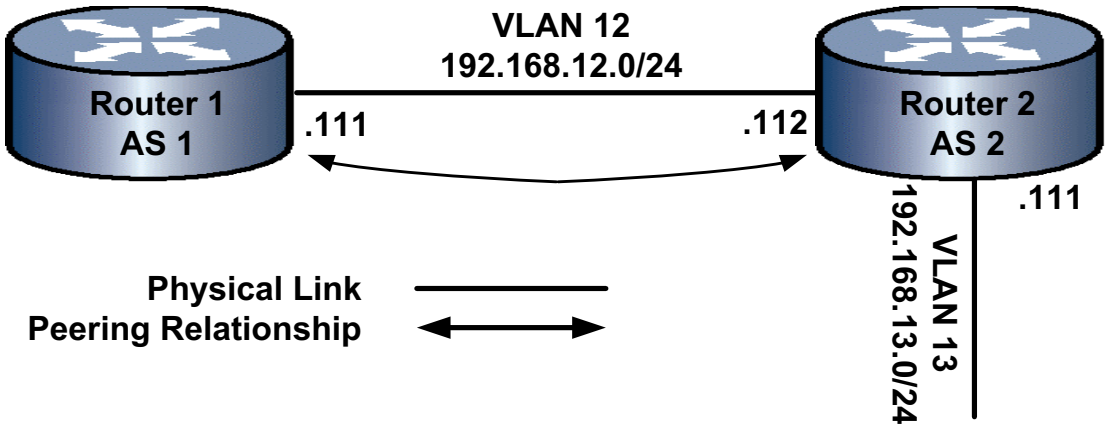
Procedure 44-2 Configuring BGP Route Injection (continued)

Step	Task	Command(s)
2.	In BGP configuration mode, optionally specify that connected routes are redistributed into BGP.	**redistribute connected** [**aspath-limit** *limit*] [**origin** *code*] [**med** *value*] [**local-pref** *value*] [**route-map** *name*]
3.	In BGP configuration mode, optionally specify that RIP routes are redistributed into BGP.	**redistribute rip** [**aspath-limit** *limit*] [**origin** *code*] [**med** *value*] [**local-pref** *value*] [**route-map** *name*]
4.	In BGP configuration mode, optionally specify that static routes are redistributed into BGP.	**redistribute static** [**aspath-limit** *limit*] [**origin** *code*] [**med** *value*] [**local-pref** *value*] [**route-map** *name*]
5.	In BGP configuration mode, optionally specify that OSPF routes are redistributed into BGP.	**redistribute ospf** *proc-id* [**aspath-limit** *limit*] [**origin** *code*] [**med** *value*] [**local-pref** *value*] [**route-map** *name*]

Configuring External BGP Basic Peering

The following example configures a basic External BGP peering as displayed in Figure 44-3 on page 44-27.

Figure 44-3 Basic EBGP Peering Topology



The example consists of two routers with a connection on subnet 192.168.12.0/24. One router belongs to AS 1. The other router belongs to AS 2. The example configuration consists of configuring, for each router:

- The connection interface and IP address
- The AS the router belongs to
- The router ID
- The connection neighbor IP address and remote AS
- The redistribution of static routes

Router 1

```
Router 1(rw)->configure
Router 1(rw-config)->interface vlan 12
```

```
Router 1(rw-config-intf-vlan.0.12)->ip address 192.168.12.111 255.255.255.0
Router 1(rw-config-intf-vlan.0.12)->no shutdown
Router 1(rw-config-intf-vlan.0.12)->exit

Router 1(rw)->configure
Router 1(rw-config)->router bgp 1
Router 1(su-config-bgp)->bgp router-id 1.1.1.1
Router 1(su-config-bgp)->neighbor 192.168.12.112 remote-as 2
Router 1(su-config-bgp)->redistribute static
Router 1(su-config-bgp)->
```

Router 2

```
Router 2(rw)->configure
Router 2(rw-config)->interface vlan 12
Router 2(rw-config-intf-vlan.0.12)->ip address 192.168.12.112 255.255.255.0
Router 2(rw-config-intf-vlan.0.12)->no shutdown
Router 2(rw-config-intf-vlan.0.12)->exit

Router 2(rw)->configure
Router 2(rw-config)->interface vlan 13
Router 2(rw-config-intf-vlan.0.13)->ip address 192.168.13.111 255.255.255.0
Router 2(rw-config-intf-vlan.0.13)->no shutdown
Router 2(rw-config-intf-vlan.0.13)->exit

Router 2(rw)->configure
Router 2(rw-config)->router bgp 2
Router 2(su-config-bgp)->bgp router-id 2.2.2.2
Router 2(su-config-bgp)->neighbor 192.168.12.111 remote-as 1
Router 2(su-config-bgp)->redistribute static
```

Router 2(su-config-bgp)->

Procedure 44-3 is a simple configuration intended for external BGP propagation.

Procedure 44-3 EBGP Basic Peering Configuration

Step	Task	Command
1.	In configuration mode, configure static routes between BGP routers to allow IP traffic transmission between remote routers.	**ip route** {*prefix mask* | *prefix*/*prefix-length*} {*ip-address* [**recursive** | **interface** *interface-name*] | **interface** *interface-name* | **vlan** *vlan-id* | **vrf** *egress-vrf* | **blackhole** | **reject**} [*distance*] [**tag** *tag-id*]
2.	In configuration mode, configure loopback and physical addresses and enter interface configuration mode.	**interface** {**vlan** *vlan-id* | **loopback** *loopback-id* | *interface-name*}
3.	In interface configuration mode, configure the IP address for the interface that serves as the BGP speaker.	**ip address** {*ip-address ip-mask* | *ip-address*/*prefixLength*} [**primary** | **secondary**]

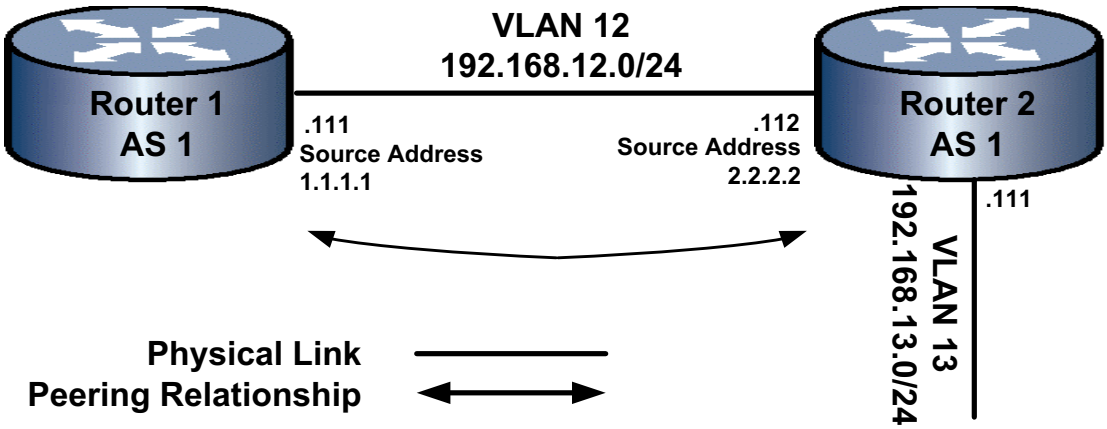
Procedure 44-3 EBGP Basic Peering Configuration (continued)

Step	Task	Command
4.	In configuration mode, specify an AS number for the router and enter BGP Configuration mode.	**router bgp** *as-number*
5.	In BGP configuration mode, configure a BGP-specific router ID to override the global router ID.	**bgp router-id** *router-id*
6.	In BGP configuration mode, configure the peer by identifying its IP address and AS.	**neighbor** *ip-address* **remote-as** *as-num* [**password** *password*]
7.	In BGP configuration mode, redistribute routes into BGP, optionally specifying a route-map. Supported *route-types* are connected, static, OSPF, and RIP.	**redistribute** *route-type* [**aspath-limit** *limit*] [**origin** *code*] [**med** *value*] [**local-pref** *value*] [**route-map** *name*]

Configuring Internal BGP Basic Peering

The following example configures a basic internal BGP peering as displayed in Figure 44-4 on page 44-29. The example configures the source IP address using the update source address feature to take advantage of the full mesh within an IBGP topology. See "Source IP Address Update to the Peer" on page 44-10 for an explanation of this feature.

Figure 44-4 Basic IBGP Peering Topology



The example consists of two routers with a connection on subnet 192.168.12.0/24. Because this is an internal BGP connection, both routers belong to the same AS. The example configuration consists of configuring, for each router:

- The loopback interface for the update source IP address
- The connection interface and IP address
- The AS the router belongs to
- The router ID
- The connection neighbor IP address (use update source IP address) and remote AS
- The connection neighbor IP address, specifying the update source IP address for this router
- The redistribution of static routes

Router 1

```
Router 1(rw)->configure
Router 1(rw-config)->interface loopback 1
Router 1(rw-config-intf-loop.0.1)->ip address 1.1.1.1 255.255.255.0
Router 1(rw-config-intf-loop.0.1)->no shutdown
Router 1(rw-config-intf-loop.0.1)->exit

Router 1(rw-config)->interface vlan 12
Router 1(rw-config-intf-vlan.0.12)->ip address 192.168.12.111 255.255.255.0
Router 1(rw-config-intf-vlan.0.12)->no shutdown
Router 1(rw-config-intf-vlan.0.12)->exit

Router 1(rw)->configure
Router 1(rw-config)->router bgp 1
Router 1(su-config-bgp)->bgp router-id 1.1.1.1
Router 1(su-config-bgp)->neighbor 2.2.2.2 remote-as 1
Router 1(su-config-bgp)->neighbor 2.2.2.2 update-source 1.1.1.1
Router 1(su-config-bgp)->redistribute static
Router 1(su-config-bgp)->
```

Router 2

```
Router 1(rw)->configure
Router 1(rw-config)->interface loopback 1
Router 1(rw-config-intf-loop.0.1)->ip address 2.2.2.2 255.255.255.0
Router 1(rw-config-intf-loop.0.1)->no shutdown
Router 1(rw-config-intf-loop.0.1)->exit

Router 2(rw-config)->interface vlan 12
Router 2(rw-config-intf-vlan.0.12)->ip address 192.168.12.112 255.255.255.0
Router 2(rw-config-intf-vlan.0.12)->no shutdown
Router 2(rw-config-intf-vlan.0.12)->exit

Router 2(rw)->configure
Router 2(rw-config)->interface vlan 13
Router 2(rw-config-intf-vlan.0.13)->ip address 192.168.13.111 255.255.255.0
Router 2(rw-config-intf-vlan.0.13)->no shutdown
Router 2(rw-config-intf-vlan.0.13)->exit

Router 2(rw)->configure
Router 2(rw-config)->router bgp 2
Router 2(su-config-bgp)->bgp router-id 2.2.2.2
Router 2(su-config-bgp)->neighbor 1.1.1.1 remote-as 1
Router 1(su-config-bgp)->neighbor 1.1.1.1 update-source 2.2.2.2
Router 2(su-config-bgp)->redistribute static
Router 2(su-config-bgp)->
```

Procedure 44-4 on page 44-31 is a simple configuration intended for internal BGP propagation.

Procedure 44-4 IBGP Basic Peering Configuration

Step	Task	Command
1.	In configuration mode, configure static routes between BGP routers to allow IP traffic transmission between remote routers.	**ip route** {*prefix mask* | *prefix*/*prefix-length*} {*ip-address* [**recursive** | **interface** *interface-name*] | **interface** *interface-name* | **vlan** *vlan-id* | **vrf** *egress-vrf* | **blackhole** | **reject**} [*distance*] [**tag** *tag-id*]
2.	In configuration mode, configure loopback and physical addresses and enter interface configuration mode.	**interface** {**vlan** *vlan-id* | **loopback** *loopback-id* | *interface-name*}
3.	In interface configuration mode, configure the IP address for the interface that serves as the BGP speaker.	**ip address** {*ip-address ip-mask* | *ip-address*/*prefixLength*} [**primary** | **secondary**]
4.	In configuration mode, specify an AS number for the router and enter BGP Configuration mode.	**router bgp** *as-number*
5.	In BGP configuration mode, configure a BGP-specific router ID to override the global router ID.	**bgp router-id** *router-id*
6.	In BGP configuration mode, specify an update source IP address assigned to a loopback interface, for this router, to be used as the source address instead of the default outgoing interface IP address.	**neighbor** *ip-address* **update-source** *source-addr*
7.	In BGP configuration mode, configure the peer by identifying its source IP address and AS.	**neighbor** *ip-address* **remote-as** *as-num*
8.	In BGP configuration mode, redistribute routes into BGP, optionally specifying a route-map. Supported *route-types* are connected, static, OSPF, and RIP.	**redistribute** *route-type* [**aspath-limit** *limit*] [**origin** *code*] [**med** *value*] [**local-pref** *value*] [**route-map** *name*]

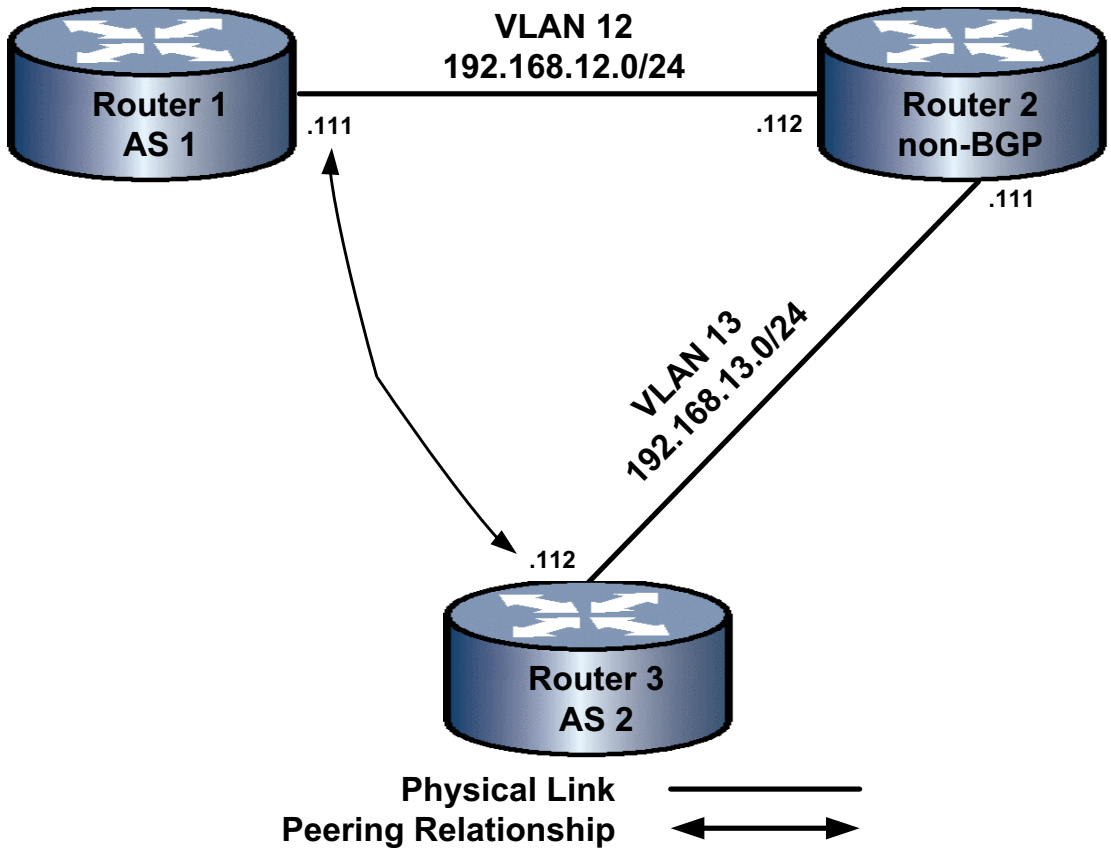
Configuring Multihop EBGP Basic Peering

An EBGP Multihop configuration is a topology where external BGP neighbors are not connected to the same subnet. Such neighbors are logically, but not physically connected. For example, BGP can be run between external neighbors across non-BGP routers.

Be aware that no IP traffic can pass to advertised BGP routes until an IGP protocol or static route is configured for those prefixes on the middle router.

See Figure 44-5 on page 44-31 for a presentation of the multihop EBGP basic peering configuration example topology.

Figure 44-5 EBGP Multihop Peering Topology



Router 1

```
Router 1(rw)->configure
Router 1(rw-config)->interface vlan 12
Router 1(rw-config-intf-vlan.0.12)->ip address 192.168.12.111 255.255.255.0
Router 1(rw-config-intf-vlan.0.12)->no shutdown
Router 1(rw-config-intf-vlan.0.12)->exit

Router 1(rw)->configure
Router 1(rw-config)->router bgp 1
Router 1(su-config-bgp)->bgp router-id 1.1.1.1
Router 1(su-config-bgp)->neighbor 192.168.13.111 remote-as 2
Router 1(su-config-bgp)->redistribute static
Router 1(su-config-bgp)->
```

Router 2

```
Router 2(rw)->configure
Router 2(rw-config)->interface vlan 12
Router 2(rw-config-intf-vlan.0.12)->ip address 192.168.12.112 255.255.255.0
Router 2(rw-config-intf-vlan.0.12)->no shutdown
Router 2(rw-config-intf-vlan.0.12)->exit

Router 2(rw-config)->interface vlan 13
```

```
Router 2(rw-config-intf-vlan.0.13)->ip address 192.168.13.111 255.255.255.0
Router 2(rw-config-intf-vlan.0.13)->no shutdown
Router 2(rw-config-intf-vlan.0.13)->exit
```

Router 3

```
Router 3(rw)->configure
Router 1(rw-config)->interface vlan 13
Router 1(rw-config-intf-vlan.0.13)->ip address 192.168.13.112 255.255.255.0
Router 1(rw-config-intf-vlan.0.13)->no shutdown
Router 1(rw-config-intf-vlan.0.13)->exit

Router 1(rw)->configure
Router 1(rw-config)->ip route 192.168.12.0/24 192.168.12.111 interface vlan 13
Router 1(rw-config)->router bgp 2
Router 1(su-config-bgp)->bgp router-id 2.2.2.2
Router 1(su-config-bgp)->neighbor 192.168.12.111 remote-as 1
Router 1(su-config-bgp)->redistribute static
Router 1(su-config-bgp)->
```

Procedure 44-5 is a simple configuration intended for multihop BGP propagation.

Procedure 44-5 Multihop BGP Basic Peering Configuration

Step	Task	Command
1.	In configuration mode, configure static routes between BGP routers to allow IP traffic transmission between remote routers.	**ip route** {*prefix mask* | *prefix*/*prefix-length*} {*ip-address* [**recursive** | **interface** *interface-name*] | **interface** *interface-name* | **vlan** *vlan-id* | **vrf** *egress-vrf* | **blackhole** | **reject**} [*distance*] [**tag** *tag-id*]
2.	In configuration mode, configure loopback and physical addresses and acquire interface configuration mode.	**interface** {**vlan** *vlan-id* | **loopback** *loopback-id* | *interface-name*}
3.	In interface configuration mode, configure the IP address for the interface that serves as the BGP speaker.	**ip address** {*ip-address ip-mask* | *ip-address*/*prefixLength*} [**primary** | **secondary**]
4.	In configuration mode, specify an AS number for the router and enter BGP Configuration mode.	**router bgp** *as-number*
5.	In BGP configuration mode, configure a BGP-specific router ID to override the global router ID.	**bgp router-id** *router-id*
6.	In BGP configuration mode, specify the network you want routes imported from and advertised to.	**network** *prefix/length* [**route-map** *name*][**aspath-limit** *limit*] [**origin** *code*] [**med** *value*] [**local-pref** *value*]
7.	In BGP configuration mode, configure the peer by identifying its IP address and AS.	**neighbor** *ip-address* **remote-as** *as-num*
8.	In BGP configuration mode, redistribute routes into BGP, optionally specifying a route-map. Supported *route-types* are connected, static, OSPF, and RIP.	**redistribute** *route-type* [**aspath-limit** *limit*] [**origin** *code*] [**med** *value*] [**local-pref** *value*] [**route-map** *name*]

Configuring BGP Neighbor Parameters

Table 44-3 describes the configuring of BGP neighbor parameters.

Table 44-3 BGP Neighbor Configuration

Task	Command
In BGP configuration mode, configure the remote AS for the peer.	**neighbor** *ip-address* **remote-as** *as-num*
In BGP configuration mode, optionally configure EBGP peer routes to not contain this neighbor's AS.	**neighbor** *ip-address* **ignore-leading-as**
In BGP configuration mode, optionally remove private autonomous system (AS) numbers from outbound updates to an external peer.	**neighbor** {*ip-address* | *groupID*} **remove-private-as**
In BGP configuration mode, optionally modify the minimum interval between the sending of EBGP routing updates.	**neighbor** *ip-address* **advertisement-interval** *interval*
In BGP configuration mode, optionally configure the conditional advertisement of routes for this neighbor.	**neighbor** {*ip-address* | *groupID*} **advertise-map** *adv-map* **non-exist-map** *non-exit-map*
In BGP configuration mode, optionally enable the inclusion of confederation information in the AS paths sent to this router's peers.	**neighbor** {*ip-address* | *groupID*} **aggregate-confed**
In BGP configuration mode, optionally modify the interval between successive update messages for route prefixes that originate in the local AS.	**neighbor** *ip-address* **as-origination-interval** *interval*
In BGP configuration mode, optionally enable checking to see if the next hop is the peer's address and do not send routes if it is.	**neighbor** *ip-address* **check-next-hop**
In BGP configuration mode, optionally clear all BGP counters for this peer.	**neighbor** *ip-address* **clear-counters**
In BGP configuration mode, optionally configure the specified neighbor as a member of the router's confederation.	**neighbor** *ip-address* **confed-member**
In BGP configuration mode, optionally modify the amount of time between attempts to reestablish a connection to configured peers that are no longer available.	**neighbor** *ip-address* **connect-retry-interval** *interval*
In BGP configuration mode, optionally force the advertisement of the default route regardless of whether the default route is present in the local routing table.	**neighbor** *ip-address* **default-originate** [**route-map** *name*]
In BGP configuration mode, explicitly enable a peer that has been administratively disabled. A configured peer is enabled by default.	**neighbor** *ip-address* **enable**
In BGP configuration mode, optionally modify the interval between returning to the idle state and reinitiating a TCP connection for this neighbor.	**neighbor** *ip-address* **idle-hold-interval** *interval*
In BGP configuration mode, optionally configure EBGP peer routes to not contain this neighbor's AS.	**neighbor** *ip-address* **ignore-leading-as**
In BGP configuration mode, optionally modify the maximum number of Outbound Route Filtering (ORF) entries that will be accepted from this neighbor.	**neighbor** *ip-address* **maximum-orf** *num*

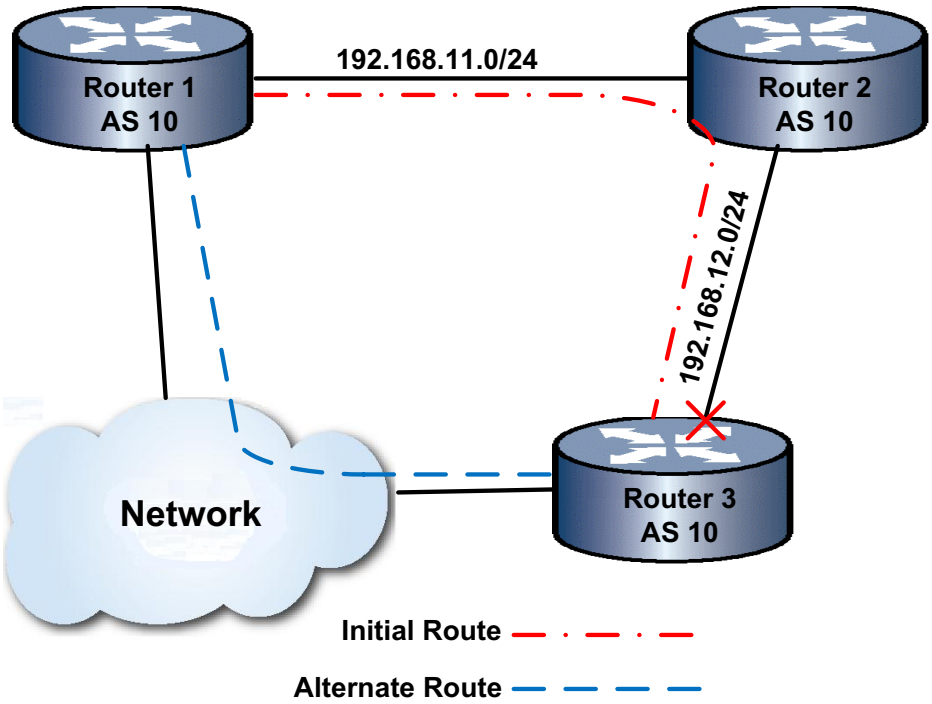
Table 44-3 BGP Neighbor Configuration (continued)

Task	Command
In BGP configuration mode, optionally modify the peak number of prefixes that BGP will accept for installation into the Routing Information Base (RIB).	**neighbor** *ip-address* **maximum-prefix** *num* [**warning-only**]
In BGP configuration mode, optionally configure BGP to always set the BGP next hop to the EBGP peer's address, overriding third-party next hops.	**neighbor** {*ip-address* | *groupID*} **next-hop-peer**
In BGP configuration mode, optionally set this neighbor's next hop as the router's own address on advertisement.	**neighbor** {*ip-address* | *groupID*} **next-hop-self**
In BGP configuration mode, optionally modify the interval between the establishment of a TCP connection and the sending of an OPEN message to open a BGP session.	**neighbor** *ip-address* **open-delay** *seconds*
In BGP configuration mode, optionally prevent the router from ever trying to open a BGP connection with the specified peer.	**neighbor** *ip-address* **passive**
In BGP configuration mode, optionally create a BGP peer group and add a peer group neighbor.	**neighbor** *groupID* **peer-group** **neighbor** *ip-address* **peer-group** *pgname*
In BGP configuration mode, optionally configure the peer type of the specified peer or group.	**neighbor** {*ip-prefix/length* | *groupID*} **peer-type** {**ibgp** | **ebgp** | **ebgp-confed**}
In BGP configuration mode, optionally configure whether updates for prefixes containing the NOPEER community will be accepted by or sent to this neighbor.	**neighbor** {*ip-address* | *groupID*} **peering-type** {**bilateral** | **unspecified**}
In BGP configuration mode, optionally specify a BGP route-map to be used for controlling the import or export of routes to and from the specified peer or group.	**neighbor** {*ip-address* | *groupID*} **route-map** *rm-name* {**in** | **out**}
In BGP configuration mode, optionally specify that the router will act as a route reflector for this neighbor.	**neighbor** *ip-address* **route-reflector-client**
In BGP configuration mode, optionally modify the interval between the advertisement and subsequent withdrawal of a route.	**neighbor** *ip-address* **route-withdraw-interval** *interval*
In BGP configuration mode, optionally enable soft-reconfiguration for a peer or peer group.	**neighbor** {*ip-address* | *groupID*} **soft-reconfiguration**
In BGP configuration mode, optionally modify holdtime and keepalive time values within a BGP peer.	**neighbor** *ip-address* **timers** *keepalive-value holdtime-value*
In BGP configuration mode, optionally modify the time to live (TTL) value.	**neighbor** *ip-address* **ttl** *ttl-num*
In BGP configuration mode, optionally specify the source IP address to be used in all TCP and BGP messages sent to the peer.	**neighbor** *ip-address* **update-source** *source-addr*

Configuring Source IP Address Update

Figure 44-6 displays an example of a Source IP address to remote peer configuration. Router 1 configures the loop-back source address 3.3.3.3 on Router 3 as its neighbor. Router 3 configures the loop-back source address 1.1.1.1 on Router 1 as its neighbor. When the outgoing interface on Router 3 for the initial route fails, because the loop-back source address 3.3.3.3 is still operational, the BGP message for this route is not discarded, but rather uses the new route after BGP reconverges.

Figure 44-6 Source IP Address to a Remote Peer



Router 1	Router 3
AS: 10	**AS:** 10
Source-address: 1.1.1.1	**Source-address:** 3.3.3.3
Neighbor: 3.3.3.3 remote-as 10	**Neighbor:** 1.1.1.1 remote-as 10

Router 1

```
S Chassis(su-config)->router bgp 10
S Chassis(su-config-bgp)->bgp router-id 1.1.1.1
S Chassis(su-config-bgp)->neighbor 3.3.3.3 remote-as 10
S Chassis(su-config-bgp)->neighbor 3.3.3.3 update-source 1.1.1.1
```

Router 3

```
S Chassis(su-config)->router bgp 10
S Chassis(su-config-bgp)->bgp router-id 3.3.3.3
S Chassis(su-config-bgp)->neighbor 1.1.1.1 remote-as 10
S Chassis(su-config-bgp)->neighbor 1.1.1.1 update-source 3.3.3.3
```

Procedure 44-6 describes how to configure the source IP address to the remote peer.

Procedure 44-6 Configuring Source IP Address to the Peer Update

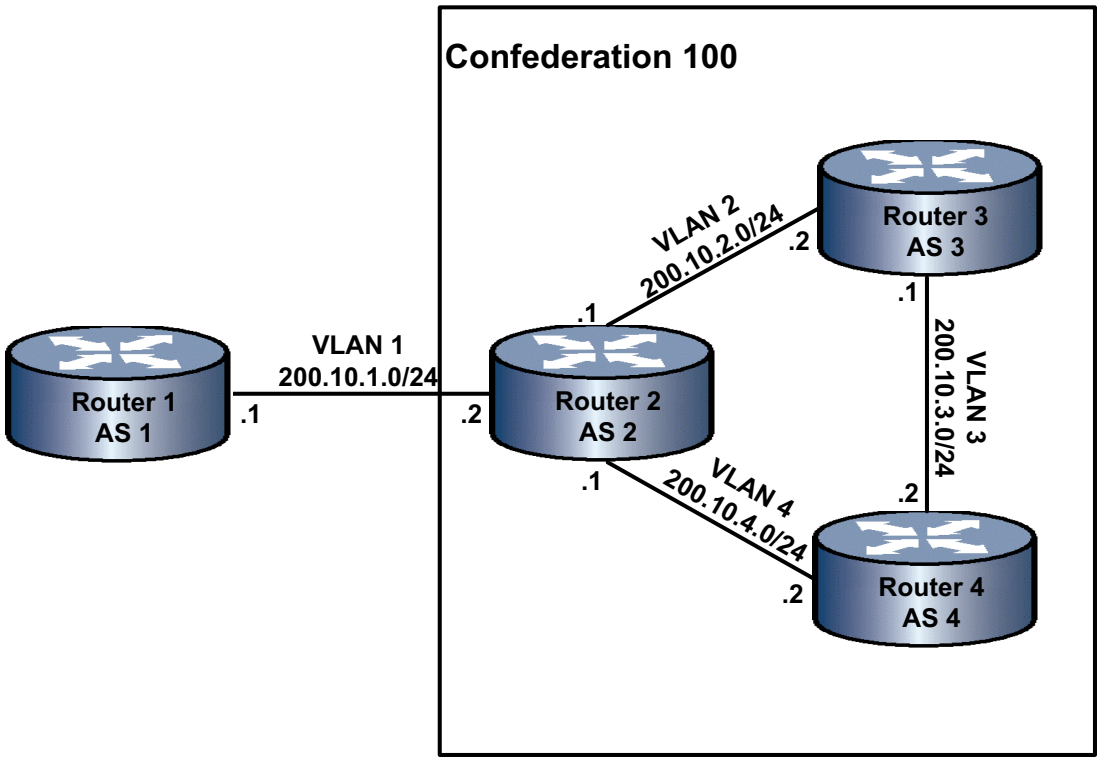
Step	Task	Command(s)
1.	In BGP configuration mode, specify the neighbor this update source IP address will be applied to.	**neighbor** *ip-address* **remote-as** *as-num*
2.	In BGP configuration mode, specify the update source IP address for this neighbor.	**neighbor** *ip-address* **update-source** *source-addr*

Configuring BGP Confederations

The following confederation configuration example presents a confederation of three routers, each belonging to different ASs, with a single EBGP connection to a router outside of the confederation. Router 1 sees Router 2 as belonging to AS (confederation) 100. Router 2 sees Router 1 as belonging to AS 1. Within the confederation, Router 2 - 4, see each other belonging to their respective ASs: 2, 3, and 4.

Figure 44-7 displays the topology for this BGP confederation example.

Figure 44-7 BGP Confederation Example Topology



Router 1

```
Router 1(rw)->configure
Router 1(rw-config)->interface vlan 1
Router 1(rw-config-intf-vlan.0.1)->ip address 200.10.1.1 255.255.255.0
Router 1(rw-config-intf-vlan.0.1)->no shutdown
Router 1(rw-config-intf-vlan.0.1)->exit

Router 1(rw)->configure
Router 1(rw-config)->router bgp 1
Router 1(su-config-bgp)->bgp router-id 1.1.1.1
Router 1(su-config-bgp)->neighbor 200.10.1.2 remote-as 100
Router 1(su-config-bgp)->redistribute static
Router 1(su-config-bgp)->
```

Router 2

```
Router 2(rw)->configure
Router 2(rw-config)->interface vlan 1
Router 2(rw-config-intf-vlan.0.1)->ip address 200.10.1.2 255.255.255.0
Router 2(rw-config-intf-vlan.0.1)->no shutdown
Router 2(rw-config-intf-vlan.0.1)->exit

Router 2(rw-config)->interface vlan 2
Router 2(rw-config-intf-vlan.0.2)->ip address 200.10.2.1 255.255.255.0
Router 2(rw-config-intf-vlan.0.2)->no shutdown
Router 2(rw-config-intf-vlan.0.2)->exit

Router 2(rw-config)->interface vlan 4
Router 2(rw-config-intf-vlan.0.4)->ip address 200.10.4.1 255.255.255.0
Router 2(rw-config-intf-vlan.0.4)->no shutdown
Router 2(rw-config-intf-vlan.0.4)->exit

Router 2(rw-config)->router bgp 2
Router 2(su-config-bgp)->bgp router-id 2.2.2.2
Router 2(su-config-bgp)->neighbor 200.10.1.1 remote-as 1
Router 2(su-config-bgp)->neighbor 200.10.2.2 remote-as 3
Router 2(su-config-bgp)->neighbor 200.10.4.2 remote-as 4
Router 2(su-config-bgp)->bgp confederation-id 100
Router 2(su-config-bgp)->neighbor 200.10.2.2 confed-member
Router 2(su-config-bgp)->neighbor 200.10.4.2 confed-member
Router 2(su-config-bgp)->redistribute static
Router 2(su-config-bgp)->
```

Router 3

```
Router 3(rw)->configure
Router 3(rw-config)->interface vlan 2
Router 3(rw-config-intf-vlan.0.2)->ip address 200.10.2.2 255.255.255.0
Router 3(rw-config-intf-vlan.0.2)->no shutdown
Router 3(rw-config-intf-vlan.0.2)->exit

Router 3(rw-config)->interface vlan 3
Router 3(rw-config-intf-vlan.0.3)->ip address 200.10.3.1 255.255.255.0
Router 3(rw-config-intf-vlan.0.3)->no shutdown
Router 3(rw-config-intf-vlan.0.3)->exit

Router 3(rw-config)->router bgp 3
Router 3(su-config-bgp)->bgp router-id 3.3.3.3
Router 3(su-config-bgp)->neighbor 200.10.2.1 remote-as 2
Router 3(su-config-bgp)->neighbor 200.10.3.2 remote-as 4
```

```
Router 3(su-config-bgp)->bgp confederation-id 100
Router 3(su-config-bgp)->neighbor 200.10.2.1 confed-member
Router 3(su-config-bgp)->neighbor 200.10.3.2 confed-member
Router 3(su-config-bgp)->redistribute static
Router 3(su-config-bgp)->
```

Router 4

```
Router 4(rw)->configure
Router 4(rw-config)->interface vlan 3
Router 4(rw-config-intf-vlan.0.3)->ip address 200.10.3.2 255.255.255.0
Router 4(rw-config-intf-vlan.0.3)->no shutdown
Router 4(rw-config-intf-vlan.0.3)->exit

Router 4(rw-config)->interface vlan 4
Router 4(rw-config-intf-vlan.0.4)->ip address 200.10.4.2 255.255.255.0
Router 4(rw-config-intf-vlan.0.4)->no shutdown
Router 4(rw-config-intf-vlan.0.4)->exit

Router 4(rw-config)->router bgp 4
Router 4(su-config-bgp)->bgp router-id 4.4.4.4
Router 4(su-config-bgp)->neighbor 200.10.3.1 remote-as 3
Router 4(su-config-bgp)->neighbor 200.10.4.1 remote-as 2
Router 4(su-config-bgp)->bgp confederation-id 100
Router 4(su-config-bgp)->neighbor 200.10.3.1 confed-member
Router 4(su-config-bgp)->neighbor 200.10.4.1 confed-member
Router 4(su-config-bgp)->redistribute static
Router 4(su-config-bgp)->
```

Procedure 44-7 describes how to configure BGP confederations.

Procedure 44-7 Configuring BGP Confederation

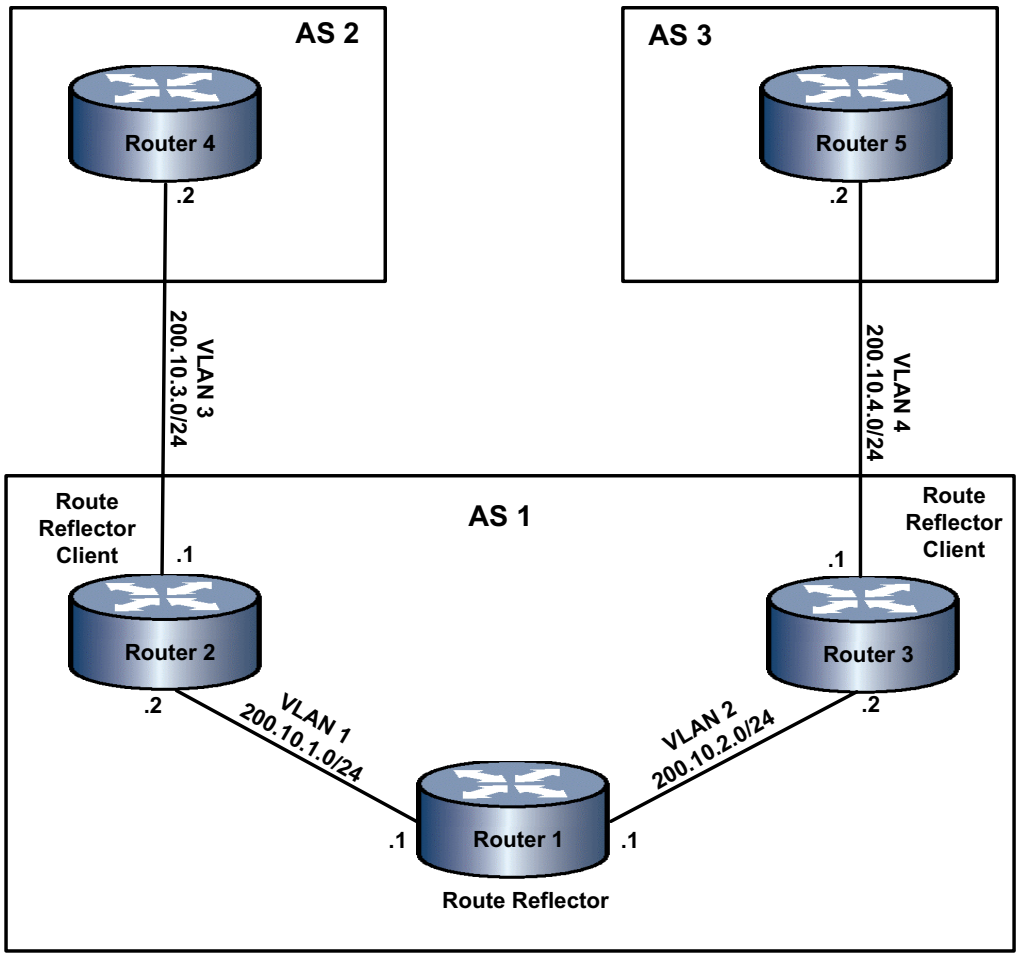
Step	Task	Command(s)
1.	In BGP configuration mode, specify the confederation this BGP router belongs to.	**bgp confederation identifier** *confed-id*
2.	In BGP configuration mode, configure the specified neighbor as a member of the router's confederation.	**neighbor** *ip-address* **confed-member**
3.	In BGP configuration mode, optionally enable the inclusion of confederation information in the AS paths sent to this router’s peers.	**neighbor** {*ip-address* | *peer-group*} **aggregate-confed**
4.	In BGP configuration mode, optionally enable BGP to drop AS-Paths with erroneous confederation segments.	**bgp strict-confeds**

Configuring Route Reflection

The following route reflection example configures a single route reflector with two clients all of which are members of AS 1. Router 1 is the route reflector. Routers 2 and 3 are route reflector clients. Router 2 has an EBGP connection to Router 4 of AS 2. Router 3 has an EBGP connection to Router 5 of AS 3. Routers 2 and 3 advertise all their routes to Router 1. Router 1 advertises routes learned from Router 2 to Router 3 and routes learned from Router 3 to Router 2.

Figure 44-8 displays the BGP route reflection topology for this example.

Figure 44-8 BGP Route Reflection Example Topology



Router 1

```
Router 1(rw)->configure
Router 1(rw-config)->interface vlan 1
Router 1(rw-config-intf-vlan.0.1)->ip address 200.10.1.1 255.255.255.0
Router 1(rw-config-intf-vlan.0.1)->no shutdown
Router 1(rw-config-intf-vlan.0.1)->exit

Router 1(rw)->configure
Router 1(rw-config)->interface vlan 2
Router 1(rw-config-intf-vlan.0.2)->ip address 200.10.2.1 255.255.255.0
Router 1(rw-config-intf-vlan.0.2)->no shutdown
```

```
Router 1(rw-config-intf-vlan.0.2)->exit

Router 1(rw)->configure
Router 1(rw-config)->router bgp 1
Router 1(su-config-bgp)->bgp router-id 1.1.1.1
Router 1(su-config-bgp)->bgp cluster-id 1.1.1.1
Router 1(su-config-bgp)->neighbor 200.10.1.2 remote-as 1
Router 1(su-config-bgp)->neighbor 200.10.1.2 route-reflector-client
Router 1(su-config-bgp)->neighbor 200.10.2.2 remote-as 1
Router 1(su-config-bgp)->neighbor 200.10.2.2 route-reflector-client
Router 1(su-config-bgp)->redistribute static
Router 1(su-config-bgp)->
```

Router 2

```
Router 2(rw)->configure
Router 2(rw-config)->interface vlan 1
Router 2(rw-config-intf-vlan.0.1)->ip address 200.10.1.2 255.255.255.0
Router 2(rw-config-intf-vlan.0.1)->no shutdown
Router 2(rw-config-intf-vlan.0.1)->exit

Router 2(rw)->configure
Router 2(rw-config)->interface vlan 3
Router 2(rw-config-intf-vlan.0.2)->ip address 200.10.3.1 255.255.255.0
Router 2(rw-config-intf-vlan.0.2)->no shutdown
Router 2(rw-config-intf-vlan.0.2)->exit

Router 2(rw)->configure
Router 2(rw-config)->router bgp 1
Router 2(su-config-bgp)->bgp router-id 2.2.2.2
Router 2(su-config-bgp)->neighbor 200.10.1.1 remote-as 1
Router 2(su-config-bgp)->neighbor 200.10.3.2 remote-as 2
Router 2(su-config-bgp)->redistribute static
Router 2(su-config-bgp)->
```

Router 3

```
Router 3(rw)->configure
Router 3(rw-config)->interface vlan 2
Router 3(rw-config-intf-vlan.0.2)->ip address 200.10.2.2 255.255.255.0
Router 3(rw-config-intf-vlan.0.2)->no shutdown
Router 3(rw-config-intf-vlan.0.2)->exit

Router 3(rw)->configure
Router 3(rw-config)->interface vlan 4
Router 3(rw-config-intf-vlan.0.4)->ip address 200.10.4.1 255.255.255.0
Router 3(rw-config-intf-vlan.0.4)->no shutdown
Router 3(rw-config-intf-vlan.0.4)->exit
```

```
Router 3(rw)->configure
Router 3(rw-config)->router bgp 1
Router 3(su-config-bgp)->bgp router-id 3.3.3.3
Router 3(su-config-bgp)->neighbor 200.10.2.1 remote-as 1
Router 3(su-config-bgp)->neighbor 200.10.4.2 remote-as 3
Router 3(su-config-bgp)->redistribute static
Router 3(su-config-bgp)->
```

Router 4

```
Router 4(rw)->configure
Router 4(rw-config)->interface vlan 3
Router 4(rw-config-intf-vlan.0.3)->ip address 200.10.3.2 255.255.255.0
Router 4(rw-config-intf-vlan.0.3)->no shutdown
Router 4(rw-config-intf-vlan.0.3)->exit

Router 4(rw)->configure
Router 4(rw-config)->router bgp 2
Router 4(su-config-bgp)->bgp router-id 4.4.4.4
Router 4(su-config-bgp)->neighbor 200.10.3.1 remote-as 1
Router 4(su-config-bgp)->redistribute static
Router 4(su-config-bgp)->
```

Router 5

```
Router 5(rw)->configure
Router 5(rw-config)->interface vlan 4
Router 5(rw-config-intf-vlan.0.4)->ip address 200.10.4.2 255.255.255.0
Router 5(rw-config-intf-vlan.0.4)->no shutdown
Router 5(rw-config-intf-vlan.0.4)->exit

Router 5(rw)->configure
Router 5(rw-config)->router bgp 3
Router 5(su-config-bgp)->bgp router-id 5.5.5.5
Router 5(su-config-bgp)->neighbor 200.10.4.1 remote-as 1
Router 5(su-config-bgp)->redistribute static
Router 5(su-config-bgp)->
```

Procedure 44-8 describes how to configure BGP route reflection.

Procedure 44-8 Configuring BGP Route Reflection

Step	Task	Command(s)
1.	In BGP configuration mode, specify that the router will act as a route reflector for the specified neighbor.	**neighbor** *ip-address* **route-reflector-client**

Procedure 44-8 Configuring BGP Route Reflection (continued)

Step	Task	Command(s)
2.	In BGP configuration mode, specify the route reflection cluster ID for the cluster the route reflector belongs to. This value defaults to the route reflector router ID if only a single route reflector is configured for the cluster, otherwise a cluster ID must be specified.	**bgp cluster-id** *router-id*

Configuring Outbound Route Filtering (ORF)

Table 44-4 describes how to configure BGP Outbound Route Filtering.

Table 44-4 Configuring BGP Outbound Route Filtering

Task	Command(s)
In BGP configuration mode, optionally specify whether the ORF capability for community filtering is to be sent to the BGP peer, received from the BGP peer, or both.	**bgp orf** {**ipv4** | **ipv6**} **unicast comm-filter** {**send** | **receive** | **both**}
In BGP configuration mode, optionally specify whether the ORF capability for extended community filtering is to be sent to the BGP peer, received from the BGP peer, or both.	**bgp orf** {**ipv4** | **ipv6**} **unicast extcomm-filter** {**send** | **receive** | **both**}
In BGP configuration mode, optionally specify whether the Outbound Route Filtering (ORF) capability for prefix filtering is to be sent to the BGP peer, received from the BGP peer, or both.	**bgp orf** {**ipv4** | **ipv6**} **unicast prefix-filter** {**send** | **receive** | **both**}

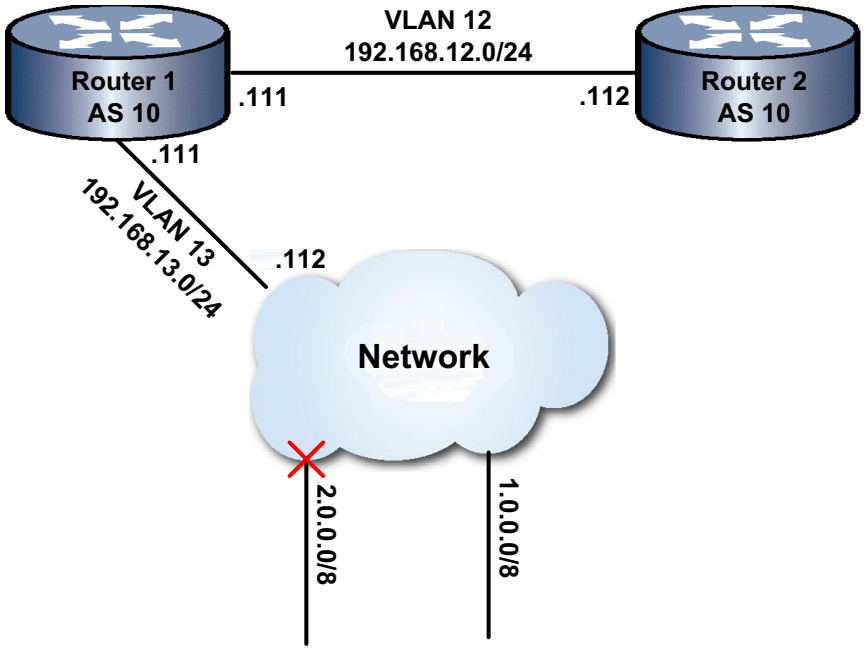
Configuring Conditional Advertisement

The conditional advertisement feature allows a service provider to advertise certain routes to a preferred subnet under normal operational conditions, while maintaining the ability to move its traffic to an alternative subnet should its preferred routes fail.

The following example configures two static routes on Router 1. One will be advertised to Router 2 under normal operational conditions, the other will only be advertised if the first route no longer exists. A prefix list is created for each static route destination prefix. The destination prefix to be advertised under normal operational conditions is assigned to the non-exist-map route-map. The destination prefix to be advertised only if the advertised prefix fails is assigned to the advertise route-map.

Figure 44-9 on page 44-43 presents the conditional advertisement example topology.

Figure 44-9 BGP Conditional Advertisement Example Topology



For Router 1

- Configure two static routes with next hops to **192.168.13.112** on **VLAN 13** to destination prefixes **1.0.0.0/8** and **2.0.0.0/8**
- Configure the non-exist map prefix-list named **non-exist-list1** and assign it to BGP route-map **non-exist-map1**, specifying prefix **2.0.0.0/8** as the non-exist map prefix
- Configure the advertise map prefix-list named **adv-list1** and assign it to BGP route-map **adv-map1**, specifying prefix **1.0.0.0/8** as the advertised prefix
- Configure the router for AS **10** with a router ID of **1.1.1.1**
- Configure IP address **192.168.12.112** as the peer
- Configure the BGP advertise map for neighbor **192.168.12.112** and assign **adv-map1** as the advertise map route-map and **non-exist-map1** as the non-exist map route-map

```
Router 1(su)->configure
Router 1(su-config)->ip route 1.0.0.0/8 192.168.13.112 interface vlan.0.13
Router 1(su-config)->ip route 2.0.0.0/8 192.168.13.112 interface vlan.0.13
Router 1(su-config)->ip prefix-list adv-list1 permit seq 1 1.0.0.0/8
Router 1(su-config)->route-map bgp adv-map1
Router 1(su-config-route-map-bgp)->match prefix-list adv-list1
Router 1(su-config-route-map-bgp)->exit
Router 1(su-config)->ip prefix-list non-exist-list1 permit seq 1 2.0.0.0/8
Router 1(su-config)->route-map bgp non-exist-map1
Router 1(su-config-route-map-bgp)->match prefix-list non-exist-list1
Router 1(su-config-route-map-bgp)->exit
Router 1(su-config)->router bgp 10
Router 1(su-config-bgp)->bgp router-id 1.1.1.1
```

```
Router 1(su-config-bgp)->neighbor 192.168.12.112 remote-as 10
Router 1(su-config-bgp)->neighbor 192.168.12.112 advertise-map adv-map1
non-exist-map non-exist-map1
Router 1(su-config-bgp)->
```

For Router 2:

- Configure a static route with the next hop **192.168.12.111** on **VLAN 12** to destination prefix **192.168.13.0/24**
- Configure Router 2 for AS **10** with a router ID of **2.2.2.2**
- Configure IP address **192.168.12.111** as the peer

```
Router 2(su)->configure
Router 2(su-config)->ip route 192.168.13.0/24 192.168.12.111 interface vlan.0.12
Router 2(su-config)->router bgp 10
Router 2(su-config-bgp)->bgp router-id 2.2.2.2
Router 2(su-config-bgp)->neighbor 192.168.12.111 remote-as 10
Router 2(su-config-bgp)->Verifying the Configuration
```

If advertised map configuration is not applied, the Router 1 advertised routes output for 192.168.12.112 would display both configured static routes as being advertised to Router 2:

```
Router 1(su)->show ip bgp peer 192.168.12.112 advertised-routes
Route status codes: adv - advertised, sup - suppressed, pw - pending w/drawal, wd
- w/drawn
Route aggregation codes:
1 - Route is not aggregating or aggregated
2 - Route is aggregating
3 - Route is unsuppressed aggregated
4 - Route is suppressed aggregated

Stat Aggr   Network             Next Hop        Rib MED Local-Pref Origin AS Path
adv  1     1.0.0.0             192.168.13.112   U   0      100    Inc
adv  1     2.0.0.0             192.168.13.112   U   0      100    Inc
```

With the advertised map configuration applied, only network 2.0.0.0 displays:

```
Router 1(su)->show ip bgp peer 192.168.12.112 advertised-routes
....
Stat Aggr   Network             Next Hop        Rib MED Local-Pref Origin AS Path
adv  1      2.0.0.0             192.168.13.112   U   0      100    Inc
```

Should static route 2.0.0.0/8 be withdrawn, network 1.0.0.0 is advertised and network 2.0.0.0 displays as withdrawn:

```
Router 1(su-config)->no ip route 2.0.0.0/8 192.168.13.112 interface vlan.0.13 1
Router 1(su-config)->show ip bgp peer 192.168.12.112 advertised-routes
....
Stat Aggr   Network             Next Hop        Rib MED Local-Pref Origin AS Path
adv  1      1.0.0.0             192.168.13.112   U   0      100    Inc
wd   1      2.0.0.0             192.168.13.112   U   0      100    Inc
```

Procedure 44-9 describes how to configure BGP conditional route advertisement.

Procedure 44-9 Configuring BGP Conditional Route Advertisement

Step	Task	Command(s)
1.	In router configuration mode, configure a prefix list for both the advertise and non-exist prefixes to be matched in the appropriate route-maps.	**ip prefix-list** *name* [**seq** *seq-value*] {**deny** | **permit**} {*prefix*/*length*} [source-address] [next-hop] [**ge** length] [**le** length] [**nlri**]
2.	In router configuration mode, create both an advertise and non-exist route-map.	**route-map bgp** *name* [**permit** | **deny**] [*sequence-number*]
3.	In BGP route-map configuration mode, add match statements containing the configured prefix lists to the appropriate BGP route-map.	**match prefix-list** *prefix-list*
4.	In BGP configuration mode, specify the advertise and non-exist maps to be applied to this neighbor.	**neighbor** {*ip-address* | *groupID*} **advertise-map** *adv-map* **non-exist-map** *non-exist-map*

Configuring BGP Soft Reset

Table 44-5 describes how to configure BGP soft reset.

Table 44-5 Configuring BGP Soft Reset

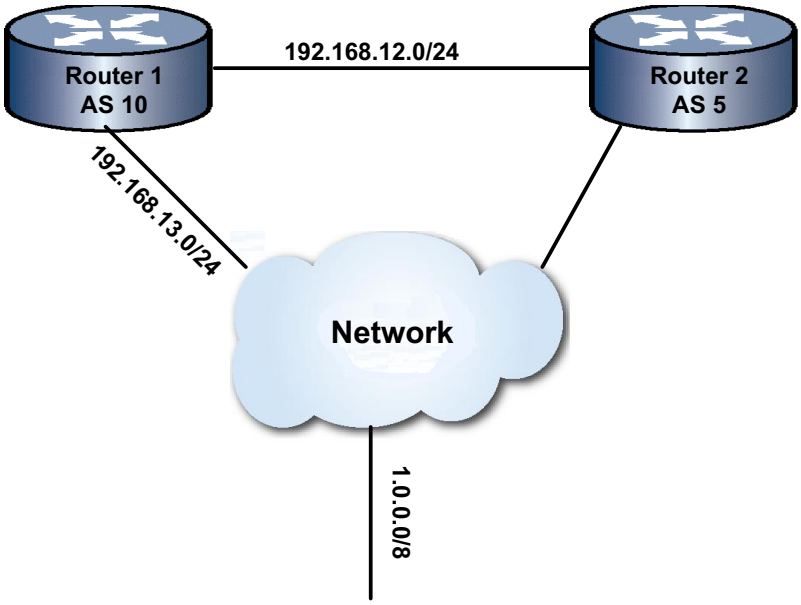
Task	Command(s)
In BGP configuration mode, optionally enable BGP soft reconfiguration for a peer or peer group.	**neighbor** {*ip-address* | *groupID*} **soft-reconfiguration**
In BGP configuration mode, optionally disable the automatic sending of route-refresh messages on inbound policy changes. Automatic route refresh is enabled by default. If soft reconfiguration is enabled, route refreshes are not sent.	**no bgp automatic-route-refresh**
In any command mode, optionally tear down one or all BGP connections. Optionally specify the **soft** option to perform a route refresh.	**clear ip bgp** {*peer-address* | ***}** [**soft**]

Configuring Flap Dampening

This section presents a flap dampening configuration example. The example configures a static route on Router 1 that is redistributed to Router 2. On Router 2 a flap table is configured and applied to an inbound route-map that will monitor the link between Router 2 and Router 1. Using show output, the example tracks the changes in route penalty over two flaps of the route that result in the route being suppressed. After waiting for the 5 minute hold-time to expire, the show output displays the route as unsuppressed.

Figure 44-10 on page 44-46 presents the route flap dampening example configuration.

Figure 44-10 Route Flap Dampening Example Configuration



Router 1	Router 2
AS: 10	**AS:** 5
Static Route:	**Neighbor:** 192.168.12.111 remote-as 10
Destination Prefix: 1.0.0.0/8	**Flap table:**
Next Hop: 192.168.13.112	**Name:** flap1
Neighbor: 192.168.12.113 remote-as 5	**Cutoff:** 150
Redistribute: static	**Reuse:** 75
	Hold-time: 300 seconds
	Route-map: map1

Router 1

Router 1 is configured by:

- Configuring a static route on the 192.168.13.0/24 subnet with a next hop of **192.168.13.112** and a destination address of **1.0.0.0/8** on Router 1 (AS **10**)
- Setting the router ID to **1.2.3.4**
- Configuring Router 1 with a BGP neighbor **192.168.12.112** (AS **5**)
- Redistributing the static route to Router 2

```
S Chassis(su)->configure
S Chassis(su-config)->ip route 1.0.0.0/8 192.168.13.112 interface vlan.0.13
S Chassis(su-config)->bgp router 10
S Chassis(su-config-bgp)->bgp router-id 1.2.3.4
S Chassis(su-config-bgp)->neighbor 192.168.12.112 remote-as 5
S Chassis(su-config-bgp)->redistribute static
```

Router 2

Router 1 is configured by:

- Configuring the **flap1** flap table on Router 2 with:
  - **cutoff** set to **150**
  - **reuse** set to **75**
  - **hold-time** set to **300**
- Applying the **flap1** flap table to the **map1** BGP route-map
- Configuring Router 2 with a BGP neighbor **192.168.12.111** (AS **10**)
- Applying the **map1** inbound route-map to neighbor **192.168.12.111**
- Clear the BGP session to initiate a readvertisement of the route from Router 1 to Router 2
- Display the route and route flap dampening details

```
S Chassis(su)->configure
S Chassis(su-config)->dampen-flap flap1
S Chassis(su-config-dampen-flap)->cutoff 150
S Chassis(su-config-dampen-flap)->reuse 75
S Chassis(su-config-dampen-flap)->hold-time 300
S Chassis(su-config-dampen-flap)->exit
S Chassis(su-config)->route-map bgp map1 permit 10
S Chassis(su-config-route-map-bgp)->set flap-table flap1
S Chassis(su-config-route-map-bgp)->exit
S Chassis(su-config)->router bgp 5
S Chassis(su-config-bgp)->bgp router-id 2.3.4.5
S Chassis(su-config-bgp)->neighbor 192.168.12.111 remote-as 10
S Chassis(su-config-bgp)->neighbor 192.168.12.111 route-map map1 in
S Chassis(su-config-bgp)->exit
S Chassis(su-config)->
```

The following displays route dampening statistics for network 1.0.0.0/8 prior to any instability:

```
S Chassis(su-config)->show ip bgp 1.0.0.0/8 detail
Route status codes: > - active

    Network                Next Hop            Rib MED Local-Pref Origin AS Path
>   1.0.0.0/8                192.168.12.111        U   0      100    Inc     10

Community attributes in route:

Extended Community attributes in route:

Route Flap Dampening configuration file name: flap1
Is route suppressed? No
Flap penalty: 0, Flap Count 0, Flap time remaining 0 seconds
S Chassis(su-config)->
```

If you were to enter the show command immediately after the first flap, the route flap dampening statistics would show:

```
S Chassis(su-config)->show ip bgp 1.0.0.0/8 detail
Route status codes: > - active
...
Route Flap Dampening configuration file name: flap1
Is route suppressed? No
Flap penalty: 99, Flap Count 1, Flap time remaining 0 seconds
```

The route remains unsuppressed because the route penalty has not exceeded the cutoff value. The flap penalty has decayed by 1 point from 100 to 99 since the route flap occurred. The flap count is 1 and because the route has not yet been suppressed, flap time is set to 0 seconds.

Entering the show command immediately after the second route flap, the route flap dampening statistics would show:

```
S Chassis(su-config)->show ip bgp 1.0.0.0/8 detail
Route status codes: > - active
...
Route Flap Dampening configuration file name: flap1
Is route suppressed? Yes
Flap penalty: 185, Flap Count 2, Flap time remaining 295 seconds
```

Because the flap penalty exceeds the cutoff setting, route flap dampening has suppressed the route. Flap time now displays the amount of seconds remaining before the hold-time is reached.

After waiting the 5 minute hold-time, the display shows that the route has been unsuppressed. If the hold-time was set to the default value of 900 seconds, and no other router flaps occurred for this route, the route penalty would have decayed to the point that the route would have been unsuppressed when the route penalty reached the reuse setting of 75.

```
S Chassis(su-config)->show ip bgp 1.0.0.0/8 detail
Route status codes: > - active
...
Route Flap Dampening configuration file name: flap1
Is route suppressed? No
Flap penalty: 86, Flap Count 2, Flap time remaining 0 seconds
```

Procedure 44-10 describes how to configure the BGP flap dampening.

Procedure 44-10 Configuring BGP Flap Dampening

Step	Task	Command(s)
1.	In router configuration mode, enter the dampen flap command mode for the named dampen flap table.	**dampen-flap** *name*
2.	In BGP flap dampening configuration mode, optionally modify the route suppression threshold.	**cutoff** *threshold*
3.	In BGP flap dampening configuration mode, optionally modify the time in seconds after which a reachable route's penalty value decays to half of its current value.	**half-life-reach** *seconds*

Procedure 44-10 Configuring BGP Flap Dampening

Step	Task	Command(s)
4.	In BGP flap dampening configuration mode, optionally modify the time in seconds after which an unreachable route's penalty value decays to half its current value.	**half-life-unreach** *seconds*
5.	In BGP flap dampening configuration mode, optionally modify the maximum time a route can be suppressed.	**hold-time** *seconds*
6.	In BGP flap dampening configuration mode, optionally modify the decay memory limit for reachable routes.	**memory-limit-reach** *seconds*
7.	In BGP flap dampening configuration mode, optionally modify the decay memory limit for unreachable routes.	**memory-limit-unreach** *seconds*
8.	In BGP flap dampening configuration mode, optionally modify the route penalty value below which a suppressed route is reused.	**reuse** *value*

Configuring Graceful Restart

Procedure 44-11 describes how to configure graceful restart.

Procedure 44-11 Configuring Graceful Restart

Step	Task	Command(s)
1.	In BGP configuration mode, optionally enable graceful restart on the router.	**bgp graceful-restart**
2.	In BGP configuration mode, optionally modify the time to defer route selection after graceful restarting.	**bgp restart-defer** *time-seconds*
3.	In BGP configuration mode, optionally modify the maximum time to wait for a graceful restart capable peer to come back after a restart.	**bgp restart-time** *time-seconds*
4.	In BGP configuration mode, optionally modify the estimated time advertised to peers in the OPEN message for the session to be reestablished after a graceful restart.	**bgp restart-timeout** *time-seconds*
5.	In BGP configuration mode, optionally modify the maximum time following a restart before removing stale routes from the peer.	**bgp stale-path-time** *time-seconds*

BGP Monitoring and Clearing

Table 44-6 describes how to monitor and clear BGP configuration.

Table 44-6 Monitoring and Clearing BGP Configuration

Task	Command(s)
In any command mode, optionally reset BGP peering sessions and optionally send route refresh requests.	**clear ip bgp** {*peer* | ***}** [**soft**]
In any command mode, optionally clear all route-flap statistics and state for the specified route-prefix.	**clear ip bgp flap-all-stats** *ip-prefix*/*length*
In any command mode, optionally clear the route-flap count for the specified route-prefix.	**clear ip bgp flap-count** *ip-prefix/length*
In any command mode, optionally display information about BGP routes installed in the BGP routing information base (RIB).	**show ip bgp** [*ip-address*] [*ip-prefix/mask*] [**longer-prefixes**] [**detail**] [**peer** *ip-addr* {**all-received-routes** | **received-routes** | **advertised-routes**}]
In any command mode, optionally display dampened routes information.	**show ip bgp dampened-routes**
In any command mode, optionally display information for BGP groups.	**show ip bgp groups**
In any command mode, optionally display information about the state of BGP's IPv4 peering sessions.	**show ip bgp neighbors** [*ip-address*]
In any command mode, optionally display a summary of the BGP configuration.	**show ip bgp summary**

Terms and Definitions

Table 44-7 lists terms and definitions used in this BGP configuration discussion.

Table 44-7 BGP Terms and Definitions

Term	Definition
4-octet AS numbers	A BGP extension, defined in RFC 4893, that allows for the encoding of 4-octet AS numbers.
aggregation	A BGP feature that provides for the aggregating of one or more specific routes into a single aggregate route, if a more specific route of the aggregate route exists in the BGP routing table.
AS	A set of routers under a single administration referred to as an autonomous system.
AS path	A BGP path attribute, used for loop detection, that provides a list of the AS numbers the route traverses.
BGP	The Border Gateway Protocol (BGP), defined in RFC 4271, that is the standard Exterior Gateway Protocol (EGP) for routing between administrative domains.
conditional advertisement	Conditional BGP announcements that are sent in addition to normal announcements, when a route specified in the configured advertise map does not exist in the configured non-exist map

Table 44-7 BGP Terms and Definitions (continued)

Term	Definition
confederation	An extension to BGP, defined in RFC 3065, which may be used to create a confederation of autonomous systems that is represented as a single autonomous system to BGP peers external to the confederation, thereby removing the "full mesh" requirement, aiding in policy administration, and reducing the management complexity of maintaining a large autonomous system.
graceful restart	A capability, defined in RFC 4724, that provides for the continued processing of the data-forwarding plane of a router should the control plane fail.
MED	The Multi-Exit Discriminator attribute used by external neighbors in the selection of the preferred path into an autonomous system (AS) that has multiple entry points.
multi-protocol BGP extensions	A set of BGP extensions, defined in RFC 2858, that enable BGP to carry routing information for multiple Network Layer protocols such as IPv6 and IPX.
outbound route filtering	A BGP capability, defined in RFC 5291, that allows a BGP speaker to send to its BGP peer a set of Outbound Route Filters (ORFs), which the peer applies in addition to its locally configured outbound filters (if any), to constrain its outbound routing updates to the speaker
peer group	A group of neighbors that share the same BGP attributes.
peer-group	A BGP capability that provides for the grouping of peers for purposes of policy, such that the group policy takes precedence for route export, and the peer configured policy takes precedence for route import.
route flap dampening	A BGP capability, defined in RFC 2439, that treats routes that are being announced and withdrawn at a rapid rate as unreachable, based upon a route penalty for each route withdrawal, and reachable again based upon a configurable decay over time of that route penalty.
route reflector	A BGP capability, defined in RFC 4456, that allows a BGP speaker (route reflector) to advertise IBGP learned routes to certain IBGP peers, relieving the scaling issue associated with a fully meshed AS.
route refresh	A BGP capability, defined in RFC 2918, which allows for the dynamic exchange of route refresh requests between BGP speakers and the subsequent re-advertisement of the respective Adj-RIB-Out.
soft reconfiguration	A BGP capability that speeds up the route installation process when an inbound policy change occurs by keeping a local copy of the routes for the specified peer or group.
TCP	Transmission Control Protocol.
TCP/MD5 authentication	A TCP extension to BGP security that defines, in RFC 2385, a TCP option for carrying an MD5 digest in a TCP segment and acts like a signature for that segment, incorporating information known only to the connection end points.

45

Network Address Translation (NAT) Configuration

This document provides the following information about configuring IPv4 and IPv6 Network Address Translation (NAT) on the Extreme Networks S-Series platform.

Note: NAT is currently not supported on the S-Series S-130 module.

Using Network Address Translation in Your Network

IPv4 and IPv6 Network Address Translation (NAT) and IPv4 Network Address Port Translation (NAPT) are methods of concealing a set of host addresses on a private network behind a pool of public addresses. Together they are referred to as traditional NAT. A traditional NAT configuration is made up of a private network and a public network that are connected by a router with NAT enabled on it.

Basic NAT maps IP addresses from one group of addresses to another, transparent to the end user. A basic NAT translation is always between a single private IP address and a single public IP address.

NAPT translates many private network addresses, along with each private address' associated TCP/UDP port, into a single public network address and its associated TCP/UDP ports. Given that there is only a single public IP address associated with the translations, it is the public port that the private address and its port are associated with that allows for the uniqueness of each translation.

The S-Series platform supports IPv4-to-IPv4 (NAT44) and IPv6-to-IPv6 (NAT66) basic NAT and IPv4-to-IPv4 NAPT.

In addition, the following features are also supported:

- Static NAT using singular IPv4 or IPv6 IP addresses
- Dynamic NAT using IPv4 or IPv6 NAT address pools
- Cone NAT for all addresses and ports (fullcone), by address (restricted cone), or by port (port restricted cone)

- NAT hairpinning
- FTPALG, DNS ALG, NAPT for ICMP Pings, and ICMP error fixups

Extreme Networks support for NAT provides a practical solution for organizations who wish to streamline their IP addressing schemes. NAT operates on a router connecting a private network to a public network, simplifying network design and conserving IP addresses. NAT can help organizations merge multiple networks together and enhance network security by:

- Helping to prevent malicious activity initiated by outside hosts from entering the corporate network
- Augmenting privacy by keeping private intranet addresses hidden from view of the public internet, thereby inhibiting scans
- Limiting the number of IP addresses used for private intranets that are required to be registered with the Internet Assigned Numbers Authority (IANA)

Implementing NAT

To implement NAT in your network:

- Enable NAT on both the inside (local) and outside (public) interfaces to be used for translation
- If you intend to use inside source address dynamic translation (see "Dynamic Address Translations" on page 5 for details):
  - Define an access-list of inside addresses
  - Define a NAT address pool of outside addresses
  - Enable dynamic translation of inside addresses specifying an access-list of inside addresses and a NAT address pool of outside addresses
    - Optionally specify a NAT cone method along with an access list specifying the protocols and ports to cone
  - Optionally configure IPv4 overload for NAPT (defaults to NAT)
  - Optionally specify the interface to which translations are applied
- If you intend to use inside source address static translation (see "Static Address Translation" on page 3 for details), enable inside source address static translation in the appropriate NAT or NAPT context
  - Optionally specify a NAT cone method along with an access list specifying the protocols and ports to cone
- Optionally change the NAT FTP control port from its default of 21
- Optionally modify maximum allowed entries and NAT translation timeout values

NAT Overview

This section provides an overview of NAT configuration. A traditional NAT configuration is made up of a private network or intranet, a public network, and a router that interconnects the two networks. The private network is made up of one or more devices each assigned an inside (internal) address that is not intended to be directly connectable to a public network device. The router interconnecting the private and public networks support traditional NAT. It is NAT's responsibility to translate the inside address to a unique outside address to facilitate communication with the public network for intranet devices.

NAT allows translations between IP addresses. NAPT allows translations between multiple inside addresses and their associated ports and a single outside IP address and its associated ports. NAT and NAPT support both static and dynamic address translation.

NAT Binding

A NAT flow has two devices associated with it that are in communication with each other: the client device belonging to the inside (private) network and the server device belonging to the outside (public) network. Each active NAT flow has a binding resource associated with it. Each flow is based upon the following criteria:

If it is a non-FTP NAT flow:

- Source IP Address - The inside client IP address
- Destination IP Address - The outside server IP address

If it is a NAPT or FTP flow:

- Source IP Address - The inside client IP address
- Destination IP Address - The outside server IP address
- Source Port - The inside client source port
- Destination Port - The outside server destination port

Static Address Translation

Static address translations are one-to-one bindings between the inside and outside IP addresses. A static address binding is not deleted until the command that defines the binding is negated. When configuring NAT for static address translation, you assign a local IP address and a global IP address. When configuring NAPT for static address translation, you assign a local IP address and one of its associated L4 ports and a global IP address and one of its associated L4 ports. You also specify whether the IP protocol is TCP or UDP.

NAT Static Address Translation

Figure 45-1 on page 45-4 shows an example of a basic static NAT address translation. The three addresses that take part in this basic static NAT example can be either IPv4 or IPv6, but they can not be a mix of the two IP address types. The three example addresses are:

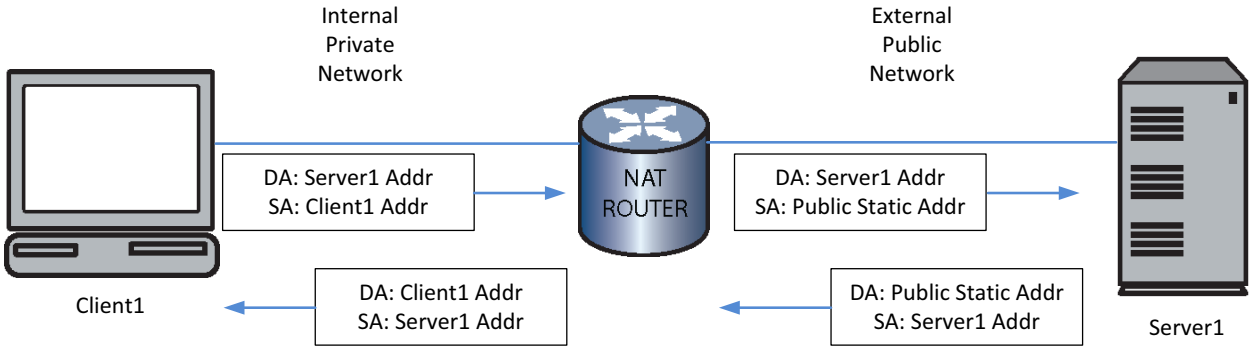
- The Client1 internal private network IP address
- The Server1 external public network IP address
- The statically configured external public network IP address

A static NAT translation is configured that maps the Client1 IP address to a publicly addressable static outside IP address.

A packet arrives at the NAT router from Client1 with a source address of Client1 IP address and a destination address of Server1 IP address. The packet leaves the NAT router with a source address of the public static IP address and a destination address of Server1 IP address. The IP packet's destination address is not changed, only the source IP address is changed. Server1 receives the packet from the NAT router with no knowledge of the internal private network Client1 IP address.

When Server1 responds to Client1, the packet arrives at the NAT router with Client1's translated public static IP address as the destination address, but leaves the NAT router with Client1's actual internal private network Client1 IP address as the destination address. The NAT router delivers Server1's response to the Client1 IP address.

Figure 45-1 Basic NAT Static Address Translation



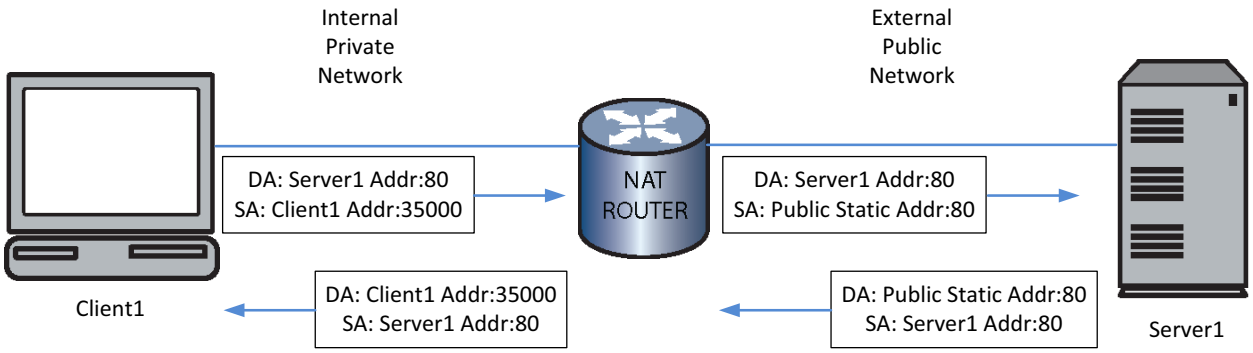
NAPT Static Address Translation

Figure 45-2 shows an example of a basic static NAPT translation. Client1 is a device on an internal private network that wants to connect to the web service at the Server1 IP address TCP port 80. The web service is in fact hosted by a Server1 on the public network. A static NAT translation is configured that maps the Client1 private network address to a static public network IP address and TCP port 80 to the Server1 public network IP address and TCP port 80.

A packet arrives at the NAT router from Client1 with the Client1 private IP address: port 35000, but leaves the NAT router with the public static source address: port 80. In both cases the destination is for Server1's public network IP address: port 80. From Server1's point of view, Client1's IP address is the public static IP address: port 80. Server1 doesn't know anything about its actual private Client1 IP address: port 35000.

When Server1 responds to Client1, its packet arrives at the NAT router with Client1's translated public static address: port 80 as the destination address, but leaves the NAT router with Client1's actual address: port 35000 as the destination address. The NAT router delivers Server1's response the Client1 IP address, port 35000.

Figure 45-2 Basic NAPT Static Address Translation



Dynamic Address Translations

Dynamic NAT is configured using a standard access-list, a NAT address pool, and a source list.

IPv4 NAT pool addresses are assigned as a range with a starting address and ending address. IPv6 NAT pools are assigned as a combination of a start address and prefix length and count, where count specifies a contiguous block of addresses from 1 to the value specified by count.

Note: IPv6 NAT pools must be assigned a prefix length of 112.

NAT pool IP addresses used in dynamic NATing are reassigned whenever they become free. Dynamic NAT bindings time out and are deleted due to idleness. A NAT translation timeout option is configurable for dynamic translations and defaults to 240 seconds.

The NAT source list is used to configure dynamic NAT. This is an association of an access-list and a NAT pool. The access list specifies the internal client source IP addresses that match the source list and the pool specifies the NAT pool to assign global IP addresses from. If a source list is configured as "overloaded" this means the NAT translations will use NAPT and the NAT pool may multiplex multiple private IP addresses to one NAT pool global address. NAPT translation is supported for IPv4 only.

You can also specify the egress VLAN interface for which this source list will be applied. Otherwise, the source list applies to all interfaces.

NAT Dynamic Address Translation

Figure 45-3 on page 45-6 shows an example of a basic dynamic NAT address translation. The overview shows two internal network clients: Client1 and Client2. Client1 displays a NATed dynamic address translation. Client2 displays a non-NATed configuration. The access-list assigned to Client1 dynamic translation must contain permits for the IP address of the local client. A NAT pool must be configured with at least a single address range of publicly available IP addresses and assigned to the inside source list. This is a NAT (not NAPT) dynamic translation so we do not assign the overload option.

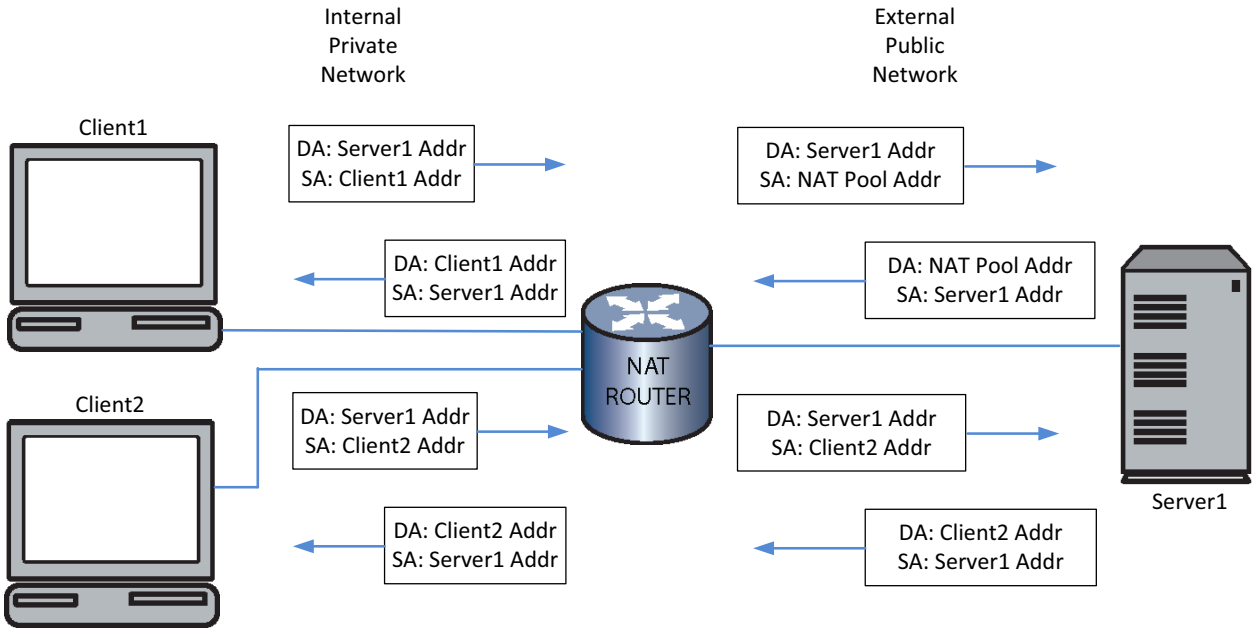
Client1 Walkthrough:

Client1 sends an IP packet to Server1 via the NAT router. The packet arrives on a VLAN configured as NAT inside and Server1 is accessible through a VLAN configured as NAT outside.

An access-list matching Client1's source IP address is configured to a NAT source list. A dynamic binding is created and a global IP address from the NAT pool is assigned to the binding. The packet is sent to Server1 with the destination IP unchanged and the source IP address changed to the NAT pool address.

Server1 sends an IP packet back to Client1 using the public NAT pool address as the destination address. This packet matches the previously created dynamic binding. Using the binding to determine the actual destination address, the NAT router sends the packet on to Client1 with the destination IP address changed from the NAT pool address to the Client1 address. The source IP address remains unchanged.

Figure 45-3 Basic NAT Dynamic Address Translation



Client2 Walkthrough:

Client2 presents an unNATed example. Client2's actual source address is seen by the external network both when Server1 receives data from and sends data to Client2.

NAPT Dynamic Inside Address Translation

Figure 45-4 on page 45-7 shows an example of a basic dynamic NAPT address translation. NAPT address translation is only supported for IPv4 addressing. The example shows network client Client1. The access-list assigned to this dynamic translation must contain permits for the Client1 IP address. A NAT pool can be configured with a single IP address for its range of publicly available IP addresses. The pool is assigned to the source list. A single public IP address will be sufficient should multiple clients be configured because NAPT will use the available L4 port range of this IP address when assigning addresses for dynamic translation. This is a NAPT dynamic translation so we must assign the overload option when configuring the source list.

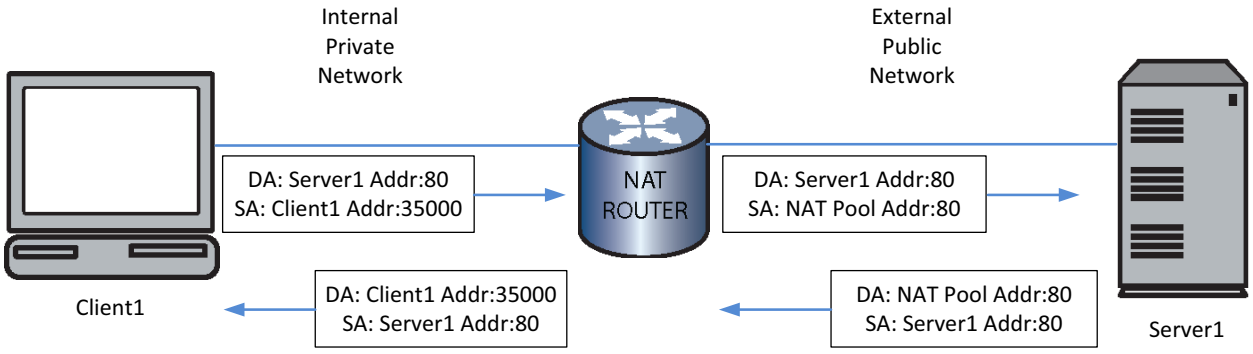
Client1 sends a TCP packet (source port 35000) to Server1 port 80, via the NAT router. The packet arrived on a VLAN configured as NAT inside and Server1 is accessible through a VLAN configured as NAT outside.

An access-list matching Client1's source IP address is configured to a NAT source list. A dynamic binding is created and a global IP address is assigned to the binding. Since the source list is overloaded the NAT pool is checked to see if Client1's original source port (35000) is in use for the global NAT pool address. If this port is already in use by some other binding, a new source port is chosen and assigned to the binding. In this example we will assume 35000 is already used and assume the NAT pool assigned source port 80.

The packet is sent to Server1 with the destination IP address and TCP port unchanged and the source IP address changed to global NAT pool address with the TCP source port changed to 80.

When Server1 responds to Client1, its packet arrives at the NAT router with Client1's translated address (global NAT pool address port 80) as the destination address, but leaves the NAT router with Client1's actual address (Client1 IP address port 35000) as the destination address. Server1's response is delivered to Client1 IP address port 35000.

Figure 45-4 Basic NAPT Dynamic Inside Address Translation



Stateful NAT Firewall

A stateful NAT firewall is a NAT feature that protects members of the inside network from access from outside network clients for which a dynamic NAT firewall binding does not exist, while at the same time allowing outside traffic not destined to the inside network to flow freely. The firewall provides protection against unwanted connections, such as a potential hacker, being established from the outside interface to the users secure network. The establishment of connections is controlled based on:

- The inside or outside interface direction
- The source and destination IP address
- Protocols such as ICMP, TCP, and UDP
- Applications such as HTTP, TCP, and TPTP
- Connection state of the request

In addition, the stateful NAT firewall feature allows privately addressed hosts to share the firewall's public IP along with the standard NAT feature that internal network addresses are not visible to the outside world.

By controlling the establishment of connections the users system can be protected from malicious and unwanted access to the user's network, while allowing clients on the user's network to access servers in the unsecured outside network.

A stateful NAT firewall is configured by creating a standard dynamic NAT list rule without specifying a NAT pool using the **ip/ipv6 nat inside source list** commands.

From Inside Network to Outside Network

When a packet originates from the inside network and transits the router to the outside network, the ACL associated with the dynamic NAT firewall list rule is checked and NAT creates a binding only if a match occurs for a permit rule. The packet is forwarded to the outside network when a permit rule match occurs, otherwise the packet is dropped.

From Outside Network to Inside Network

When you configure a standard dynamic NAT list rule and specify a NAT pool, if a packet arrives on an outside interface destined for the original source address (the natted global address obtained from the configured NAT Pool), and there is no matching binding. The packet is dropped because it was destined for a NAT Global address.

With a dynamic NAT firewall list rule (no pool specified), the inside address is not Natted and is visible to the outside world. Therefore, it is possible for a packet to arrive on an outside network destined to an inside address defined in the dynamic NAT firewall list rule without matching an existing binding.

In order to insure that packets from the outside network do not leak through to the inside network when no binding exists, the ACL configured on the dynamic NAT firewall list rule is examined in reverse: the packet destination IP or port is matched against the ACL source IP and port and the packet source IP or port is matched against the destination IP and port. When no binding exists, packets on the outside network matching an ACl permit rule are dropped by the firewall. A packet arriving on an outside network matching an ACL deny rule is forwarded by the firewall.

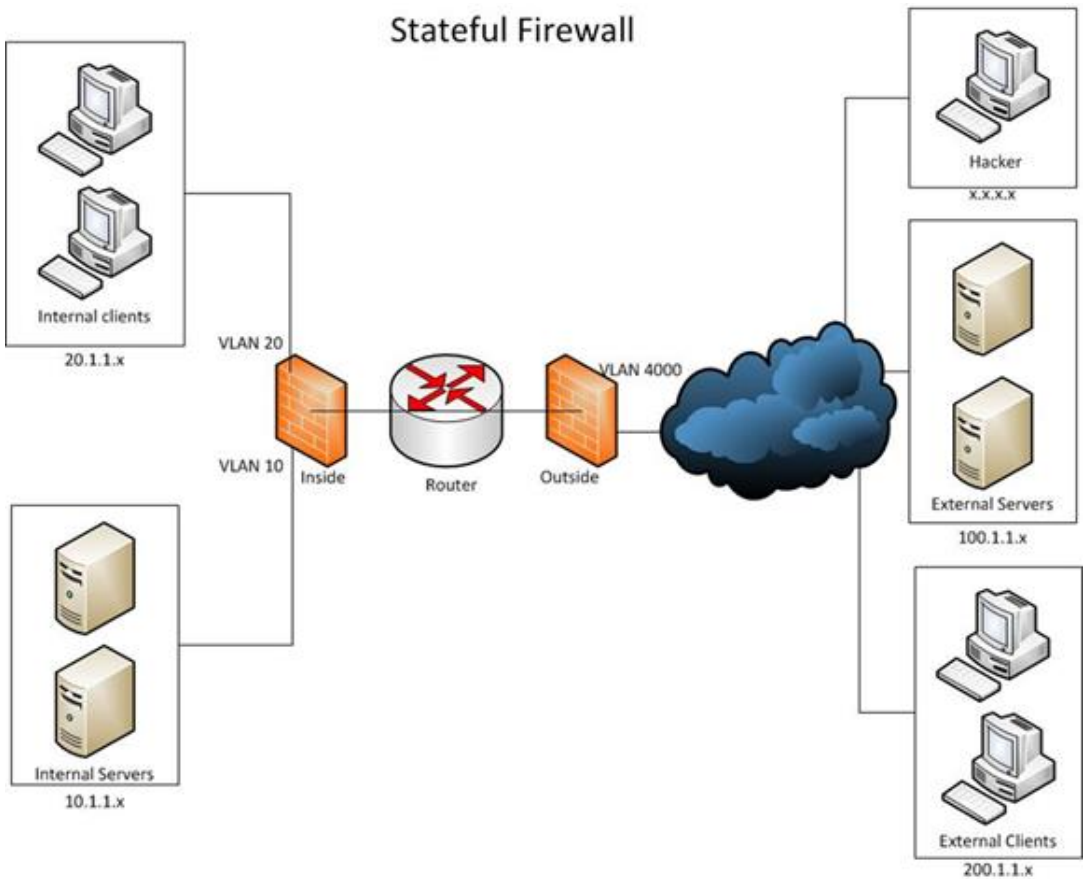
NAT Firewall Configuration Example

Figure 45-5 displays a NAT stateful firewall configuration example where packets flowing from the inside network to the outside network will result in NAT firewall bindings only for TCP and ICMP packets originating from either the Internal Clients or the Internal Servers. Packets for all other protocols on VLANs 10 and 20 are dropped.

Packets returning from the outside network matching a dynamic NAT firewall binding will be forwarded.

Any packets arriving on the outside network, not matching an existing dynamic NAT firewall binding and destined to either the Internal Clients or the Internal Servers will be dropped. This prohibits any user coming from the outside network from initiating a connection to the inside network, providing a secure inside network.

Figure 45-5 NAT Stateful Firewall Configuration Example



This NAT firewall configuration example:

- Creates an inside network VLAN 10 for internal servers
- Creates an inside network VLAN 20 for internal clients
- Creates an outside network VLAN 4000
- Creates a firewall ACL (“firewall_acl”) that permits TCP, ICMP, and IP packets for both the internal servers and clients
- Configures the dynamic NAT firewall list rule

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 10
S Chassis(rw-config-intf-vlan.0.10)->description “Internal Servers”
S Chassis(rw-config-intf-vlan.0.10)->ip address 10.1.1.1 255.255.255.0 primary
S Chassis(rw-config-intf-vlan.0.10)->ip nat inside
S Chassis(rw-config-intf-vlan.0.10)->no shutdown
S Chassis(rw-config-intf-vlan.0.10)->exit
S Chassis(rw-config)->interface vlan 20
S Chassis(rw-config-intf-vlan.0.20)->description “Internal Clients”
S Chassis(rw-config-intf-vlan.0.20)->ip address 20.1.1.1 255.255.255.0 primary
S Chassis(rw-config-intf-vlan.0.20)->ip nat inside
S Chassis(rw-config-intf-vlan.0.20)->no shutdown
S Chassis(rw-config-intf-vlan.0.20)->exit
S Chassis(rw-config)->interface vlan 4000
S Chassis(rw-config-intf-vlan.0.4000)->description “Outside Network”
S Chassis(rw-config-intf-vlan.0.4000)->ip address 100.1.1.1 255.255.255.0 primary
S Chassis(rw-config-intf-vlan.0.4000)->ip nat outside
S Chassis(rw-config-intf-vlan.0.4000)->no shutdown
S Chassis(rw-config-intf-vlan.0.4000)->exit
S Chassis(rw-config)->ip access-list extended firewall_acl
S Chassis(rw-cfg-ext-acl)->permit tcp 10.1.1.1 0.0.0.255 any log-verbose
S Chassis(rw-cfg-ext-acl)->permit icmp 10.1.1.1 0.0.0.255 any log-verbose
S Chassis(rw-cfg-ext-acl)->permit tcp 20.1.1.1 0.0.0.255 any log-verbose
S Chassis(rw-cfg-ext-acl)->permit icmp 20.1.1.1 0.0.0.255 any log-verbose
S Chassis(rw-cfg-ext-acl)->permit ip any 10.1.1.1 0.0.0.255 log-verbose
S Chassis(rw-cfg-ext-acl)->permit ip any 20.1.1.1 0.0.0.255 log-verbose
S Chassis(rw-cfg-ext-acl)->exit
S Chassis(rw-config)->ip nat inside source list firewall_acl overload
S Chassis(rw-config)->
```

Cone NAT

The cone NAT feature defines additional methods by which external hosts can communicate with an internal private network client using the external public network address mapped in a NAT binding. These additional cone NAT methods are required by products such as Microsoft Xbox LIVE.

When configuring a cone NAT, an access list permitting one or more protocols and ports is assigned to the cone NAT configuration. In order for the cone NAT binding to be created, the

packet sent by the internal client must pass the protocol and port criteria listed in the cone NAT access list. Once passed, the listed protocol and port criteria become part of the binding. If the packet initially sent by the internal client does not pass the cone NAT access list protocol and port criteria, a non-cone NAT binding is created.

There are two packet flow directions for any cone NAT binding. Forward is from the perspective of the internal Client to the external host. Reverse is from the perspective of the external host to the internal client. For each cone NAT method the forward direction has the same behavior as a basic NAT binding, with the exception that the packet must pass the cone NAT access list protocol and port criteria. Once an internal IP address and port is mapped to an external IP address and port, any packets from the internal address matching the cone NAT access list criteria will be sent through the external address as it is forwarded to the external host.

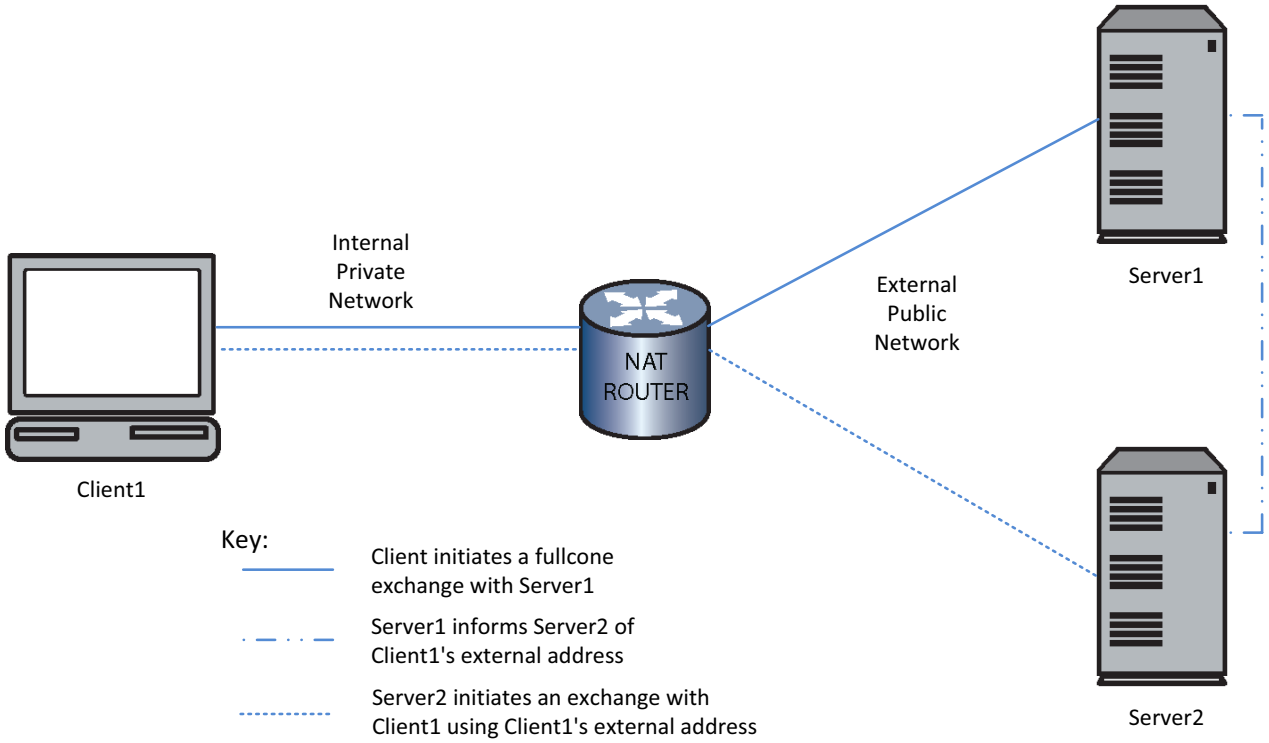
There are three cone NAT methods and they are defined by their reverse packet flow behavior.

Fullcone NAT

The fullcone NAT method allows any external host (regardless of whether the internal client has initiated any contact with it or not) to send packets to the internal client IP address and port, using the external natted address defined by the NAT binding. The reverse packet flow for the fullcone NAT binding can be from any server, port, using any protocol. The only requirement is that the sending server know the external public address of the fullcone NAT binding.

Figure 45-6 on page 45-10 shows an example of fullcone NAT.

Figure 45-6 Fullcone NAT



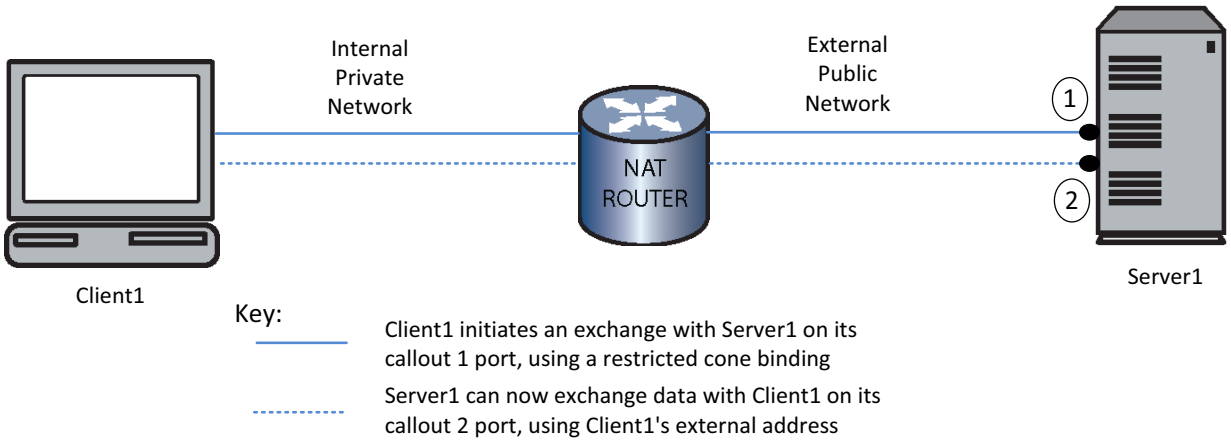
(Address) Restricted Cone NAT

The restricted cone NAT restricts an external host's ability to initiate a packet exchange with the internal client by IP address. The restricted cone NAT method requires that the internal client has already initiated a packet exchange with the external host that passed the protocol and port criteria listed in the access list assigned to the restricted cone NAT configuration. Once the internal

client initiates a packet exchange with the external host, that host can initiate a packet exchange for that restricted cone binding using any port or protocol. For the restricted cone NAT method, the external server must be the server the client initiated the exchange with (address restriction) and it must know the external public address of the internal client. The external server can initiate an exchange from any port, using any protocol.

Figure 45-7 on page 45-11 shows an example of address restricted cone NAT.

Figure 45-7 Restricted Cone NAT

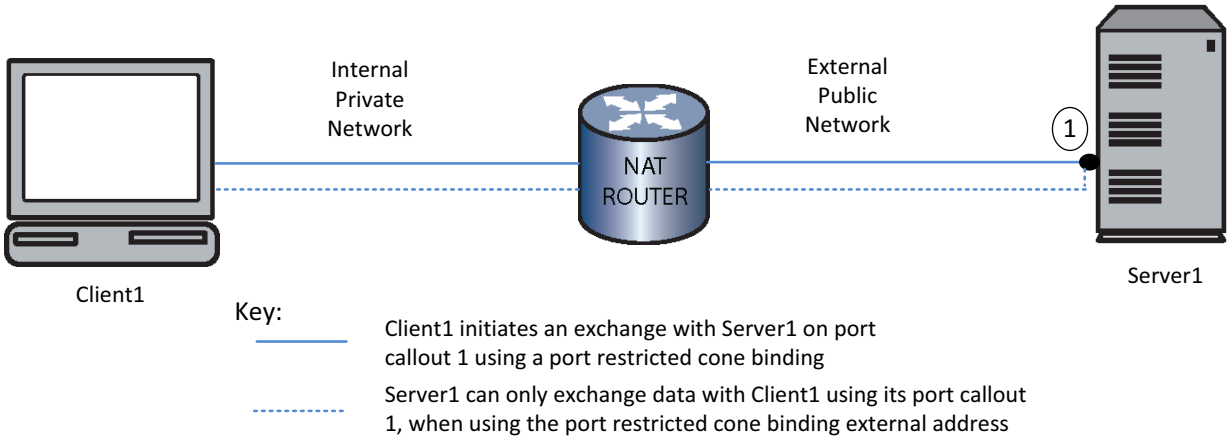


Port Restricted Cone NAT

The port restricted cone NAT restricts an external host's ability to initiate a packet exchange with the internal client by both IP address and port. The restricted cone NAT method requires that the internal client has already initiated a packet exchange with the external host port that passed the protocol and port criteria listed in the access list assigned to the port restricted cone NAT configuration. Once the internal client initiates a packet exchange with the external host, that host can only initiate a packet exchange with the internal client using the port the internal client sent the initial packet flow to. The external server can initiate an exchange using any protocol.

Figure 45-8 shows an example of port restricted cone NAT.

Figure 45-8 Port Restricted Cone NAT

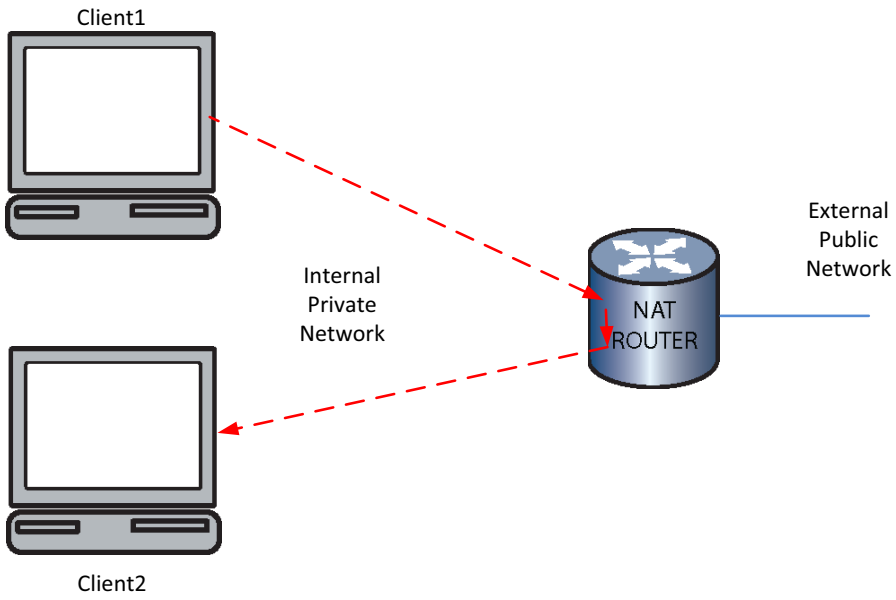


NAT Hairpinning

NAT hairpinning allows an internal client to forward packets to another internal client using the destination internal client's global NAT address. NAT hairpinning is necessary for two internal endpoints to communicate when only their external mapped addresses are known to each other. NAT hairpinning does not require any CLI configuration. The NAT router will NAT the incoming inside packet (source address and port) according to standard NAT rules. The NAT router examines the packet destination IP address and port. If a NAT binding exists for the destination IP address and port, the NAT router forwards the packet to the mapped internal client.

Figure 45-9 on page 45-12 shows an example of NAT hairpinning. In this example, Client1 initiates communication Client2. Client1 sends a packet to the global address mapped to Client2's internal address. Because the NAT router supports NAT hairpinning, it recognizes Client1 as an internal address and the packet destination address as a global address bound to Client2's internal address. The NAT router remaps the packet destination address to Client2's internal address and forwards the packet.

Figure 45-9 NAT Hairpinning



NAT Translation Protocol Rules

Translation protocol rules are provided as a dynamic means of setting NAT binding idle time out and "one-shot" settings, based on IP protocol or TCP/UDP port number. Generally these rules apply only to bindings that track the IP protocol (and UDP/TCP ports where applicable). This means that, in general, they only apply to NAPT dynamic bindings or special case bindings like FTP Control/Data that require a binding per connection. A one-shot binding works as a normal binding in that when a packet is received, the processing of the packet results in the creation of the binding, and the packet is forwarded to its destination. When a return packet is received and processed, the packet is sent back to the peer and the binding is deleted. One-shot bindings are useful for processing simple bidirectional traffic that sends one packet in each direction, like ICMP and some UDP traffic like DNS. One-shot bindings provide the benefit of being able to quickly clean up the bindings that may otherwise hang around waiting to time out, using up a NAT binding resource that would never be reused. One-shot bindings are only usable with NAPT and can not be used with the TCP protocol.

Use the **ip | ipv6 nat translation protocol** in global configuration command mode to create a translation protocol rule for a specified IP protocol, UDP, or TCP port.

NAT Timeouts

The maximum timeout value in seconds per flow is configurable for the following flow types:

- Dynamic translation
- UDP and TCP
- ICMP
- DNS
- FTP (IPv4 only)
- TCP finish reset (FIN/RST)

DNS, FTP and ICMP Support

NAT works with DNS by having the DNS Application Layer Gateway (ALG) translate an address that appears in a Domain Name System response to a name or inverse lookup.

NAT works with FTP by having the FTP ALG translate the FTP control payload. Both FTP PORT CMD packets and 227 Passive Response packets, containing IP address information within the data portion, are supported. The FTP control port is configurable. NAT also supports the FTP extended modes as defined in RFC2428.

The NAT implementation also supports translation of the IP address embedded in the data portion of the following types of ICMP error message: destination unreachable (type3), source quench (type4), redirect (type5), time exceeded (type 11) and parameter problem (type 12). NAT also supports an ALG for ICMP echo request/reply messages when they are forwarded via an overloaded (port-NATed) list rule.

NAT DNS Packet Inspection and Fixup

NAT provides an ALG (Application Layer Gateway) for the inspection and fixup of DNS packets that are being forwarded by the NAT process. NAT DNS packet inspection and fixup consists of parsing DNS request or response packets, identifying IP addresses contained within that may need to be NATed, and fix up the DNS packet with the appropriate NAT translations.

NAT inspection of DNS packets is disabled by default.

Use the **ip | ipv6 nat inspect dns** command in global configuration command mode to enable NAT DNS packet inspection and fixup.

Enabling NAT

When traffic subject to translation originates from or is destined to an interface, that interface must be enabled for NAT. If the interface is part of the internal private network, it should be enabled as an inside interface by configuring an inside source list. If the interface is part of the external public network, it should be enabled as an outside interface by configuring an outside source list.

Use the **ip | ipv6 nat inside** command in interface configuration mode to enable an inside interface.

Use the **ip | ipv6 nat outside** command in interface configuration mode to enable an outside interface.

Configuring NAT

This section provides details for the configuration of NAT on the S-Series products.

Table 45-1 lists NAT parameters and their default values.

Table 45-1 Default NAT Parameters

Parameter	Description	Default Value
Overload	Specifies that NAPT translation should take place for this dynamic pool binding.	NAT translation
Timeout	Specifies the timeout value applied to dynamic translations.	240 seconds
UDP timeout	Specifies the timeout value applied to the UDP translations.	240 seconds
TCP timeout	Specifies the timeout value applied to the TCP translations.	240 seconds
ICMP timeout	Specifies the timeout value applied to the ICMP translations.	240 seconds
DNS timeout	Specifies the timeout value applied to the DNS translations.	240 seconds
FTP timeout	Specifies the timeout value applied to the FTP translations. (IPv4 only)	240 seconds

Table 45-2 lists NAT resource limits.

Table 45-2 NAT Resource Limits

Resource	S-Series
Global Bindings	65536
IP Addresses	2000
Pools	10
Port Mapped Addresses	20
Static Rules	1000

Configuring Traditional NAT Static Inside Address Translation

Procedure 45-1 describes how to configure traditional NAT for a static configuration.

Procedure 45-1 Traditional NAT Static Configuration

Step	Task	Command(s)
1.	Enable NAT, in interface configuration mode, on both the inside and outside interfaces.	**ip | ipv6 nat** {**inside** | **outside**}

Procedure 45-1 Traditional NAT Static Configuration

Step	Task	Command(s)
2.	Enable, in global configuration mode, any static NAT translations of IPv4 or IPv6 inside source addresses. Inside source static rules allow NAT translation of data ingressing a NAT outside interface destined to the static rule's global-ip address.	**ip nat inside source static** *local-ip global-ip* [**inside-vrf** *vrf-name*] [**fullcone** *acl* | **restricted-cone** *acl* | **port-restricted-cone** *acl*] **ipv6 nat inside source static** *local-ipv6*/*prefix-length global-ipv6*/*prefix-length* [**inside-vrf** *vrf-name*] [**fullcone** *acl* | **restricted-cone** *acl* | **port-restricted-cone** *acl*]
3.	Enable, in global configuration mode, any static NAPT translations of IPv4 inside source addresses, specifying whether the L4 port is a TCP or UDP port. Inside source static rules allow NAT translation of data ingressing a NAT outside interface destined to the static rule's protocol, global-ip address and global-port.	**ip nat inside source static** {**tcp** | **udp**} *local-ip local-port global-ip global-port*

Configuring Traditional NAT Dynamic Inside Address Translation

Procedure 45-2 describes how to configure traditional NAT for a dynamic configuration.

Procedure 45-2 Traditional NAT Dynamic Configuration

Step	Task	Command(s)
1.	Enable NAT, in interface configuration mode, on both the inside and outside interfaces.	**ip | ipv6 nat** {**inside** | **outside**}
2.	Define, in global configuration mode, an IPv4 or IPv6 access-list of permits for all inside addresses to be used by this dynamic translation.	**ip** | **ipv6 access-list** *list-number* {**deny** | **permit**} *source*
3.	Define, in global configuration mode, a NAT address pool for all IPv4 or IPv6 outside addresses to be used by this dynamic translation.	**ip nat pool** *name start-ip-address end-ip-address* [**netmask** *netmask* | **prefix-length** *prefix-length*] **ipv6 nat pool** *name start-ip-address*/*prefix-length* **count** *count*
4.	Enable, in global configuration mode, dynamic translation of inside source addresses. Specify the overload option for NAPT translations. Do not specify a pool when configuring a dynamic NAT firewall list rule.	**ip** | **ipv6 nat inside source list** *access-list* [**pool** *pool-name*] [**interface** *interface-name*] [**overloaded**] [**inside-vrf** *vrf-name*] [**fullcone** *acl* | **restricted-cone** *acl* | **port-restricted-cone** *acl*]

Managing a Traditional NAT Configuration

Table 45-3 describes how to manage traditional NAT configurations. All traditional NAT management configuration is optional.

Table 45-3 Managing a Traditional NAT Configuration

Task	Command(s)
In global configuration mode, specify a non-default IPv4 NAT FTP control port.	**ip nat ftp-control-port** *port-number*
In global configuration mode, set the maximum number of IPv4 or IPv6 translation entries.	**ip** | **ipv6 nat translation max-entries** *number*
In global configuration mode, set IPv4 or IPv6 NAT translation timeout values. FTP timeout is IPv4 only.	**ip** | **ipv6 nat translation** {**timeout** | **udp-timeout** | **tcp-timeout** | **icmp-timeout** | **dns-timeout** | **finrst-timeout**} *seconds* **ip nat translation ftp-timeout** *seconds*
In global configuration mode, create an IPv4 or IPv6 NAT translation protocol rule.	**ip** | **ipv6 nat translation protocol** *protocol* **timeout** [*seconds*] [**one-shot**]
In global configuration mode, enable logging to log a message for each created or deleted IPv4 or IPv6 NAT binding.	**ip** | **ipv6 nat log translations**
In global configuration mode, enable IPv4 or IPv6 NAT inspection and fixup of DNS packets forwarded by the NAT process.	**ip** | **ipv6 nat inspect dns**
In global configuration mode, clear IPv4 or IPv6 NAT bindings.	**clear ip** | **ipv6 nat bindings** {**all** | **pool** *pool* | **id** *id* | **match** {*protocol* | * | **icmp** {*sip* | *} {*dip* | *} | **tcp** {*sip* | * *port* | *} {*dip* | * *port* | *} | **udp** {*sip* | *} {*dip* | *}} [**detail**]
In global configuration mode, clear IPv4 or IPv6 NAT statistics.	**clear ip** | **ipv6 nat statistics**

Displaying NAT Statistics

Table 45-4 describes how to display NAT statistics.

Table 45-4 Displaying NAT Statistics

Task	Command(s)
Display NAT bindings.	**show ip** | **ipv6 nat bindings** [**id** *binding-id*] [**pool** *pool* [**detail**]] [**match** *protocol* {*sip dip* [**detail**] | *}] [**summary**]
Display NAT information.	**show ip** | **ipv6 nat info**
Display NAT lists matching rules.	**show ip** | **ipv6 nat lists** [*list-name*] [**detail**]
Display NAT pools.	**show ip** | **ipv6 nat pools** [*name*] [*detail*]
Display NAT static matching rules.	**show ip** | **ipv6 nat statics** [**detail**]
Display NAT statistics.	**show ip** | **ipv6 nat statistics** [**-all_vrfs**] [**-interesting**]

NAT Configuration Examples

This section provides a configuration example for both the static and dynamic configurations. Each example includes both the NAT and NAPT translation methods.

Note: For purposes of our examples we will not modify the maximum number of translation entries. This parameter should only be modified to assure availability to features that share translation resources such as TWCB and LSNAT. It is recommended that you consult with Extreme Networks Customer Support before modifying this parameter value.

We will also assume that the FTP control port will use the default value.

IPv4 NAT Static Configuration Example

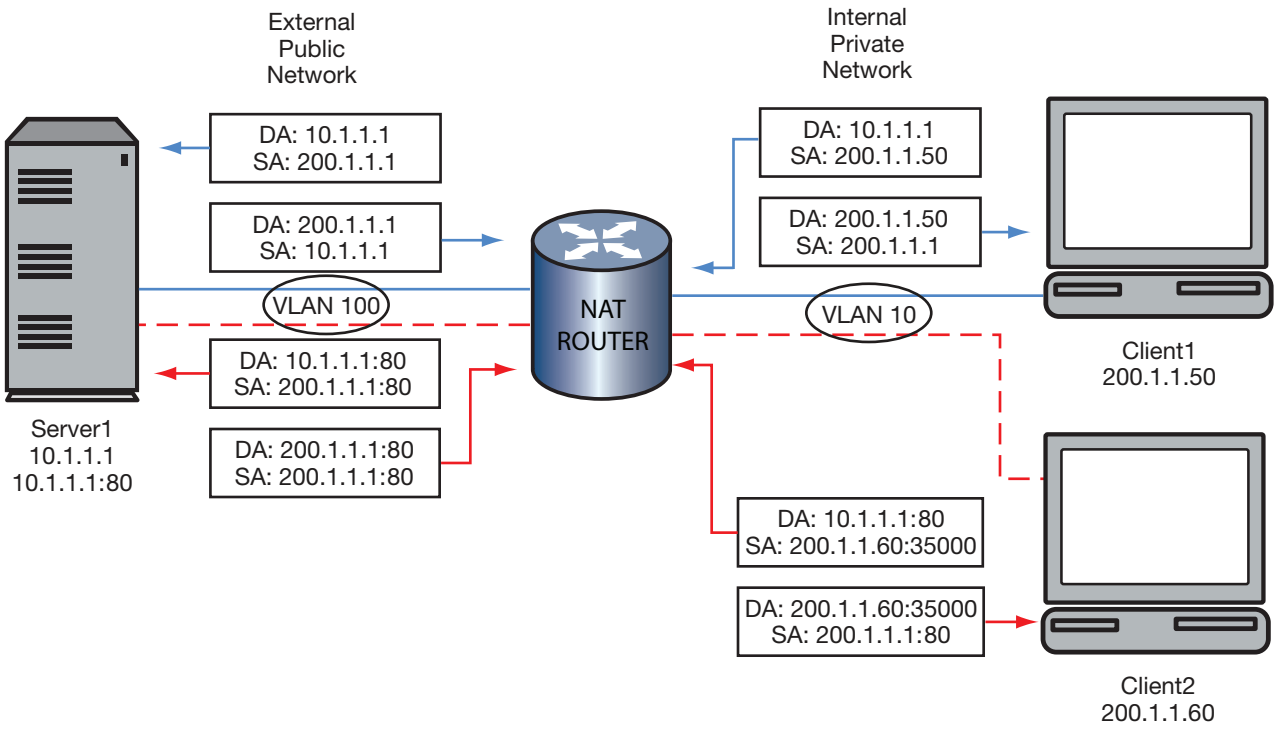
This example steps you through an IPv4 NAT static configuration for both NAT and NAPT translation methods. See Figure 45-10 on page 45-18 for a depiction of the IPv4 NAT static configuration packet flow.

Our static NAT configuration example configures two clients: Client1 with NAT translation and Client2 with NAPT translation. Both clients are on the internal private network VLAN 10 interface and communicate with Server1 over the external public network VLAN 100 interface. NAT is enabled on VLAN 10 as an inside interface. NAT is enabled on VLAN 100 as an outside interface. These are the only VLANs over which translation occurs for the static portion of this configuration example.

To configure Client1 on the NAT router, we enable static NAT translation of the inside source address specifying local IP address 200.1.1.50 and global IP address 200.1.1.1. Server1 will only see Client1 as IP address 200.1.1.1.

To configure Client2 on the NAT router, we enable static NAT translation of the inside source address specifying local IP address 200.1.1.60:35000 and global IP address 200.1.1.1:80. Server1 will only see Client2 as IP address 200.1.1.1:80.

Figure 45-10 IPv4 NAT Static Configuration Example



Enable NAT Inside and Outside Interfaces

Enable NAT inside interface:

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 10
S Chassis(rw-config-intf-vlan.0.10)->ip nat inside
S Chassis(rw-config-intf-vlan.0.10)->exit
S Chassis(rw-config)->
```

Enable NAT outside interface:

```
S Chassis(rw-config)->interface vlan 100
S Chassis(rw-config-intf-vlan.0.100)->ip nat outside
S Chassis(rw-config-intf-vlan.0.100)->exit
S Chassis(rw-config)->
```

Enable Static Translation of Inside Source Addresses

Enable the NAT static translation of the inside source address:

```
S Chassis(rw-config)->ip nat inside source static 200.1.1.50 200.1.1.1
```

Enable the NAPT static translation of the inside source address:

```
S Chassis(rw-config)->ip nat inside source static tcp 200.1.1.60:35000 200.1.1.2
80
```

IPv6 NAT Static Configuration Example

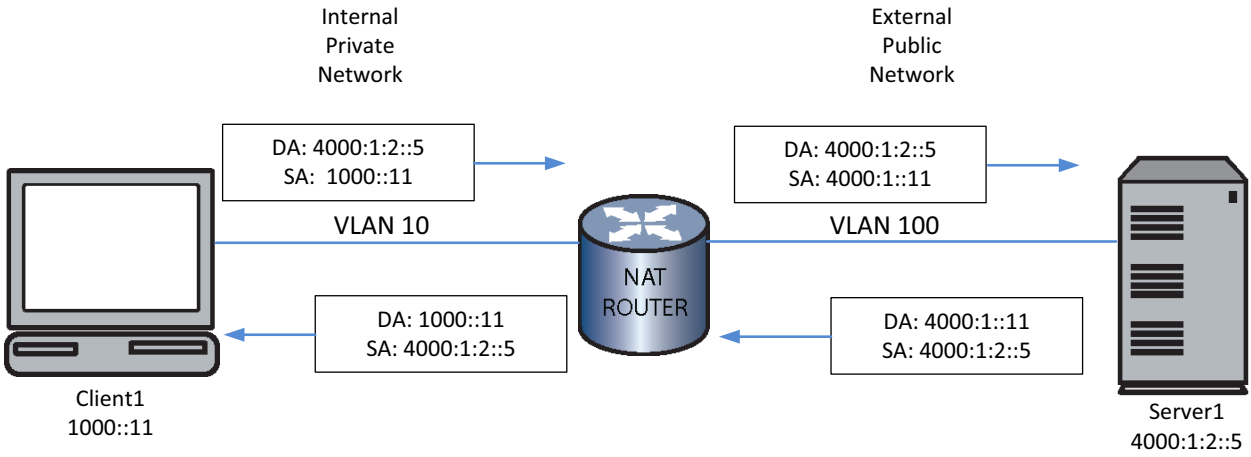
This example steps you through an IPv6 fullcone NAT static configuration. See Figure 45-11 on page 45-19 for a presentation of the IPv6 NAT static configuration packet flow.

Our static NAT configuration example configures Client1 with NAT translation on the internal private network VLAN 10 interface. Client1 communicates with Server1 over the external public network VLAN 100 interface. NAT is enabled on VLAN 10 as an inside interface. NAT is enabled on VLAN 100 as an outside interface. These are the only VLANs over which translation occurs for this configuration example. The static configuration is for fullcone NAT.

To configure Client1 on the NAT router, we enable static NAT translation of the inside source address specifying local IP address 1000::/48 and global IP address 4000:1::/48. Server1 will only access Client1 with its external IP address using a binding created by this static configuration. In this case: external IP address 4000:1::11. The fullcone option is specified and the extended IPv6 access list cone_acl is applied to it. The access list permits TCP packets from any port on source 1000::/48 to destination 4000:1:2::/48.

Once Client1 communicates with Server1, Server1 will be able to inform any other server of its 4000:1::11 external address. Any server that learned Client1's external address from Server1 can then initiate communications with Client1. Figure 45-11 on page 45-19 only displays the packet flow between Client1 and Server1. See Figure 45-6 on page 45-10 for a graphic depiction of the fullcone NAT feature.

Figure 45-11 IPv6 NAT Static Configuration Example



Enable NAT Inside and Outside Interfaces

Enable NAT inside interface:

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 10
S Chassis(rw-config-intf-vlan.0.10)->ipv6 nat inside
S Chassis(rw-config-intf-vlan.0.10)->exit
S Chassis(rw-config)->
```

Enable NAT outside interface:

```
S Chassis(rw-config)->interface vlan 100
S Chassis(rw-config-intf-vlan.0.100)->ipv6 nat outside
S Chassis(rw-config-intf-vlan.0.100)->exit
S Chassis(rw-config)->
```

Create the Fullcone Access List

```
S Chassis(rw-config)->ipv6 access-list extended cone_acl
S Chassis(rw-cfg-ipv6-ext-acl)->permit tcp 1000::/48 range 0 65565 4000:1:2::/48
S Chassis(rw-cfg-ipv6-ext-acl)->exit
S Chassis(rw-config)->
```

Enable Static Translation of Inside Source Addresses

Enable the NAT static translation of the inside source address:

```
S Chassis(rw-config)->ipv6 nat inside source static 1000::/48 4000:1::/48
full-cone cone_acl
```

NAT Dynamic Configuration Example

This example steps you through a NAT Dynamic Configuration for:

- IPv4 and IPv6 basic dynamic NAT
- IPv4 dynamic NAPT
- IPv6 fullcone dynamic NAT

See Figure 45-12 on page 45-21 for a presentation of the IPv4 dynamic NAT and NAPT example setup. See Figure 45-13 on page 45-22 for a presentation of the IPv6 dynamic NAT and fullcone NAT example setup.

The dynamic NAT configuration example configures two IPv4 and two IPv6 clients. Table 45-5 provides configuration details for each client. In all cases, the packet flow destination is Server1 IPv4 address 200.1.1.50 or IPv6 4000:1:2::5.

Table 45-5 Client Configuration Table

Client	Description
Client1	An IPv4 basic dynamic configuration (Figure 45-12 on page 45-21). • IPv4 standard access list **clientIPv4_acl** permits Client1's local IP address **10.1.1.1**. • The IPv4 NAT pool **natIPv4_pool** allows an address range of **200.1.1.1** through **200.1.1.10** to be used as the global IPv4 address pool. IPv4 external address **200.1.1.1** is used. • VLAN **10** is enabled as a NAT inside interface. VLAN **100** is enabled as a NAT outside interface.
Client2	An IPv6 basic dynamic configuration (Figure 45-13 on page 45-22). • IPv6 standard access list **clientIPv6_acl** permits Client2's local IP address **1000::20**. • The **natIPv6_pool1** allows an address range of **4000:1:1:1::/112** count **100** to be used as the global IPv6 address pool. IPv6 address **4000:1:1:1::10** is used. • VLAN **10** is enabled as a NAT inside interface. VLAN **100** is enabled as the NAT outside interface.

Table 45-5 Client Configuration Table

Client	Description
Client3	An IPv6 basic dynamic fullcone configuration (Figure 45-13 on page 45-22). • IPv6 standard access list **clientIPv6_acl** permits Client3's local IP address **1000::30**. • The **natIPv6_pool2** allows an address range of **4000:2:2:2::/112** count **100** to be used as the global IPv6 address pool. IPv6 address **4000:2:2:2::20** is used. • VLAN **20** is enabled as a NAT inside interface. VLAN **200** is enabled as the NAT outside interface. • Fullcone NAT is configured and **cone_acl** is assigned to the configuration. The extended access list **cone_acl** permits TCP packets for Xbox LIVE related ports **160** through **168** sourced from **1000::/48** and destined to **4000:1:2::/48**. If the Client3 sourced packet passes the **cone_acl** entry, a fullcone NAT binding will be applied and any server that knows the **4000:2:2:2::20** global address can initiate communications with Client3
Client4	An IPv4 NAPT dynamic configuration (Figure 45-12 on page 45-21). • IPv4 standard access list **clientIPv4_acl** permits Client1's local IP address **10.1.1.4.** • The IPv4 NAPT pool **naptIPv4_pool** has a single entry address of **200.1.1.20** to be used as the NAPT global IPv4 address. For this example, the source address port is **125**. In the example, source port 125 is already in use. Port **80** is used instead. IPv4 external address **200.1.1.20:80** is used. • VLAN **20** is enabled as a NAT inside interface. VLAN **200** is enabled as a NAT outside interface.

Figure 45-12 IPv4 NAT Dynamic Configuration Example

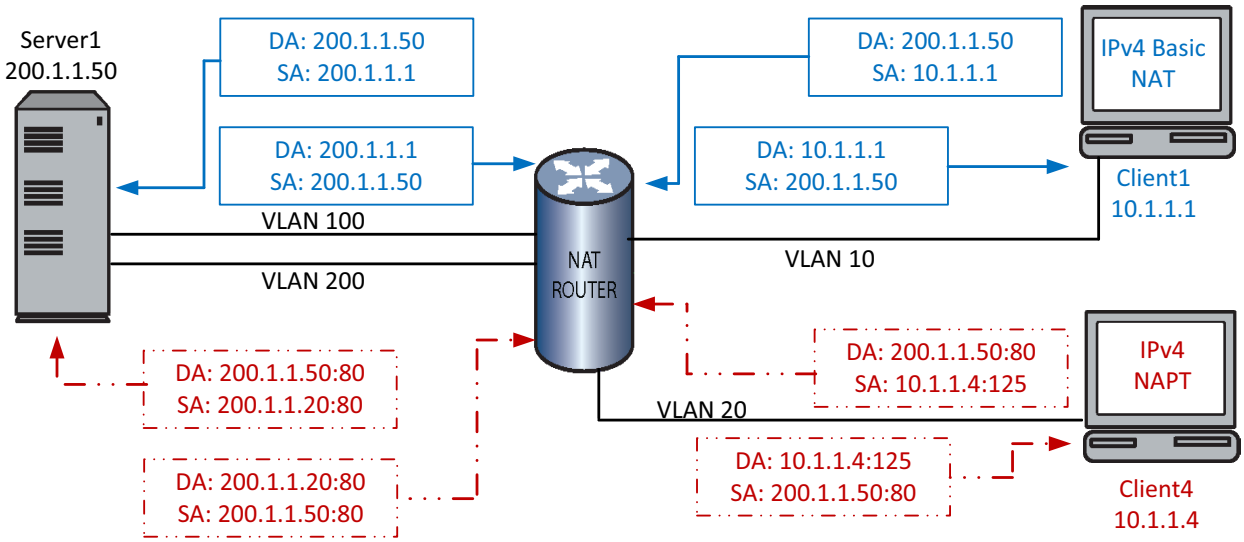
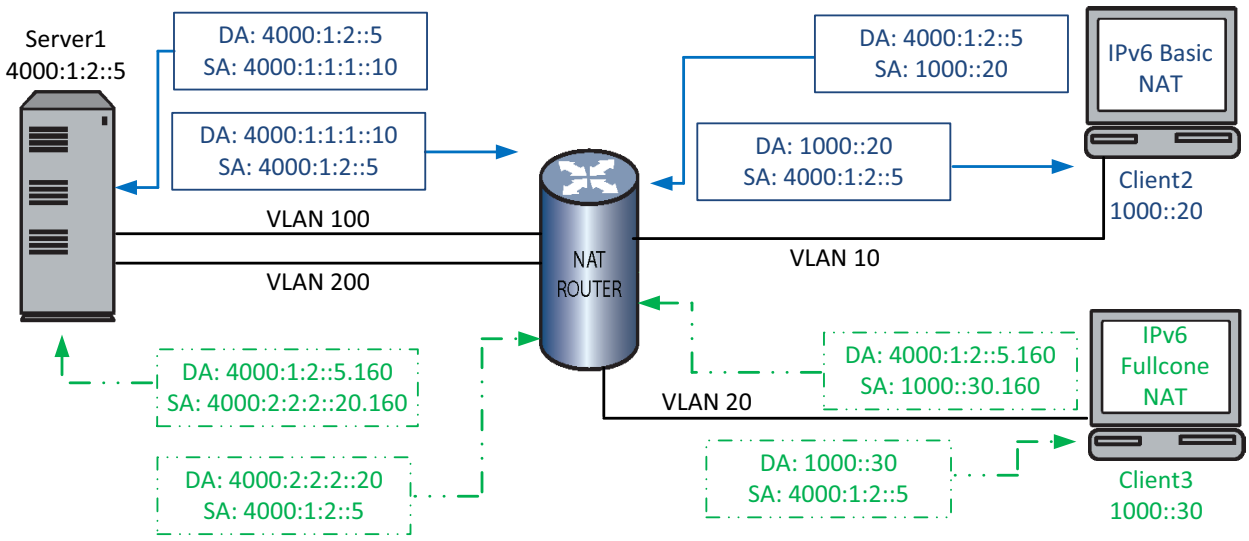


Figure 45-13 IPv6 NAT Dynamic Configuration Example



Enable NAT Inside and Outside Interfaces

Enable NAT inside interface:

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 10
S Chassis(rw-config-intf-vlan.0.10)->ip nat inside
S Chassis(rw-config-intf-vlan.0.10)->exit
S Chassis(rw-config)->interface vlan 20
S Chassis(rw-config-intf-vlan.0.20)->ip nat inside
S Chassis(rw-config-intf-vlan.0.20)->exit
S Chassis(rw-config)->
```

Enable NAT outside interface:

```
S Chassis(rw-config)->interface vlan 100
S Chassis(rw-config-intf-vlan.0.100)->ip nat outside
S Chassis(rw-config-intf-vlan.0.100)->exit
S Chassis(rw-config)->interface vlan 200
S Chassis(rw-config-intf-vlan.0.200)->ip nat outside
S Chassis(rw-config-intf-vlan.0.200)->exit
S Chassis(rw-config)->
```

Define Inside Address Access-Lists

Define IPv4 inside address access-list clientIPv4_acl for NAT clients Client1 and Client4:

```
S Chassis(rw-config)->ip access-list standard clientIPv4_acl
S Chassis(rw-cfg-std-acl)-> permit host 10.1.1.1
S Chassis(rw-cfg-std-acl)-> permit host 10.1.1.4
S Chassis(rw-cfg-std-acl)-> exit
S Chassis(rw-config)->
```

Define IPv6 inside address access-list clientIPv6_acl for NAT clients Client2 and Client3:

```
S Chassis(rw-config)->ipv6 access-list standard clientIPv6_acl
S Chassis(rw-cfg-ipv6-std-acl)-> permit host 1000::20
S Chassis(rw-cfg-ipv6-std-acl)-> permit host 1000::30
S Chassis(rw-cfg-ipv6-std-acl)-> exit
S Chassis(rw-config)->
```

Define Fullcone Access-Lists

Define IPv6 fullcone access-list cone_acl for NAT Client3:

```
S Chassis(rw-config)->ipv6 access-list extended cone_acl
S Chassis(rw-cfg-std-ext)-> permit tcp 1000::/48 range 160 168 4000:1:2::/48
S Chassis(rw-cfg-std-ext)-> exit
S Chassis(rw-config)->
```

Define the NAT Pools for Global Addresses

Define the NAT Pool for the IPv4 NAT clients:

```
S Chassis(rw-config)->ip nat pool natIPv4_pool 200.1.1.1 200.1.1.10 netmask
255.255.255.0
```

Define the NAPT Pool for the IPv4 NAPT clients:

```
S Chassis(rw-config)->ip nat pool naptIPv4_pool 200.1.1.20 200.1.1.20 netmask
255.255.255.0
```

Define the NAT Pool for the IPv6 non-Fullcone NAT clients:

```
S Chassis(rw-config)->ipv6 nat pool natIPv6_pool1 4000:1:1:1::/112 count 100
```

Define the NAT Pool for the IPv6 Fullcone NAT clients:

```
S Chassis(rw-config)->ipv6 nat pool natIPv6_pool2 4000:2:2:2::/112 count 100
```

Enable Dynamic Translation of Inside Source Addresses

Enable the NAT dynamic translation of the inside source address for Client1:

```
S Chassis(rw-config)->ip nat inside source list clientIPv4_acl pool natIPv4_pool
interface vlan.0.10
```

Enable the NAT dynamic translation of the inside source address for clients Client2 and Client3:

```
S Chassis(rw-config)->ipv6 nat inside source list clientIPv6_acl pool
natIPv6_pool1 interface vlan.0.10
S Chassis(rw-config)->ipv6 nat inside source list clientIPv6_acl pool
natIPv6_pool2 interface vlan.0.20 fullcone cone_acl
```

Enable the NAPT dynamic translation of the inside source address for Client4:

```
S Chassis(rw-config)->ip nat inside source list clientIPv4_acl pool naptIPv4_pool
overload
```

Terms and Definitions

Table 45-6 lists terms and definitions used in this NAT configuration discussion.

Table 45-6 NAT Configuration Terms and Definitions

Term	Definition
Basic NAT	Refers to Network Address Translation (NAT) only.
Dynamic Address Binding	Provides a binding based upon an internal algorithm between an address from an access-list of local addresses to an address from a pool of global addresses for NAT and TCP/UDP port number translations for NAPT.
Dynamic Nat Firewall	A NAT feature that protects members of the inside network from access from outside network clients for which a dynamic NAT firewall binding does not exist, while at the same time allowing outside traffic not destined to the inside network to flow freely.
Inside (private) address	An IP address internal to the network only reachable by the external network by translation.
List Rule (Dynamic Rule)	Defines a relation between an access-list used to match NAT inside addresses and a NAT pool to dynamically allocate NAT outside addresses from.
NAT Address Pool	A grouping of global addresses used by both NAT and NAPT dynamic address binding.
NAT Binding	Defines a logical mapping between two stations and the NAT router.
NAT Cone	Configures a NAT binding that allows any server, a specific server and any port, or a specific server and port, depending upon the configured NAT cone type, access to an internal network client using the external network address.
NAT Hairpinning	Allows an internal client to forward packets to another internal client using the destination internal client's global NAT address.
Network Address Port Translation (NAPT)	Provides a mechanism to connect a realm with private addresses to an external realm with globally unique registered addresses by mapping many network addresses, along with their associated TCP/UDP ports into a single network address and its associated TCP/UDP ports.
Network Address Translation (NAT)	Provides a mechanism to connect an internal realm with private addresses to an external realm with globally unique registered addresses by mapping IP addresses from one group to another, transparent to the end user.
Outside (public) address	A registered global IP address external to the private network that the inside address is translated to.
Static Address Binding	Provides a one-to-one binding between local addresses to global addresses for NAT and TCP/UDP port number translations for NAPT.
Static Rule	Defines a mapping between a local-ip and a global-ip with optional protocol and port definitions.
Traditional NAT	Refers to both NAT and NAPT.

46

Load Sharing Network Address Translation (LSNAT) Configuration

This document provides the following information about configuring IPv4 and IPv6 LSNAT on the Extreme Networks S-Series platform.

Using LSNAT on Your Network

LSNAT is a load balancing routing feature. It provides load sharing between multiple real servers that are grouped into server farms that can be tailored to an individual service or all services, without requiring any modification to clients or servers. Examples of well-known services are HTTP on port 80, SMTP (e-mail) on port 25, FTP on port 21, and TFTP on port 69. LSNAT is defined in RFC 2391.

The LSNAT configuration components are:

- The virtual server, configured on the LSNAT router, that intercepts the service request and determines the physical (real) server the request will be forwarded to
- The real servers that are the physical servers that makeup the server farm
- The server farm that is a logical entity containing the multiple real servers, one of which will service the client's request

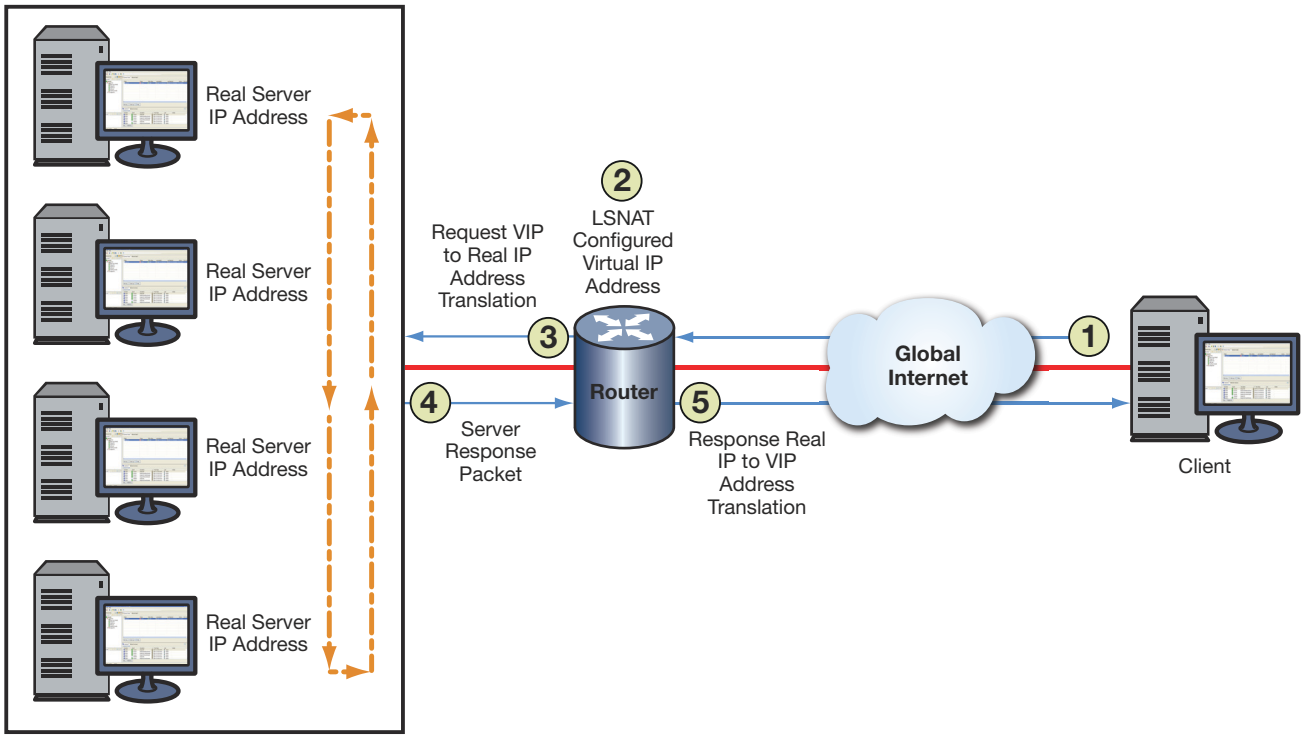
LSNAT is supported over any combination of VLAN, L3 tunnel, and L2 Tunnel interfaces.

The S-Series supports IPv4-to-IPv4, IPv6-to-IPv6, IPv4-to-IPv6, and IPv6-to-IPv4 addressing expressed as virtual server IP address (client side) to real server (server farm side). On the client side, the client and the virtual server IP address type must agree. On the server farm side, the real servers and the server farm IP address types must agree. Mixed IP addressing configuration allows for migrating an IPv4 LSNAT configuration to IPv6 by one side of the configuration (client or server) at a time.

Figure 46-1 on page 46-2 provides the following example of an LSNAT deployment:

1. A request for service is sent by the client to a virtual server. The client and the virtual server must both be either IPv4 or IPv6 addressed, – they can not be mixed.
2. The destination address for the service request is the virtual server's unique Virtual IP (VIP) address. A VIP address is defined by an IP Address (or IP Address range), IP Protocol, and UDP/TCP port number. The same IP address can be used for multiple virtual servers if a different port address is used. This is called overloading. The LSNAT configured router recognizes the VIP address and knows that LSNAT must select a real server to forward the request to.
3. Before forwarding the request, based upon the server load balancing process configured (round robin is displayed), LSNAT selects the real server for this request. LSNAT changes the destination IP address from the VIP address to the address of the selected real server member of the server farm associated with the virtual server address. The packet is then forwarded to the selected real server. The source address is taken from a configured source NAT pool.
4. The real server sends a service response back to the client with its address as the response source address.
5. At the router, LSNAT sees the real server address and knows it must first translate it back to the VIP address before forwarding the packet on to the client.

Figure 46-1 LSNAT Overview



The need for load sharing arises when a single server is not able to cope with the service demand. Legacy load sharing schemes were often ad-hoc and platform-specific, having the problem of lengthy reordering times on the servers and the inability to account for server load variations. LSNAT configuration and operation is separate from the client and servers and therefore does not care which client, server, or service is involved, or which address type is used by the client or server. It merely maps a single VIP to multiple real server IP address and port combinations, based upon a configured load balancing algorithm, and forwards packets accordingly.

With load sharing over multiple servers, reliability is increased by allowing you to take an individual server offline for scheduled maintenance, without disrupting ongoing service

operations. The servers are easily removed and replaced in the queue making maintenance a transparent activity, eliminating maintenance related downtime for the site.

Load sharing also provides redundancy in the case of a server failure. LSNAT automatically removes the failed server from the selection process. When the failed server becomes active again, LSNAT automatically adds the server back into the selection process.

Server and TCP/UDP port verification can ensure that the ports used by LSNAT are operational. TCP/UPD port service verification is capable of determining whether a server is active before creating a session. This feature eliminates the point of failure vulnerability by automatically recognizing a server is down and taking it out of the LSNAT load balancing process.

Security is improved since only the VIP is known, not the specific server addresses, ensuring that only the appropriate traffic goes to the servers.

LSNAT improves network performance by leveling traffic over many systems. Using LSNAT in conjunction with Aggregate Links removes the performance bottleneck and reliability concerns of one physical link to a server by bundling multiple links, with fail over if a link goes down. Utilizing the IP-Policy and QoS features of the S-Series device with the LSNAT feature further improves the performance and security of the network. When tied with the Virtual Redundant Router Protocol (VRRP), the network becomes even more reliable and secure.

For all these reasons, LSNAT is ideal for enterprise account web servers, application servers, or database servers.

Implementing LSNAT

To implement LSNAT in your network:

1. Configure one or more server farms by:
   - Specifying a server farm name
   - Configuring real servers as members of the server farm
   - Specifying a load balancing algorithm for each server farm
2. Configure each real server by:
   - Optionally configuring and assigning a probe(s) to monitor real server state, port verification and application content verification
   - Optionally limiting the maximum number of active connections for this real server
   - Optionally specifying a round robin weight value for this real server
   - Enabling the real server for service
3. Configure a virtual server by:
   - Specifying a virtual server name
   - Associating a virtual server with a server farm
   - Configuring a virtual server IP address (VIP)
   - Configuring a source NAT pool
   - Optionally restricting access to specific virtual server clients
   - Optionally specifying a sticky type and idle timeout
   - Enabling the virtual server for service
4. Configure global virtual server settings by:

- Optionally defining a non-standard FTP port (IPv4 only) or TFTP port to be used by virtual servers
- Optionally allowing all clients to directly access all services provided by real servers

5. Manage a real server by clearing load balancing connections or statistics

LSNAT Overview

This section provides an overview of the LSNAT components.

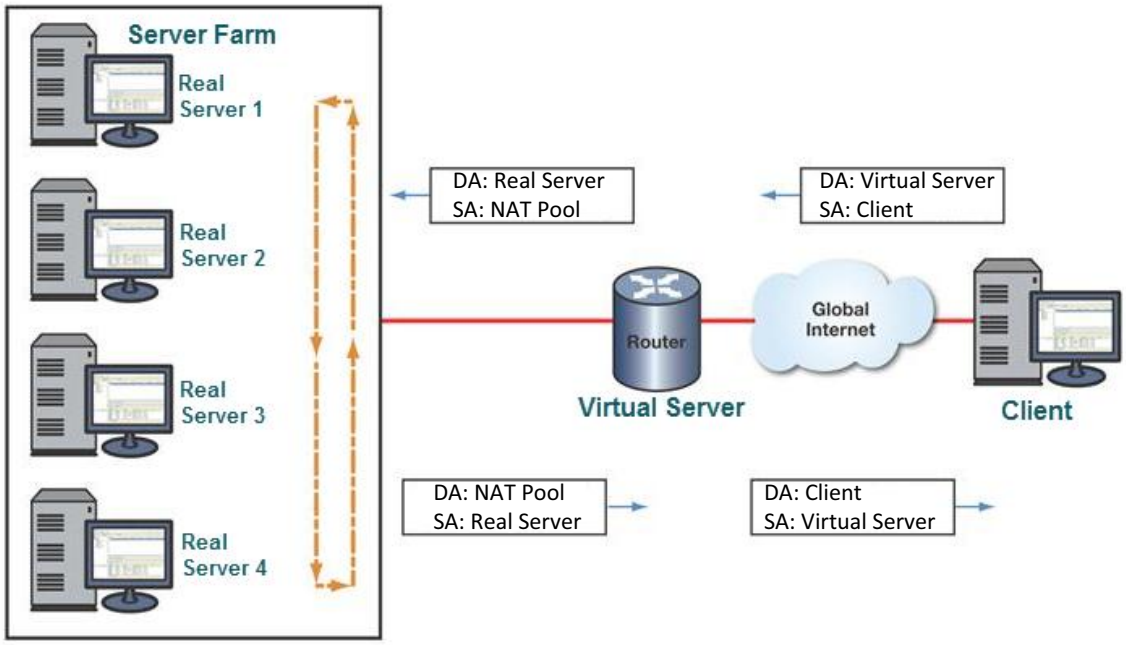
The LSNAT configuration is made up of one or more server farms, each containing multiple real servers that face the client through a configured virtual server. All aspects of an LSNAT configuration relate to the configuration or management of one of these three LSNAT components: server farm, real server, and virtual server. LSNAT components are accessible over any combination of VLAN, L3 tunnel, and L2 tunnel interfaces.

Figure 46-2 on page 46-5 presents a generic LSNAT packet flow. The actual IP address type depends upon the client and real server IP address configurations. In any case, the client and virtual server IP address type must agree, and the NAT pool and server farm address type must agree.

A request for services is sent by the client to the Virtual server IP address (VIP) on the LSNAT configured router. The source address for this request is the client IP address. The destination address for the request is the LSNAT virtual server (VIP) address. The LSNAT router recognizes the VIP address and based upon the server load balancing algorithm (round robin is displayed) LSNAT changes the destination address from the VIP address to the address of one of the real server members of the server farm associated with the VIP address. The packet is forwarded to the selected real server with a source address taken from the configured source NAT pool and the real server as the destination address.

When the real server sends a response back to the client, LSNAT sees the real server address and translates it back to the virtual server before forwarding the packet on to the client.

Figure 46-2 LSNAT Packet Flow



LSNAT IP Address Combination Support

The S-Series LSNAT implementation supports IPv4 and IPv6 addressing for both the client and real server sides of the LSNAT configuration. IPv4 and IPv6 combined configurations are expressed as LSNAT*xy* where *x* refers to the client side IP address type and *y* refers to the real server side IP address type as a **4** for IPv4 or a **6** for IPv6. For example, LSNAT46 refers to an LSNAT configuration where the client and virtual server use IPv4 addressing and the real server, server farm, and source NAT pool use IPv6 addressing.

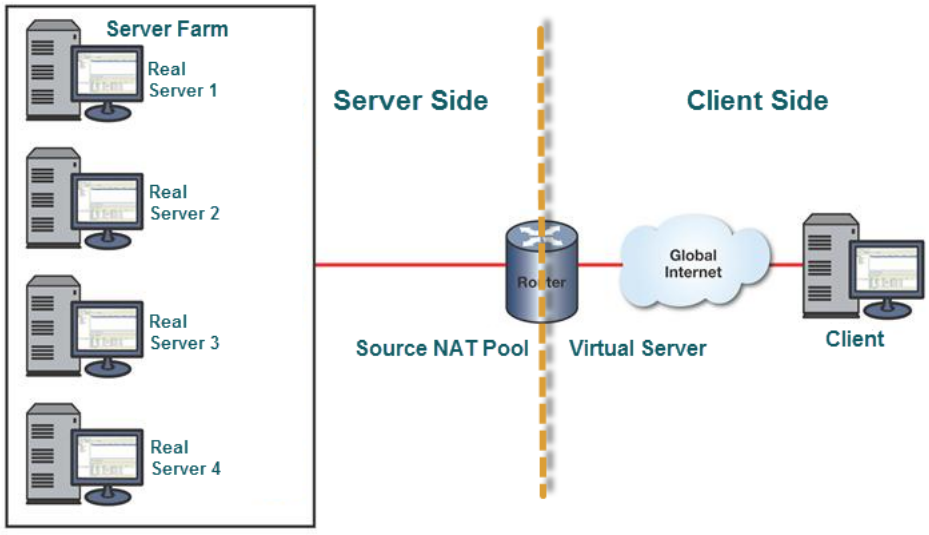
LSNAT IP address combination support allows any client access to both server farm IP address types based upon the virtual server the client is configured to. LSNAT combination IP address configuration rules are:

- The same IPv4 client can be configured to an LSNAT44 and an LSNAT46 virtual server
- The same IPv6 client can be configured to an LSNAT66 and an LSNAT64 virtual server
- The same IPv4 server farm can be configured to an LSNAT44 and an LSNAT64 virtual server
- The same IPv6 server farm can be configured to an LSNAT66 and an LSNAT46 virtual server

See "LSNAT Configuration Example" on page 46-19 for a configuration example consisting of two clients, four virtual servers, and two server farms that provides for all LSNAT IP address combinations.

Figure 46-3 displays the client and server sides of an LSNAT configuration.

Figure 46-3 LSNAT Configuration Client and Server Side Components



The LSNAT configuration client side consists of the:

- Client
- Virtual server

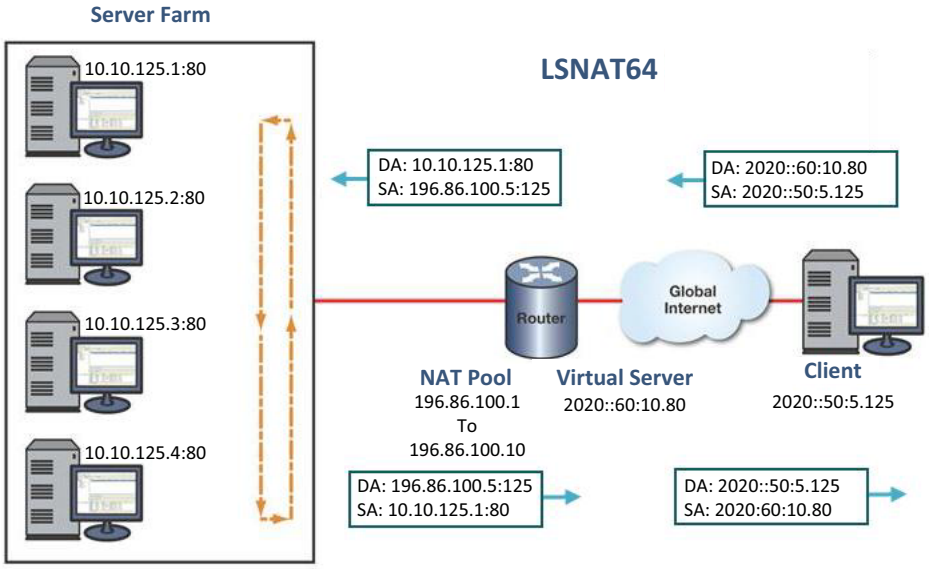
The LSNAT configuration server side consists of the:

- Server farm
- Real servers
- Source NAT pool

LSNAT components belonging to a side must be in IP address type (IPv4 or IPv6) agreement. The client and server sides do not have to be in IP address type agreement with each other. This lack of agreement between LSNAT configuration sides allows for LSNAT64 and LSNAT46 configurations.

Figure 46-4 an LSNAT64 configuration packet flow displaying destination and source addresses by IP address type. The client side is configured for IPv6 and the server side is configured for IPv4.

Figure 46-4 LSNAT64 Packet Flow Example



The packet egressing the client is source addressed to the client IPv6 address 2020::60:10.80 and destination addressed to the router configured IPv6 virtual server address 20.20::60:10.80 the client is assigned to. This LSNAT64 virtual server is configured with:

- An IPv6 virtual server address
- One or more IPv6 clients including the source client for this example
- An IPv4 server farm (an IPv4 server farm can be configured for an LSNAT44 or LSNAT64 virtual server)
- An IPv4 source NAT pool

When the packet ingresses the router, the IPv6 virtual IP address is natted to an IPv4 address from the IPV4 source NAT pool configured for the virtual server (196.86.100.5). The packet egresses the router with source NAT pool address and a destination address of one of the IPv4 real servers of the IPv4 server farm (10.10.125.1:80) assigned to the virtual server.

The packet returning from the real server has an IPv4 source address of the real server and an IPv4 destination address from the IPv4 source NAT pool. When the packet ingresses the router, the IPv4 source NAT pool address is natted to the LSNAT64 virtual server address the source NAT pool is assigned to. The packet egresses the router with an IPv6 source address of the IPv6 virtual server and an IPv6 destination address of the IPv6 client.

In the case of an LSNAT46 configuration, the IP address types for this discussion are reversed. A virtual server in an LSNAT46 configuration is configured with:

- An IPv4 virtual server address for an LSNAT46 virtual server
- One or more IPv4 clients including the source client for the LSNAT46 configuration
- An IPv6 server farm (an IPv6 server farm can be configured for an LSNAT66 or LSNAT46 virtual server)
- An IPv6 source NAT pool

IPv4 and IPv6 Address Type Configuration Differences

This sections summarizes LSNAT configuration differences between LSNAT IPv4 and IPv6 configurations.

FTP is not supported for any IPv6 LSNAT configuration. Only LSNAT44 supports FTP.

For the following IPv6 LSNAT components, the IPv6 address associated with the component must be defined in an IPv6 standard ACL using permit statements:

- Allowing client access to real servers without address translation using the **ipv6 slb real-server access client** command
- Clients assigned to an IPv6 virtual server

When assigning IPv6 addresses to the source NAT pool for LSNAT66 and LSNAT46 configurations, the prefix length must be 111 or less.

When assigning IPv4 addresses to the source NAT pool for LSNAT44 and LSNAT64 configurations, the prefix length must be 15 or less.

Table 46-1 provides a cross-reference of the IP address type used for each LSNAT component by LSNAT configuration type.

Table 46-1 LSNAT IP Address Type by LSNAT Configuration

LSNAT Configuration	Server Farm, Real Server, and Source NAT Pool IP Address Type	Client and Virtual Server IP address Type
LSNAT44	IPv4	IPv4
LSNAT66	IPv6	IPv6
LSNAT64	IPv4	IPv6
LSNAT46	IPv6	IPv4

The Server Farm

The server farm is a logical entity made up of multiple real servers. You configure a server farm by naming it and populating it with real server members. A virtual server will use the server farm to select a real server to send requests to. A server farm can be configured to any number of virtual servers. Each server farm is configured to use a load balancing algorithm. The load balancing algorithm determines the real server selection process for this server farm. The server farm defaults to a round robin load balancing algorithm.

Server Selection Process

The server selection process determines the manner in which a real server will be selected for this session. The server selection process is one of three configurable load balancing algorithms, also referred to as predictors: round robin, weighted round robin, and least connections.

Round Robin

The round robin algorithm treats all servers equally by ordering the real servers and selecting them one at a time for each new session request. When it gets to the last real server in the ordering, it starts at the beginning again.

Weighted Round Robin

Weighted round robin is the round robin algorithm that also takes into account a weight assigned to each real server. Weight is a way of accounting for the resource differences between servers. If a real server has the capacity to handle twice the number of sessions as another real server, its weight ratio to the other server can be set to 2:1. The default weight for all real servers is **1**. When all real servers are configured with the default weight, each real server is treated equally. When a

non-default weight is applied to any real servers in the server farm, the algorithm takes that weight into account when assigning sessions to the real servers.

Consider the following example. A server farm contains three real servers with the following weights: server A has a weight of **1**, server B has a weight of **2**, and server C has a weight of **3**. For each six (the sum of the three weights) active sessions, server A will be assigned 1 session, server B will be assigned 2 sessions, and server C will be assigned 3 sessions in a round robin fashion. For this example, the weight ratio between the three servers would be **1:2:3**.

Least Connections

The least connections algorithm always assigns the next session to the real server with the least number of active connections currently assigned.

Stickiness

Stickiness refers to the ability of a virtual server to associate some set of IP network tuple information to a real server.

A virtual server using stickiness will create a sticky entry when it creates a binding. The sticky entry contains a mapping of client source IP address, and optionally, destination IP and destination UDP/TCP port number, and the real server that was selected. The bindings can come and go but the sticky entries persist using a separate idle timer. When a new request is processed by a virtual server, the sticky table is checked for an entry matching the virtual server's sticky type. If an entry is found, then the load balancing algorithm is skipped and the request is mapped to the sticky entry's indicated real server.

In this way a virtual server associates particular clients to a real server for as long as the sticky entry remains in the table.

A sticky entry will only start aging when it has no associated bindings.

The Real Server

A real server is an actual physical server that is a member of a server farm. Once a real server becomes a member of a server farm, you must enable it for service. All other real server configurations are optional.

The same physical real servers may belong to multiple server farms. Each server farm is accessed by a unique virtual server.

Each real server can be optionally configured for fail detection, maximum number of active connections, and real server weight used by the weighted round robin load balancing algorithm.

Fail Detection

It is important for LSNAT to know whether a real server can provide the requested service. There are three methods supported to determine the state of a real server, server ports, and its applications:

- **Ping** - The real server is pinged.
- **TCP/UDP Port Service Verification** - The application service port is verified.
- **Application Content Verification (ACV)** - The content of an application is verified.

Fail detection methods are configured within probes using the tracked object manager facility. Probe creation and configuration is detailed, along with fail detection method details in Chapter 13, **Tracked Object Manager Configuration**.

ICMP ping probe monitoring of a real server occurs by default, using the predefined ICMP probe **$slb_default**. See "Preset Default ICMP Probes" on page 13-7 for preset default ICMP probe details.

LSNAT server load balancing supports the assigning of up to two probes per server: an ICMP ping and a UDP or TCP probe that can be configured for port verification and optionally for application content verification. Probes are assigned to a real server configuration using the **faildetect probe** command in real server configuration mode. When assigning a probe to a real server, specify probe **one** or **two**, and the name of the probe. The **$slb_default** default ICMP ping probe is auto-assigned to probe **one**, whenever probe **one** is not configured with an administratively created probe.

The probe type setting allows you to set whether configured probes are active or inactive for a server context. The probe type setting does not change the probe configuration. When probe type is set to **probe**, the probe configuration for the server context is active; probes are sent to the server in accordance with the configured settings. When probe type is set to **none**, the probe configuration is inactive; no probes are sent for the server context, and the real server is set to UP. The default probe type is **probe**. Use the **probe type** command in real server configuration mode to set the probe type for the server context.

In a server configuration context, probe configuration can be reset to factory default values by resetting fail detection for that server context. Resetting fail detection in a server configuration context:

- Sets the probe type to the default value of **probe**
- Sets the probe for probe **one** to the default probe for the server context
- Removes any configured probe configuration for probe **two**

Any preexisting probe is overwritten when assigning a probe.

This example shows how to:

- Create a TCP probe named **TCP-HTTP**
- Set the fail detection interval to **5** seconds
- Set the pass detection interval to **5** seconds
- Configure the ACV request and reply strings
- Place the probe inservice
- Display a detailed level of configuration information for the probe
- Assign the probe to probe **one** of the **10.1.2.3** port **80** real server in the server farm **myproductHTTP**:
- Enable the real server configuration

```
S Chassis(su)->configure
S Chassis(su-config)->probe TCP-HTTP tcp
S Chassis(su-config-probe)->faildetect interval 5
S Chassis(su-config-probe)->passdetect interval 5
S Chassis(su-config-probe)->acv request "GET / HTTP/1.1\\r\\nHost:
2.0.0.5\\r\\n\\r\\n"
S Chassis(su-config-probe)->acv reply "HTTP/1.1 200 OK\\r\\n"
S Chassis(su-config-probe)->inservice
S Chassis(su-config-probe)->show probe TCP-HTTP detail
Probe:                        TCP-HTTP  Type:                           tcp-acv
Administrative state:        inservice  Session count:                        1
```

```
Fail-detect count:                    3  Pass-detect count:                    3
Fail-detect interval:                 5  Pass-detect interval:                 5
3-way TCP handshake wait time:        5  Server response wait time:           10
Application Content Verification:
 Request-string: GET / HTTP/1.1\\r\\nHost: 2.0.0.5\\r\\n\\r\\n
 Reply-string:   HTTP/1.1 200 OK\\r\\n
 Close-string:
 Search-Depth:   255
S Chassis(su-config-probe)->exit
S Chassis(su-config)->ip slb serverfarm myproductHTTP
S Chassis(su-config-slb-sfarm)->real 10.1.2.3 port 80
S Chassis(su-config-slb-real)->faildetect probe one TCP-HTTP
S Chassis(su-config-slb-real)->inservice
S Chassis(su-config-slb-real)->
```

The Virtual Server

The virtual server functions as a public face to the client for the services the client wishes to access. The client accesses a service by directing service requests to the Virtual IP (VIP) address configured on the virtual server.

Before enabling a virtual server you must name it, associate it with a server farm, configure the VIP, and configure the source NAT pool. The source address is natted to a source NAT pool entry before egressing the router. Optionally, you can restrict access to the virtual server to specified clients, by specifying the sticky type.

You must configure a virtual server with a VIP. The same IP address can be used for the VIP on multiple virtual servers provided a different port is specified for each VIP.

In cases where there is only one load balancing decision made for this client to virtual server for all TCP/UDP connections, the “match source-port any” binding mode allows Server Load Balancing (SLB) connections through the virtual server to create a single binding that will match any source port the client uses destined to the same virtual server VIP address and UDP/TCP port. Match source port any is only supported in an IPv4 context. Configure the “match source-port any” binding mode using the **binding match source-port** command.

Configuring Direct Access to Real Servers

When the LSNAT router has been configured with server farms, real servers, and virtual servers and these LSNAT components have been placed “in service,” the real servers are protected from direct client access for all services.

If you want to provide direct client access to real servers configured as part of a server farm, there are two mechanisms that can provide direct client access.

The first mechanism allows you to identify specific client networks that can set up connections directly to a real server’s IP address, as well as continue to use the virtual server IP address. This method is configured in global configuration mode with the **ip slb real-server access client** command for an IPv4 real server and the **ipv6 slb real-server access client** command for an IPv6 real server.

The second mechanism allows all clients to directly access all services provided by real servers, except for those services configured for server load balancing. This method is configured in global configuration mode with the **ip slb real-server access unrestricted** command for an IPv4 real server and the **ipv6 slb real-server access unrestricted** command for an IPv6 real server.

The Source NAT Pool

LSNAT supports Network Address Translating (NAT) of the client IP address as described in Section 3.3 of RFC 2391. See Chapter 45, **Network Address Translation (NAT) Configuration** for NAT configuration details.

With a standard LSNAT connection, the client's IP address is passed through the router un-natted. The consequence of this is that the real server must have a route for the client IP address that returns traffic back through the LSNAT router. Since the client IP addresses are usually unknown to the real server, most real servers end up setting their default router to the LSNAT router. If the LSNAT router is not configured as the default router, the LSNAT router and real server must be located somewhere in the network topology that guarantees that return traffic flows through the LSNAT router.

If instead, the client IP address is natted, this allows the real servers to be located anywhere in a network, since the packets from router to real-server will be source natted with an IP address owned by the router itself. Client IP addresses must be natted for non-LSNAT44 configurations.

When assigning IPv6 addresses to the source NAT pool, the prefix length must be 111 or less.

When assigning IPv4 addresses to the source NAT pool, the prefix length must be 15 or less.

Note: In an LSNAT44 configuration, specifying a source NAT pool is optional. If the source NAT pool is not specified for any IPv6 LSNAT configuration type, the virtual server will not become active. See "LSNAT IP Address Combination Support" on page 46-5 for a discussion of LSNAT configuration types.

Use the **source nat pool** command to specify an IP address and prefix length or a NAT pool to use for source NATing. The NAT pool is used in an overload mode. Specifying an IP address and prefix length is supported for all LSNAT IP address combinations.

The FTP and TFTP Control Port

The FTP port assignment defaults to port 21 and is only supported in an LSNAT44 configuration. The TFTP port assignment defaults to port 69 and is supported for all LSNAT IP address combinations. You can globally assign a non-standard FTP or TFTP control port in global configuration mode that will be used by all virtual servers.

The Virtual Server, Virtual Port, and Real Server Port

When configuring a virtual server and real server, the port must be configured for a protocol type and port value. This section specifies port protocol and port value considerations to take into account when configuring a virtual server or real server.

Virtual Server Virtual Port

The configuration of the virtual server virtual port has two meanings depending upon whether the port has a zero or non-zero value:

- If a non-zero value is set, then incoming packets' destination ports are matched to that port.
- If a zero value is set, then the incoming packets' destination ports will only match that virtual server if there is no non-zero port match with another virtual server. In this case the zero port is a catch all that means match any port.

The virtual server virtual port protocol (UDP/TCP) must always match the real server port protocol.

The virtual server is identified by its Virtual IP Address (VIP), port protocol, and port number. A virtual server configured for a given VIP and port number must be configured for either UDP or TCP, but can not be configured for both.

Real Server Port

The configuration of the real server port has two meanings:

- If a non-zero value is set to the real server port, then any bindings created using that real server will use the real server's destination port.
- If a zero value is set to the real server port, then any bindings created using that real server will use the client's original destination port.

If the real server's port is set to 0, the only valid fail detect types for the real server is none or ping.

Managing Connections and Statistics

There are three aspects to managing connections:

- Clearing all LSNAT counters and bindings or selectively clearing bindings based on ID or matching network tuple information (sip, sport, dip, dport).
- Setting LSNAT limits for the number of bindings, cache size, and number of configurations.
- Displaying LSNAT statistics.

Configuring UDP-One-Shot

Many UDP applications send only two packets in the form of a request and a reply. For such applications it is a waste of resources to set up a new binding and hardware connection for every request and then let each binding idle age out. With UDP-one-shot configured, a binding is created and the request packet is sent. The reception of a reply packet back causes the binding to be deleted within one second. Bindings created by UDP-one-shot will not result in the installation of a hardware connection.

Use the **udp-one-shot** command in SLB virtual server configuration command mode to enable UDP-one-shot on a virtual server.

Configuring LSNAT

This section provides details for the configuration of LSNAT on the S-Series products.

Table 46-2 lists LSNAT parameters and their default values.

Table 46-2 Default LSNAT Parameters

Parameter	Description	Default Value
Port Number (FTP)	The port number for the FTP control port for all IPv4 virtual servers. (IPv4 only)	21
Port Number (TFTP)	The port number for the TFTP control port for all virtual servers	69
Predictor	The load balancing algorithm for this server farm.	Round Robin

Table 46-2 Default LSNAT Parameters (continued)

Parameter	Description	Default Value
Faildetect probe one and two	Default probe for server load balancing faildetect probe one and two.	probe one: $slb_default probe two: empty
Faildetect Type	Specifies whether the current fail detection configuration is active (**probe**) or inactive (**none**) for the real server context.	probe
FIN/RST idle time	The idle time in seconds after the TCP finish reset (FIN/RST) is observed on an IPv4 or IPv6 NAT binding.	3 seconds
Match Source-Port Binding Mode	Use this command to set the source port to virtual server binding behavior for this virtual server.	exact
Maximum Connections	Specifies the maximum number of connections allowed to an LSNAT real server.	Unlimited
Weight	Specifies a real server weight value for the weighted round robin load balancing algorithm.	1
Service Type	A special service type, such as FTP or TFTP, if the virtual port number is different than the default for that service.	None
Stickiness Type	The type of stickiness to use for the virtual server.	None
Sticky Timeout	Specifies the age out interval for sticky entries that have no associated bindings.	SIP: 7200 seconds SIP DIP-PORT: 7200 seconds

Table 46-3 lists LSNAT resource limits.

Table 46-3 LSNAT Resource Limits

Resource	S-Series	SSA
Bindings	65536	131072
Reals	800	800
Server Farms	400	400
Sticky Entries	65536	65536
VIP Addresses	1000	1000
Virtual Servers	500	500

Configuring an LSNAT Server Farm

Procedure 46-1 describes how to configure an LSNAT server farm.

Procedure 46-1 LSNAT Server Farm Configuration

Step	Task	Command(s)
1.	In global router configuration command mode, specify a name for this server farm for the IPv4 or IPv6 context.	{**ip** | **ipv6**} **slb serverfarm** *serverfarmname*
2.	In SLB server farm configuration command mode, specify the load balancing algorithm for this IPv4 or IPv6 server farm.	**predictor** [**roundrobin** | **leastconns**]
3.	In SLB server farm configuration command mode, enable the this IPv4 or IPv6 server farm. The default setting for server farms is inservice.	**inservice**

Configuring an LSNAT Real Server

Procedure 46-2 describes how to configure an LSNAT real server.

Procedure 46-2 Configuring an LSNAT Real Server

Step	Task	Command(s)
1.	In SLB server farm configuration command mode, configure the real server members for the IPv4 or IPv6 server farm and enter SLB real server configuration command mode in which the remaining commands in this procedure are configured.	**real** *ip-address* [**port** *number*]
2.	Optionally, apply a configured probe to probe **one** or probe **two** to monitor this IPv4 or IPv6 real server. An ICMP ping and TCP or UDP probe can be configured on separate command lines.	**faildetect probe** {**one** | **two**} *probe-name*
3.	Optionally, specify whether the currently configured probes are active or inactive for this IPv4 or IPv6 real server.	**faildetect type** {**none** | **probe**}
4.	Optionally, reset fail detection configuration to the factory default settings for this IPv4 or IPv6 real server.	**faildetect reset**
5.	Optionally, limit the maximum number of active connections for this IPv4 or IPv6 real server.	**maxconns** *maximum-number*
6.	Optionally configure a weight for this IPv4 or IPv6 real server to be used by the round robin load balancing algorithm.	**weight** *weight-number*
7.	Enable each IPv4 or IPv6 real server for service.	**inservice**
8.	Exit the LSB real server configuration command mode.	**exit**

Configuring an LSNAT Virtual Server

Procedure 46-3 describes how to configure an LSNAT virtual server.

Procedure 46-3 Configuring an LSNAT Virtual Server

Step	Task	Command(s)
1.	In global router configuration command mode, specify a name for this IPv4 or IPv6 virtual server. The virtual server IP address context must match the client context.	{**ip** | **ipv6**} **slb vserver** *vserver-name*
2.	In SLB virtual server configuration command mode, optionally specify a match source port to virtual server binding behavior. Any is not supported by IPv6. (Default = exact).	**binding match source-port** {**any** | **exact**}
3.	In SLB virtual server configuration command mode, associate this IPv4 or IPv6 virtual server with a server farm. The server farm IP address context can be either IPv4 or IPv6.	**serverfarm** *serverfarm-name*
4.	In SLB virtual server configuration command mode, configure the virtual server IP address (VIP) or proceed to the next step and configure a range of virtual server IP addresses. You must specify whether the VIP uses TCP or UDP. For TCP ports you can optionally specify the FTP service; for UDP ports you can optionally specify the TFTP service. The virtual IP address type must agree with the client IP address type.	**virtual** *ip-address* {**tcp** | **udp**} *port* [**service** *service-name*] [**all-vrfs**]
5.	In SLB virtual server configuration command mode, if you did not configure a VIP in the preceding step, configure a range of virtual server IP addresses. You must specify whether the VIPs will use TCP or UDP. For TCP ports you can optionally specify the FTP service; for UDP ports you can optionally specify TFTP service. The virtual IP address type must agree with the client IP address type.	**virtual-range** *start-address end-address* {**tcp** | **udp**} *port* [**service** *service-name*] [**all-vrfs**]
6.	In SLB virtual server configuration command mode, optionally configure a client source NAT pool to source NAT the traffic through the virtual server with the IP addresses from the NAT pool for an LSNAT44 configuration. For any IPv6 LSNAT configuration (LSNAT46, LSNAT64, or LSNAT66) you must configure a source NAT pool specifying an IP address and prefix length.	**source nat pool** {*poolname* | *ip-address/prefix-len*}
7.	In SLB virtual server configuration command mode, optionally set the number of seconds of idle time to elapse before a binding will be deleted for both an IPv4 or IPv6 virtual server configuration. (Default = 240 seconds).	**idle timeout** *timeperiod*
8.	In SLB virtual server configuration command mode, enable the virtual server for service	**inservice**

Procedure 46-3 Configuring an LSNAT Virtual Server (continued)

Step	Task	Command(s)
9.	In SLB virtual server configuration command mode, optionally configure this IPv4 or IPv6 virtual server to participate in VRRP state changes. Specify the VLAN on which the VRRP is configured and the virtual router ID associated with the routing interface for this VRRP.	**vrrp vlan** *vlan vrid*
10.	In SLB virtual server configuration command mode, optionally restrict access to this IPv4 or IPv6 virtual server to configured clients. In an IPv6 virtual server context an ACL list must be specified.	**client** {*ip-address network-mask* | *ip-address/prefixlength* | *acl-list*}
11.	In SLB virtual server configuration command mode, optionally configure UDP application connections to delete the binding when the reply packet is received. Bindings created by UDP-one-shot will not result in the installation of a hardware connection.	**udp-one-shot**
12.	In SLB virtual server configuration command mode, optionally configure the stickiness type.	**sticky type** [**sip** | **sip dip-dport**]
13.	In SLB virtual server configuration command mode optionally configure the sticky entry timeout value for this virtual server.	**sticky timeout** *timeperiod*
14.	Exit the SLB virtual server configuration command mode to get to global configuration command mode.	**exit**
15.	In global configuration command mode, optionally allow specific clients to access the load balancing IPv4 or IPv6 real servers in a particular LSNAT server farm without address translation.	**ip slb real-server access client** {*ip-address mask* | *ip-prefix/length* | *acl-list*} **ipv6 slb real-server access client** *acl-list*
16.	In global configuration command mode, allow all clients to access the IPv4 or IPv6 real servers directly without restriction.	{**ip** | **ipv6**} **slb real-server access unrestricted**
17.	In global configuration command mode, configure the router to return a TCP RST (reset) packet when a client tries to access an IPv4 or IPv6 real server directly on a TCP port used by LSNAT.	{**ip** | **ipv6**} **slb real-server access tcp-reset**
18.	Optionally clear sticky entries or remove bindings.	**clear ip slb** {**sticky** | **bindings**} {**all** | **id** *id* | **match** {*sip* | *****} {*sport* | *****} {*dip* | *****} {*dport* | *****}}

Configuring Global Settings

Table 46-4 describes how to configure LSNAT global settings.

Table 46-4 Configuring LSNAT Global Settings

Task	Command(s)
In global configuration command mode, optionally specify a non-default FTP control port for all IPv4 virtual servers. FTP is not supported for IPv6. (Default = 21).	**ip slb ftpctrlport** *port-number*
Optionally specify a non-default TFTP control port for all IPv4 or IPv6 virtual servers. (Default = 69).	{**ip** | **ipv6**} **slb tftpctrlport** *port-number*
Optionally specify an idle time in seconds after the TCP finish reset (FIN/RST) is observed on an IPv4 or IPv6 NAT binding. (Default = 3 seconds).	{**ip** | **ipv6**} **slb binding finrst-timeout** {[*idle-time*] [**apply-to-half-closed**]}
Optionally disable the TCP FIN/RST idle timer for IPv4 or IPv6 connections.	{**ip** | **ipv6**} **slb binding finrst-timeout disabled**
Optionally allow all IPv4 or IPv6 clients to directly access all services provided by real servers, except for those services configured for server load balancing.	{**ip** | **ipv6**} **slb real-server access unrestricted**
Optionally, allow specific IPv4 or IPv6 client networks to access the real servers without address translation.	**ip slb real-server access client** {*ip-address mask* | *ip-prefix/length* | *acl-list*} **ipv6 slb real-server access client** *acl-list*

Displaying LSNAT Configuration Information and Statistics

Table 46-5 describes how to display LSNAT configuration information and statistics.

Table 46-5 Displaying LSNAT Configurations and Statistics

Task	Command(s)
Display the specified or all server farm configurations	**show** {**ip** | **ipv6**} **slb serverfarms** [**detail** | *serverfarmname*]
Display all real server configurations for this system or those for the specified server farm.	**show** {**ip** | **ipv6**} **slb reals** [**detail** | **serverfarm** *serverfarmname* [**detail**]]
Display all or the specified virtual servers for this system.	**show** {**ip** | **ipv6**} **slb vservers** [**detail** | *virtserver-name*]
Display server load balancing statistics.	**show** {**ip** | **ipv6**} **slb statistics** [**-all_vrfs**] [**-interesting**]
Display SLB bindings.	**show** {**ip** | **ipv6**} **slb bindings** {**match** [*ip-address* | *****] | **id** *id* | **summary**}
Display LSNAT configuration information.	**show** {**ip** | **ipv6**} **slb info**

Table 46-5 Displaying LSNAT Configurations and Statistics (continued)

Task	Command(s)
Display active server load balancing sticky mode connections.	**show** {**ip** | **ipv6**} **slb sticky** {**match** *sip port dip port* | **id** *id* | **summary**}
Display sticky statistics.	**show** {**ip** | **ipv6**} **slb statistics-sticky**

LSNAT Configuration Example

This section provides an LSNAT configuration example that includes both an IPv4 and IPv6 server farm. The real servers belonging to each server farm will service both HTTP and SMTP requests from clients configured with the same IP address type as the server farm.

Two virtual servers are configured for each server farm:

- An HTTP virtual server for internet traffic on port 80
- An SMTP virtual server for e-mail traffic on port 25

An IPv4 source NAT pool provides natted source addresses for packets forwarded to the IPv4 server farm. An IPv6 source NAT pool provides natted source addresses for the packets forwarded to the IPv6 server farm.

The real servers configured on the IPv6 server farm, named **serverFarmIPv6**, will handle all HTTP and SMTP requests from clients on network 2020::50:5/64. HTTP requests will use virtual server **virtualServerIPv6-80** configured with IP address of 2020::60:10.80. SMTP requests will use virtual server **virtualServerIPv6-25** configured with IP 2020::60:20.25. All requests will be source natted using the IPv6 source NAT pool 2020::65:0/111.

The real servers configured on the IPv4 server farm, named **serverFarmIPv4**, will handle all HTTP and SMTP requests from clients on network 196.86.100.12/32. HTTP requests will use virtual server **virtualServerIPv4-80** configured with IP address of 184.56.13.2:80. SMTP requests will use virtual server **virtualServerIPv4-25** configured with IP 184.56.13.3:25. All requests will be source natted using the IPv4 source NAT pool 196.86.100.1/15.

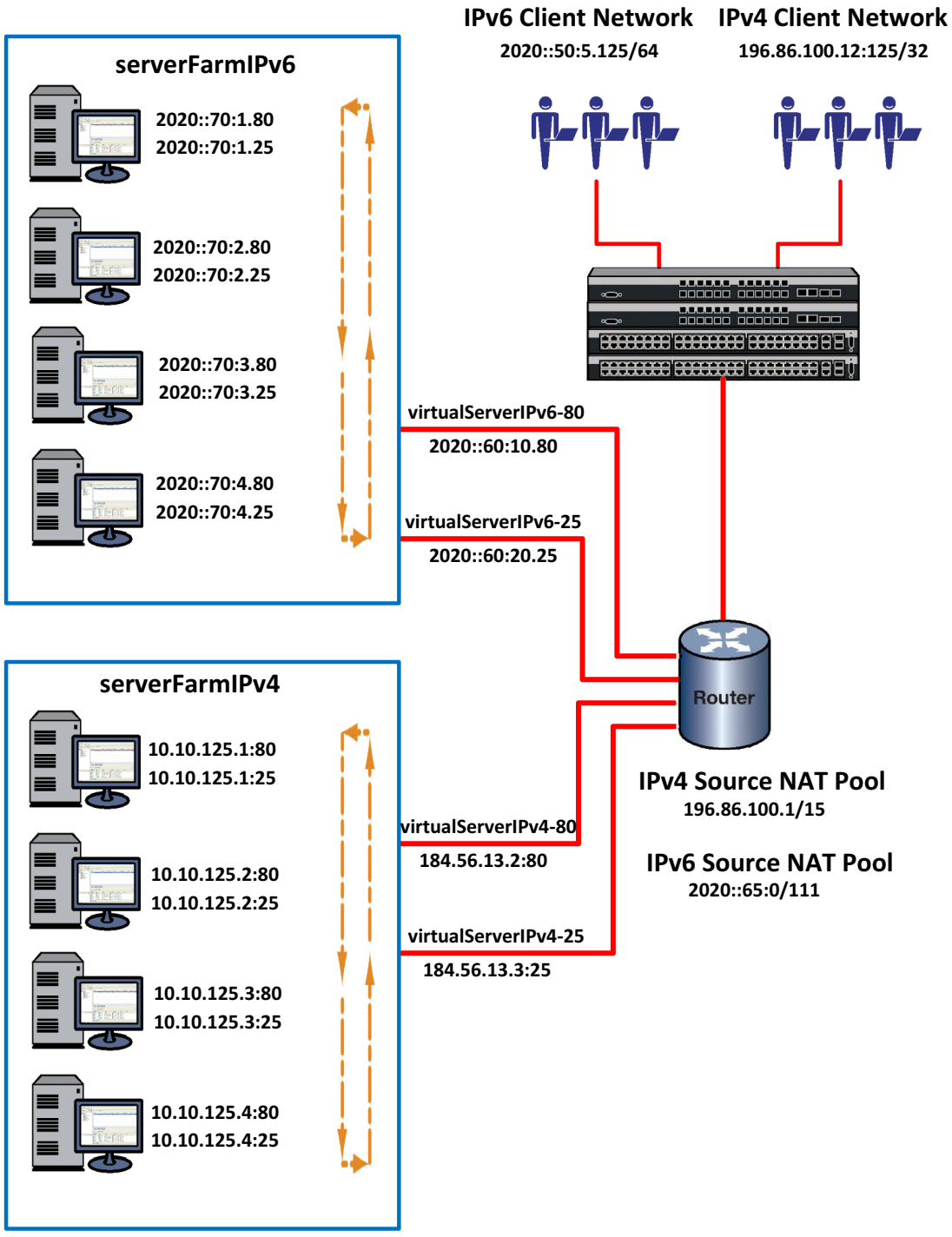
Each real server, for both server farms will:

- Apply a TCP-HTTP probe that verifies port 80 and uses Application Content Verification TCP fail detection
- Apply both the default ICMP ping probe and a TCP-SMTP probe for verification of port 25

Real servers will be configured for the round robin predictor. Real servers in the IPv4 server farm will use weighted round robin with a ratio of 2:3:2:3. This weighted round robin selection process takes into account the resource differences between the four servers in the IPv4 server farm.

See Figure 46-5 on page 46-20 for a graphic presentation of this LSNAT configuration example.

Figure 46-5 LSNAT Configuration Example



Configuring the serverFarmIPv6 Server Farm and Real Servers

Configure the **serverFarmIPv6** server farm by:

- Naming the server farm **serverFarmIPv6**
- Configuring round robin as the load balancing algorithm for this server farm (weight will be be the default: each real server is treated equally)

Configure the real servers on the **serverFarmIPv6** server farm by:

- Configuring probe **TCP-HTTP** for application content verification and search-depth, modifying the faildetect and passdetect intervals, applying the probe to probe **two** of each HTTP server, and using the default ICMP ping probe in probe **one**
- Configuring the following real servers: 2020::70:1.80 through 2020::70:4.80 and 2020::70:1.25 through 2020::70:4.25
- Enabling each real server by placing each server in service

Note: We will not modify the maximum number of active connections allowed on any real server for this configuration example.

serverFarmIPv6 Server Farm and Real Server CLI Input

```
S Chassis(rw)->configure
S Chassis(su-config)->probe TCP-HTTP tcp
S Chassis(su-config-probe)->faildetect interval 5
S Chassis(su-config-probe)->passdetect interval 5
S Chassis(su-config-probe)->acv request "GET / HTTP/1.1\\r\\nHost:
2.0.0.5\\r\\n\\r\\n"
S Chassis(su-config-probe)->acv reply "HTTP/1.1 200 OK\\r\\n"
S Chassis(su-config-probe)->acv search-depth 50
S Chassis(su-config-probe)->inservice
S Chassis(su-config-probe)->exit
S Chassis(rw)->configure
S Chassis(su-config)->probe TCP-SMTP tcp
S Chassis(su-config-probe)->faildetect interval 5
S Chassis(su-config-probe)->passdetect interval 5
S Chassis(su-config-probe)->inservice
S Chassis(su-config-probe)->exit
S Chassis(rw-config)->ipv6 slb serverfarm serverFarmIPv6
S Chassis(rw-config-slb-sfarm)->predictor roundrobin
S Chassis(rw-config-slb-sfarm)->real 2020::70:1 port 80
S Chassis(rw-config-slb-real)->faildetect probe two TCP-HTTP
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
S Chassis(rw-config-slb-sfarm)->real 2020::70:1 port 25
S Chassis(rw-config-slb-real)->faildetect probe two TCP-SMTP
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
S Chassis(rw-config-slb-sfarm)->real 2020::70:2 port 80
```

```
S Chassis(rw-config-slb-real)->faildetect probe two TCP-HTTP
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
S Chassis(rw-config-slb-sfarm)->real 2020::70:2 port 25
S Chassis(rw-config-slb-real)->faildetect probe two TCP-SMTP
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
S Chassis(rw-config-slb-sfarm)->real 2020::70:3 port 80
S Chassis(rw-config-slb-real)->faildetect probe two TCP-HTTP
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
S Chassis(rw-config-slb-sfarm)->real 2020::70:3 port 25
S Chassis(rw-config-slb-real)->faildetect probe two TCP-SMTP
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
S Chassis(rw-config-slb-sfarm)->real 2020::70:4 port 80
S Chassis(rw-config-slb-real)->faildetect probe two TCP-HTTP
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
S Chassis(rw-config-slb-sfarm)->real 2020::70:4 port 25
S Chassis(rw-config-slb-real)->faildetect probe two TCP-SMTP
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
```

Configuring virtualServerIPv6-80 and -25 Virtual Servers

Configure the virtual servers for the **serverFarmIPv6** server farm by:

- Creating a standard IPv6 access list to permit IPv6 clients to use this virtual server
- Naming the virtual servers **virtualServerIPv6-80** and **virtualServerIPv6-25**
- Associating the virtual servers with the **serverFarmIPv6** server farm
- Assigning the virtual server IP addresses and ports
- Setting the idle timeout value of 360 seconds
- Configuring the source NAT pool
- Configure the clients that will be permitted access to the virtual server
- Placing the virtual server in service

IPv6 Virtual Server CLI Input

```
S Chassis(rw-config)->ipv6 access-list standard ipv6Clients
S Chassis(rw-cfg-ipv6-std-acl)->permit 2020::50:5/64
S Chassis(rw-cfg-ipv6-std-acl)->exit
S Chassis(rw-config)->ipv6 slb vserver virtualServerIPv6-80
S Chassis(rw-config-slb-vserver)->serverfarm serverFarmIPv6
S Chassis(rw-config-slb-vserver)->virtual 2020::60:10 tcp port 80
```

```
S Chassis(rw-config-slb-vserver)->idle timeout 360
S Chassis(rw-config-slb-vserver)->source nat pool 2020::65:0/111
S Chassis(rw-config-slb-vserver)->client ipv6Clients
S Chassis(rw-config-slb-vserver)->inservice
S Chassis(rw-config-slb-vserver)->exit
S Chassis(rw-config)->ipv6 slb vserver virtualServerIPv6-25
S Chassis(rw-config-slb-vserver)->serverfarm serverFarmIPv6
S Chassis(rw-config-slb-vserver)->virtual 2020::60:10 tcp port 25
S Chassis(rw-config-slb-vserver)->idle timeout 360
S Chassis(rw-config-slb-vserver)->source nat pool 2020::65:0/111
S Chassis(rw-config-slb-vserver)->client ipv6Clients
S Chassis(rw-config-slb-vserver)->inservice
S Chassis(rw-config-slb-vserver)->exit
```

Configuring the serverFarmIPv4 Server Farm and Real Servers

Configure the **serverFarmIPv4** server farm by:

- Naming the server farm **serverFarmIPv4**
- Configuring round robin as the load balancing algorithm for this server farm (in real server mode, weight will be configured for a ratio of 2:3:2:3 for the four real servers, to take into account the resources available on the servers)

Configure the real servers on the **serverFarmIPv4** server farm by:

- Configuring probe **TCP-HTTP** for application content verification and search-depth, modifying the faildetect and passdetect intervals, applying the probe to probe **two** of each HTTP server, and using the default ICMP ping probe in probe **one**
- Configuring the following real servers: 10.10.125.1:80 through 10.10.125.4:80 and 10.10.125.1:25 through 10.10.125.4:25
- Configuring the real server weighted round robin ratio to 2:3:2:3
- Enabling each real server by placing each server in service

serverFarmIPv4 Server Farm and Real Server CLI Input

```
S Chassis(rw)->configure
S Chassis(su-config)->probe TCP-HTTP tcp
S Chassis(su-config-probe)->faildetect interval 5
S Chassis(su-config-probe)->passdetect interval 5
S Chassis(su-config-probe)->acv request "GET / HTTP/1.1\\r\\nHost:
2.0.0.5\\r\\n\\r\\n"
S Chassis(su-config-probe)->acv reply "HTTP/1.1 200 OK\\r\\n"
S Chassis(su-config-probe)->acv search-depth 50
S Chassis(su-config-probe)->inservice
S Chassis(su-config-probe)->exit
S Chassis(rw-config)->ip slb serverfarm serverFarmIPv4
S Chassis(rw-config-slb-sfarm)->predictor roundrobin
S Chassis(rw-config-slb-sfarm)->real 10.10.125.1 port 80
S Chassis(rw-config-slb-real)->faildetect probe two TCP-HTTP
```

```
S Chassis(rw-config-slb-real)->weight 2
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
S Chassis(rw-config-slb-sfarm)->real 10.10.125.1 port 25
S Chassis(rw-config-slb-real)->faildetect probe two TCP-SMTP
S Chassis(rw-config-slb-real)->weight 2
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
S Chassis(rw-config-slb-sfarm)->real 10.10.125.2 port 80
S Chassis(rw-config-slb-real)->faildetect probe two TCP-HTTP
S Chassis(rw-config-slb-real)->weight 3
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
S Chassis(rw-config-slb-sfarm)->real 10.10.125.2 port 25
S Chassis(rw-config-slb-real)->faildetect probe two TCP-SMTP
S Chassis(rw-config-slb-real)->weight 3
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
S Chassis(rw-config-slb-sfarm)->real 10.10.125.3 port 80
S Chassis(rw-config-slb-real)->faildetect probe two TCP-HTTP
S Chassis(rw-config-slb-real)->weight 2
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
S Chassis(rw-config-slb-sfarm)->real 10.10.125.3 port 25
S Chassis(rw-config-slb-real)->faildetect probe two TCP-SMTP
S Chassis(rw-config-slb-real)->weight 2
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
S Chassis(rw-config-slb-sfarm)->real 10.10.125.4 port 80
S Chassis(rw-config-slb-real)->faildetect probe two TCP-HTTP
S Chassis(rw-config-slb-real)->weight 3
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
S Chassis(rw-config-slb-sfarm)->real 10.10.125.4 port 25
S Chassis(rw-config-slb-real)->faildetect probe two TCP-SMTP
S Chassis(rw-config-slb-real)->weight 3
S Chassis(rw-config-slb-real)->inservice
S Chassis(rw-config-slb-real)->exit
```

Configuring virtualServerIPv4-80 and -25 Virtual Servers

Configure the virtual servers for the **serverFarmIPv4** server farm by:

- Naming the virtual servers **virtualServerIPv4-80** and **virtualServerIPv4-25**
- Associating the virtual servers with the **serverFarmIPv4** server farm

- Assigning the virtual server IP addresses and ports
- Setting the idle timeout value of 360 seconds
- Configuring the source NAT pool
- Configure the clients that will be permitted access to the virtual server
- Placing the virtual server in service

IPv6 Virtual Server CLI Input

```
S Chassis(rw-config)->ip slb vserver virtualServerIPv4-80
S Chassis(rw-config-slb-vserver)->serverfarm serverFarmIPv4
S Chassis(rw-config-slb-vserver)->virtual 184.56.13.2 tcp port 80
S Chassis(rw-config-slb-vserver)->idle timeout 360
S Chassis(rw-config-slb-vserver)->source nat pool 196.86.100.1/15
S Chassis(rw-config-slb-vserver)->client 196.86.100.12/32
S Chassis(rw-config-slb-vserver)->inservice
S Chassis(rw-config-slb-vserver)->exit
S Chassis(rw-config)->ip slb vserver virtualServerIPv4-25
S Chassis(rw-config-slb-vserver)->serverfarm serverFarmIPv4
S Chassis(rw-config-slb-vserver)->virtual 184.56.13.3 tcp port 25
S Chassis(rw-config-slb-vserver)->idle timeout 360
S Chassis(rw-config-slb-vserver)->source nat pool 196.86.100.1/15
S Chassis(rw-config-slb-vserver)->client 196.86.100.12/32
S Chassis(rw-config-slb-vserver)->inservice
S Chassis(rw-config-slb-vserver)->exit
```

This completes the LSNAT configuration example.

Terms and Definitions

Table 46-6 lists terms and definitions used in this LSNAT configuration discussion.

Table 46-6 LSNAT Configuration Terms and Definitions

Term	Definition
application content verification (ACV)	A fail detection method for the verification of application content on a server.
binding	A resource that tracks a connection from client to the LSNAT router and from the LSNAT router to the real server.
ICMP ping	A fail detection method that sends a ping packet to the IP address of the remote service before a session is created.
least connections	A load balancing algorithm that assigns sessions based upon the server in the pool with the least current active sessions assigned.
load balancing	An LSNAT feature that assigns sessions over multiple real servers based upon a configured predictor.
LSNAT	LSNAT is a load balancing routing feature that provides load sharing between multiple servers grouped into server farms. LSNAT can be tailored to individual services or all services.

Table 46-6 LSNAT Configuration Terms and Definitions (continued)

Term	Definition
port service verification	A tracked object manager fail detection feature that assures that the protocol port is in an up state before beginning a session.
predictor	A load balancing (sharing) algorithm such as round robin, weighted round robin and least connection.
probe	A tracked object manager object of protocol type ICMP, UDP, or TCP that tracks the availability of a remote service, by actively transmitting network packets to a specified remote host.
probe one and two	Up to two probes, that can be a default probe or administratively created probe, labelled **one** and **two**, applied to a server context.
real server	The actual physical server that provides the services requested by the client.
request packet	A data packet sent by the client to the virtual server requesting services.
response packet	A data packet sent by the real server to the service requesting client.
server farm	A logical entity of multiple real servers that faces the client through a virtual server.
session sticky type	The concept that the client will be directed to the same physical server for the duration of a session based upon a configured binding type (TCP, SIP, or SIP DPORT).
simple round robin	A load balancing algorithm that assigns sessions based upon an equal weight ordering of the servers. When all servers in the ordering have been assigned a session, the algorithm returns to the first server in the server list.
sticky mode	An LSNAT feature that assures all service requests from a particular client will be directed to the same real server for that session.
tracked object manager	An application that determines the state of a remote service using administratively configured and default probes.
Virtual IP (VIP) address	The IP address of the LSNAT virtual server that functions as the public face of the real server.
virtual server	A logical entity that the client interacts with by acting as the public face for the real server.
weighted round robin	A load balancing algorithm that assigns sessions based upon the configured server weight. For instance, if there are two servers the first of which has a weight of 2 and the second has a weight of 3, then for every 5 sessions, the first will be assigned 2 sessions and the second will be assigned 3 sessions.

47

Transparent Web Cache Balancing (TWCB) Configuration

This document provides the following information about configuring Transparent Web Cache Balancing on the Extreme Networks S-Series platform.

Using Transparent Web Cache Balancing (TWCB) on Your Network

Transparent Web Caching is a means of transparently redirecting a client's IPv4 or IPv6 HTTP traffic to a cache server that will service the client's HTTP requests. The cache stores HTTP information and tries to service the client's requests with the information it has stored. For most networks, web services are the primary consumer of network bandwidth. Web caching reduces network traffic and aides in optimizing bandwidth usage by localizing web traffic patterns, allowing content requests to be fulfilled locally. Web caching allows end-users to access web objects stored on local cache servers with a much faster response time than accessing the same objects over an internet connection or through a default gateway. This can also result in substantial cost savings by reducing the internet bandwidth usage.

Transparent Web Cache Balancing (TWCB) provides a means of load balancing HTTP requests over a server farm (a group of servers) or web caches.

TWCB adds three important elements to standard web caching: transparency, load balancing, and scalability:

- In standard web caching, network users must set their browsers to cache web traffic. Because web caching is highly sensitive to user preference, users sometimes balk at this requirement, and the inability to control user behavior can be a problem for the network administrator. TWCB is said to be transparent to the user because web traffic is automatically rerouted, and the ability to configure caching is removed from the user and resides instead in the hands of the network administrator. With TWCB the user can not by-pass web caching once set up by the network administrator. On the other hand, the network administrator can add users for whom web caching is not desired to a host redirection list, denying these users access to TWCB functionality.

- In standard web caching, a user-cache is configured and assigned to a single cache server. TWCB provides for load balancing across all cache servers of a given server farm that can be configured for heavy web-users using a predictor round-robin algorithm.
- Scalability is provided by the ability to associate multiple cache servers with the web cache. This scalability is further refined by the ability to logically associate cache servers with multiple server farms.

Implementing TWCB

Implementing TWCB requires a routed network with IP interfaces that allow the S-Series router to send requests for the internet to the correct web caching device.

There are five aspects to TWCB configuration:

- Create the server farms that will cache the web objects and populate them with cache servers.
- Optionally associate heavy web-users with a round-robin list which caches those users' web objects across all servers associated with the configured server farm.
- Optionally specify the hosts whose HTTP requests will or will not be redirected to the cache servers.
- Create a TWCB web cache that the server farms will be associated with.
- Optionally configure TWCB source and destination NAT providing a public facing IP address for clients owned by the TWCB router.
- Apply the TWCB web cache to an outbound interface (which can include VLAN, L3 tunnel, and L2 tunnel) to redirect HTTP traffic on that interface to the cache servers.

TWCB Overview

A TWCB configuration is made up of one or more cache servers that are logically grouped in a server farm and one or more server farms that are associated with a web cache.

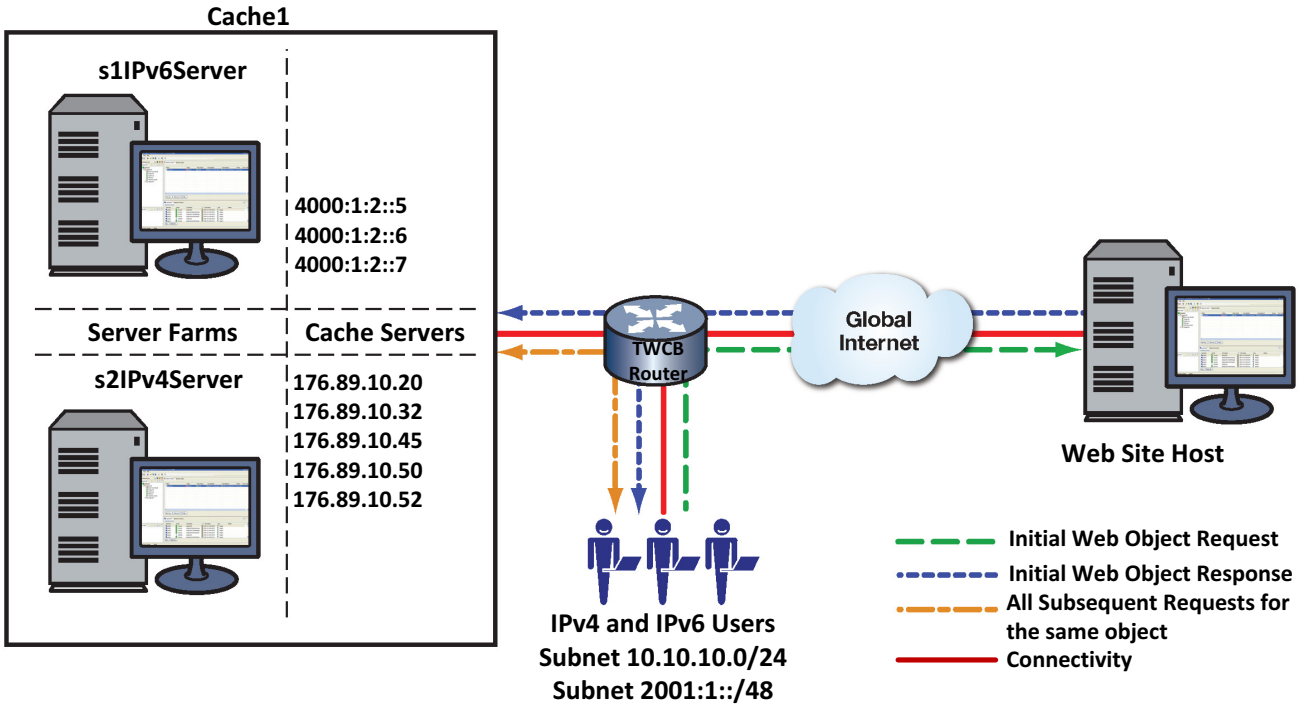
Figure 47-1 provides an overview of a TWCB configuration. In our overview, Cache1 is the name of the web cache. It is made up of two server farms: s1IPv6Server and s2IPv4Server. The s1IPv6Server server farm is configured with three IPv6 cache servers from the 4000:1:2:: subnet. The s2IPv4Server server farm is configured with five IPv4 cache servers from the 176.89.0.0 subnet. IPv6 end-user web objects are cached on the s1IPv6Server. IPv4 end-user web objects are cached on the s2IPv4Server.

The S-Series router does not act as a cache for web objects; rather, it redirects HTTP requests to local servers on which web objects are cached. The cache servers should have a web-based transparent proxy cache running. The Squid application is an example of a web-based transparent proxy cache.

In our example an IPv6 user on the 2001:1::/48 subnet or an IPv4 user on the 10.10.10.0/24 subnet initiates a web request, which it sends to the router. The router determines that the destination address is accessible through a VLAN that has a TWCB web cache applied to it. TWCB determines that the request is eligible for redirection, selects a cache server from the appropriate server farm, and sends the request to that cache server. The cache server will either service the request from it's cache or go out to the Internet (using its own source IP address) and retrieve the needed information. The cache server will respond to the client using the web site's IP address as the source IP address. From the client's perspective it is communicating with the actual web site, when in fact it is really conversing with a local transparent cache.

Once a web object resides in the cache, any future requests for that web object will be handled by the cache server until the cache entry expires. Cache entry expiration is configured in the web-based transparent proxy cache application installed on the cache server.

Figure 47-1 TWCB Configuration Overview



S-Series devices support both standard and source and destination NAT TWCB. Standard TWCB assumes that both the webcache and user clients are either directly attached to the TWCB router or that the cache server response to the user client will transit the TWCB router. Most of the discussion in this section assumes a standard TWCB configuration context. TWCB source and destination NAT allows user clients, the TWCB router, and the webcache to be located anywhere in the network. See "TWCB Source and Destination NAT" on page 47-8 for a TWCB source and destination NAT discussion.

There are five components in a TWCB configuration:

- The server farm
- The cache server
- The web cache
- The outbound interface
- The switch and router

The Server Farm

The server farm consists of a logical grouping of cache servers. Each server farm belongs to a web cache. TWCB supports the configuration of up to five server farms that can be associated with the web cache.

There are three aspects to configuring a server farm:

- Creating the server farm

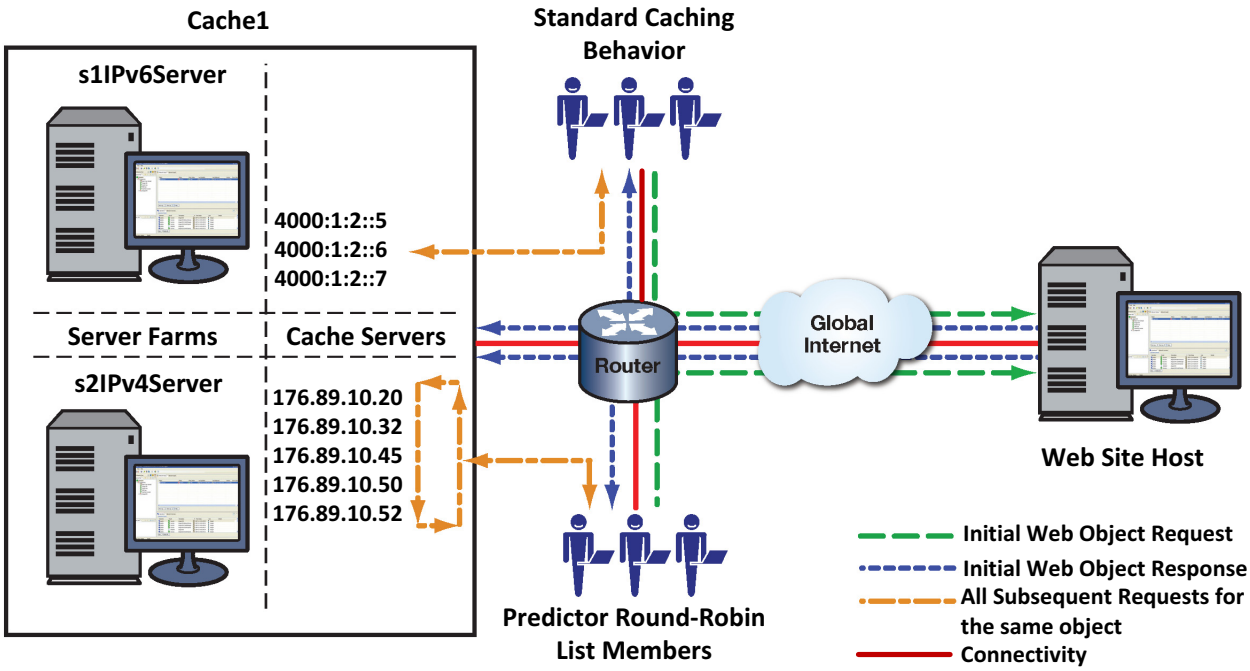
- Associating one or more cache servers with the server farm
- Optionally configuring some users to be members of a round-robin list on that server farm.

You create a server farm by naming it. Upon naming a server farm, you are placed in web cache server farm configuration mode. The cache server is the IP address of the actual transparent proxy web cache server.

The default behavior for selecting a cache from a server farm is to use a hash of the destination IP addresses. Should a single cache server be associated with one or more heavy traffic destination IP addresses, then the round robin selection mechanism can be used to balance traffic to particular ranges of destination IP addresses among the caches configured to the server farm.

In Figure 47-2 we see how requests destined for one particular destination IP, configured for standard caching, only accesses cached web objects from the cache server where its cache resides. In this case, the destination IP addresses reside on the s1IPv6Server server farm 4000:1:2::6 cache server. The s2IPv4Server server farm is configured with a predictor round-robin list. Each list member has its web objects cached across all the cache servers on the s2IPv4Server server farm.

Figure 47-2 Predictor Round-Robin Overview



The predictor round-robin feature allows for the creation of up to 10 user lists. Members of a predictor round-robin list no longer have a single cache on a single cache server. Instead, web objects for list members are cached across all cache servers associated with this server farm in a round robin fashion. A server farm with a configured predictor of round-robin will only cache members of predictor round-robin lists associated with that server farm.

In an IPv6 TWCB server farm round robin context, you must use an IPv6 access list to define the cache servers used by the round robin. In an IPv4 TWCB server farm round robin context, you can either use an IPv4 access list or specify a beginning and end IP address for a range of cache servers.

The Cache Server

The cache server is the IP address of the actual transparent proxy cache server. Each cache server belongs to a server farm. You create a cache server by entering its IP address within the server farm configuration command mode. Once entered, you are placed in TWCB cache server configuration command mode.

Within TWCB cache server configuration command mode, you can select the type of fail detection that will be used by this cache server and set its parameters. Fail detection specifies the method that will be used by the router to determine whether the cache server is in an up or down state. Fail detection type can be set to ping, application TCP, or both. The application method defaults to a check of service availability on port 80. A non-standard HTTP port can be configured. The application method will use this configuration when checking service availability. Both the interval between retries and the number of retries for each method are configurable.

You can configure the maximum number of connections (bindings) allowed for this cache server.

Once a cache server is configured, you must place it in service and the cache server should be reporting that the server is "up" for the cache server to be active on the server farm.

Cache Server Weight

Weighted round robin is a round robin algorithm that takes into account a weight assigned to each cache server. Weight is a way of accounting for the resource differences between servers. If a server has the capacity to handle twice the number of sessions as another server, its weight ratio to the other server can be set to 2:1. The default weight for all cache servers is **1**. When all cache servers are configured with the default weight, each cache server is treated equally. When a non-default weight is applied to any cache servers in the web cache server farm, the algorithm takes that weight into account when assigning sessions to the cache servers.

Consider the following example. A server farm contains three cache servers with the following weights: server A has a weight of **1**, server B has a weight of **2**, and server C has a weight of **3**. For each six (the sum of the three weights) active sessions, server A will be assigned 1 session, server B will be assigned 2 sessions, and server C will be assigned 3 sessions in a round robin fashion. For this example, the weight ratio between the three servers would be 1:2:3.

Fail Detection

It is important for TWCB to know whether a cache server can provide the requested service. There are three fail detection methods for determining the state of a cache server, server port, and application content:

- **Ping** - The real server is pinged.
- **TCP Port Service Verification** - The application service port is verified.
- **Application Content Verification (ACV)** - The content of an application is verified.

Fail detection methods are configured within probes using the tracked object manager facility. Probe creation and configuration is detailed, along with fail detection method details in Chapter 13, **Tracked Object Manager Configuration**.

ICMP ping probe monitoring of a cache server occurs by default, using the predefined ICMP probe **$twcb_default**. See "Preset Default ICMP Probes" on page 13-7 for preset default ICMP probe details.

TWCB supports the assigning of up to two probes per server: an ICMP ping and a TCP or UDP probe that can be configured for port verification and optionally for ACV. Probes are assigned to a cache server configuration using the **faildetect probe** command in cache server configuration mode. When assigning a probe to a cache server, specify probe **one** or **two**, and the name of the probe. The **$twcb_default** default ICMP ping probe is auto-assigned to probe **one**.

The probe type setting allows you to set whether configured probes are active or inactive for a server context. The probe type setting does not change the probe configuration. When probe type is set to **probe**, the probe configuration for the server context is active; probes are sent to the server in accordance with the configured settings. When probe type is set to **none**, the probe configuration is inactive; no probes are sent for the server context. The default probe type is **probe**. Use the **probe type** command in real server configuration mode to set the probe type for the server context.

In a server configuration context, probe configuration can be reset to factory default values by resetting fail detection for that server context. Resetting fail detection in a server configuration context:

- Sets the probe type to the default value of **probe**
- Sets the probe for probe **one** to the **$twcb_default** default probe for the server context
- Removes any configured probe configuration for probe **two**

TWCB fail detection sets the application port to **80** by default. Use the **faildetect app-port** command in cache server configuration mode to set the TCP port on the cache server to a value other than 80 if required.

Any preexisting probe is overwritten when assigning a probe.

This example shows how to:

- Create a TCP probe named **TCP-HTTP**
- Configure the ACV request and reply strings
- Place the probe inservice
- Display a detailed level of configuration information for the probe
- Assign the probe to probe **one** of the **186.89.10.51** cache server on the TWCB server farm **s1Server**:
- Assign port **8080** as the TCP port to be monitored.
- Enable the real server configuration

```
S Chassis(su)->configure
S Chassis(su-config)->probe TCP-HTTP tcp
S Chassis(su-config-probe)->inservice
S Chassis(su-config-probe)->acv request "GET / HTTP/1.1\\r\\nHost:
2.0.0.5\\r\\n\\r\\n"
S Chassis(su-config-probe)->acv reply "HTTP/1.1 200 OK\\r\\n"
S Chassis(su-config-probe)->show probe TCP-HTTP detail
Probe:                        TCP-HTTP  Type:                              tcp-acv
Administrative state:        inservice  Session count:                         1
Fail-detect count:                   3  Pass-detect count:                    3
Fail-detect interval:                5  Pass-detect interval:                 5
3-way TCP handshake wait time:       5  Server response wait time:           10
Application Content Verification:
 Request-string: GET / HTTP/1.1\\r\\nHost: 2.0.0.5\\r\\n\\r\\n
 Reply-string:   HTTP/1.1 200 OK\\r\\n
 Close-string:
 Search-Depth:   255
```

```
S Chassis(su-config-probe)->exit
S Chassis(su-config)->ip twcb wcserverfarm s1Server
S Chassis(config-twcb-wcsfarm)->cache 186.89.10.51
S Chassis(config-twcb-cache)->faildetect probe one TCP-HTTP
S Chassis(config-twcb-cache)->faildetect app-port 8080
S Chassis(config-twcb-cache)->inservice
S Chassis(config-twcb-cache)->
```

The Web Cache

The web cache is a logical entity in which server farms are added and rules are configured that govern what TCP data flows should be redirected. Multiple web caches can be configured on a device. Use the **show router limit** command to determine the number of web caches supported on the device. A web cache supports a single protocol port such as port 80, 443 or 8080. A web cache can be configured per protocol port for each VRF segment configured on the device.

You create a web cache by naming it in router configuration command mode. Once entered, you are placed in TWCB web cache configuration command mode. Once in TWCB web cache configuration command mode, you can:

- Add up to 10 server farms to a web cache.
- Optionally specify a non-standard port for the redirection of HTTP requests. Outbound HTTP requests are directed to port 80 by default.
- Create bypass lists containing a range of host web sites for which HTTP requests are not redirected to the cache servers for this web cache.
- Specify the clients (source IP addresses) whose HTTP requests are or are not redirected to the cache server. Clients permitted redirection take part in TWCB. Clients denied redirection do not take part in TWCB. All clients are permitted redirection by default.
- Configure TWCB source and destination NAT allowing the TWCB router, user clients, and webcache to be located anywhere in the network.

The Outbound Interface

The outbound interface is typically an interface that connects to the internet. If a TWCB web cache is configured to an interface, all TCP packets routed out that interface that match the configuration of the web cache will be considered by TWCB for redirection. Within the interface configuration command mode, you can configure this interface to redirect outbound HTTP traffic to the web cache. Multiple web caches can be specified on a single interface. TWCB is supported on VLAN, L3 tunnel interfaces, and on L2 tunnel, including tunnel bridge port.

The Switch and Router

The TWCB Binding

A TWCB binding has three devices associated with it: a client that initiates a service request, the destination device that responds to the service request, and a cache server that caches the response data. Each binding is based upon the following criteria:

- Source IP Address - The client IP address
- Destination IP Address - The IP address of the destination device
- Destination Port - The Destination Device Port

- Cache Server IP Address - The IP address of the cache server

TWCB matches bindings based upon the following four tuples: TCP protocol, source IP address, destination IP address, and destination web cache HTTP port value. Use the **show ip twcb bindings** or **show ipv6 twcb bindings** command to display active TWCB bindings for this device.

TWCB Source and Destination NAT

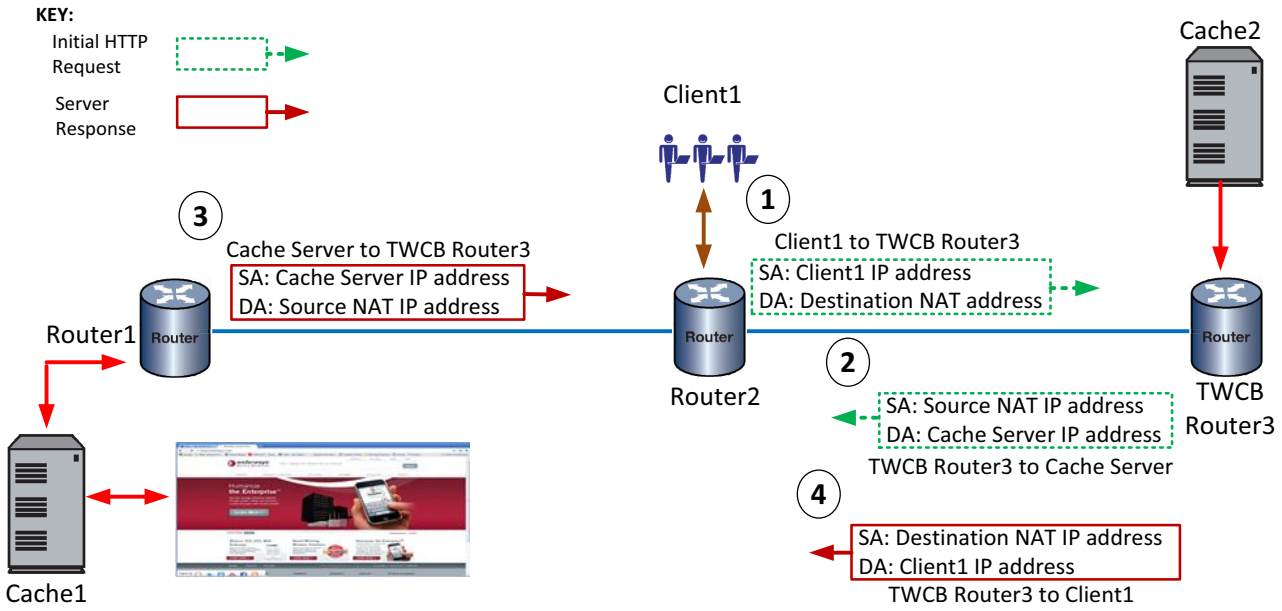
Standard TWCB operation requires that a cache server have a route back to the client through the TWCB router. A second consideration is that client addresses are often unknown to the cache server. The TWCB source and destination NAT feature addresses these two issues. The configuration of TWCB source and destination NAT allows the client, TWCB router, and cache server to reside anywhere in the network and still provide for the forwarding of an HTTP request from the client to the web cache server. TWCB source and destination NAT also provides for the reverse forwarding from the web cache server to the client, assuring that the packet flow will pass through the TWCB router.

Figure 47-1 on page 47-3 and Figure 47-2 on page 47-4 illustrate examples of standard TWCB configurations. In both cases, clients and cache servers are directly connected to the TWCB router, assuring that the cache server both knows the address of the requesting client and has a route back to that client through the TWCB router.

Figure 47-3 on page 47-9 illustrates a typical TWCB source and destination NAT configuration. Any requests directed to Cache1, can not assure that the reverse path will transit the TWCB router. Only Cache2, being directly connected to the TWCB router, can assure that the reverse path will transit the TWCB router. Therefore, this TWCB configuration requires source and destination NAT to account for any requests going to Cach1. The administrator configures TWCB Router3 for source and destination NAT by specifying both a Destination NAT address range and a Source NAT address range.

1. Client1, directly attached to Router2, makes an HTTP request to the TWCB Router3 for www.extremenetworks.com. The packet flow source address is the Client1 IP address. The packet flow destination address is the TWCB destination NAT address configured on TWCB Router 3. When the packet reaches Router3, the TWCB router can direct the request to either web cache in the configuration. TWCB selects Cache1 for this request.
2. Before it forwards the request to Cache1, TWCB selects a source NAT address for the HTTP request packet flow from the configured range and sets the destination address to a cache server on Cache1 and forwards the request to the cache server.
3. The Cache1 cache server retrieves the request and reverse forwards it back to the TWCB router using its own address as the source and the source NAT address as the destination.
4. When the packet flow arrives at the TWCB router, it forwards it on to Client1 using the destination NAT address as the packet flow source and Client1's address as the destination.

Figure 47-3 TWCB Source and Destination NAT Overview



TWCB Destination NAT

TWCB Destination NAT IP addressing provides a web cache public facing address. This address is owned by the TWCB router. The client making an HTTP request uses the TWCB destination NAT address to reach the cache server from anywhere in the network. The public web cache addresses are defined in a standard access list that is assigned to a web cache configuration using the **destination ip** command. TWCB forwards the HTTP request to the appropriate cache server for processing.

TWCB Source NAT

Before the TWCB router forwards the HTTP request to the web cache server, it first selects a source NAT address from the IPv4 source NAT pool or IPv6 source NAT address range defined using the **source nat pool** command. Using this public facing source NAT address assures that the web cache server reverse packet flow will pass through the TWCB router on its way back to the client.

For IPv4, one or more overloaded public facing IP addresses are assigned to a NAT pool, allowing multiple clients to use the same external address, with NAPT assigning an unused port to differentiate between clients. For IPv6 clients, an IPv6 address and prefix length is specified providing a range of external IP addresses.

The IPv6 address definition requires a prefix length of 111 or less in order to account for the checksum-neutral calculation of the IPv6 client address.

Configuring TWCB

This section provides details for the configuration of TWCB on the S-Series products.

For information about...	Refer to page...
Configuring the Server Farm	47-10
Configuring the Cache Server	47-11

Table 47-1 lists TWCB parameters and their default values.

Table 47-1 Default TWCB Parameters

Parameter	Description	Default Value
faildetect	Specifies whether the ping, application, or both ping and application detection method will be used to determine TWCB cache server up or down status.	both
idle-timeout	Specifies the number of seconds an IPv4 or IPv6 binding remains idle before being deleted.	240 seconds.
maxconns	Specifies the maximum number of bindings allowed for this server.	0 (no limit)
weight	Specifies a cache weight value to IPv4 or IPv6 cache servers in a web cache server farm.	1

Configuring the Server Farm

Procedure 47-1 describes how to configure a TWCB server farm.

Procedure 47-1 TWCB Server Farm Configuration

Step	Task	Command(s)
1.	Create the server farm.	**ip twcb wcserverfarm** *serverfarm-name* **ipv6 twcb wcserverfarm** *serverfarm-name*
2.	Associate a cache server with the server farm.	**cache** *ip-address*
3.	Optionally, configure a predictor round-robin list.	For IPv4: **predictor** {**dest-ip-hash** | **roundrobin** {*ipv4-address-begin ipv4-address-end* | *acl-list*}} For IPv6: **predictor** {**dest-ip-hash** | **roundrobin** *acl-list*}
4.	Optionally, configure a cache server round-robin weight.	**weight** *weight*
5.	Optionally, configure a description for this server farm.	**description** *description*

Configuring the Cache Server

Procedure 47-2 describes how to configure a TWCB cache server.

Procedure 47-2 TWCB Cache Server Configuration

Step	Task	Command(s)
1.	Create the cache server.	**cache** *ip-address*
2.	In cache server configuration command mode, optionally apply a configured probe to probe **one** or probe **two** to monitor this real server. An ICMP ping and TCP or UDP probe can be configured on separate command lines.	**faildetect probe** {**one** | **two**} *probe-name*
3.	In cache server configuration command mode, optionally specify whether the currently configured probes are active or inactive for this cache server.	**faildetect type** {**none** | **probe**}
4.	In cache server configuration command mode, optionally reset failure detection configuration to the factory default settings for this real server.	**faildetect reset**
5.	In cache server configuration command mode, optionally change the port number the assigned probe will monitor for this TWCB cache server context ,	**faildetect app-port** *port-number*
6.	In cache server configuration command mode, optionally apply a cache weight value to IPv4 or IPv6 cache servers in a web cache server farm	**weight** *weight*
7.	In cache server configuration command mode, optionally change the maximum number of bindings allowed for this cache server.	**maxconns** *number*
8.	Optionally, configure a description for this cache server.	**description** *description*
9.	In cache server configuration command mode, place the cache server in service.	**inservice**

Configuring the Web Cache

Procedure 47-3 describes how to configure a TWCB web cache.

Procedure 47-3 TWCB Web Cache Configuration

Step	Task	Command(s)
1.	Create a web cache using the specified name.	**ip twcb webcache** *web cache-name* **ipv6 twcb webcache** *web cache-name*
2.	Optionally specify the number of seconds a binding remains idle before being deleted for this web cache.	**idle timeout** *seconds*
3.	Add the specified server farm to this web cache.	**serverfarm** *serverfarm-name*
4.	Place this web cache server farm in service.	**inservice**

Procedure 47-3 TWCB Web Cache Configuration (continued)

Step	Task	Command(s)
5.	Optionally redirect outbound HTTP requests to a non-standard HTTP port number.	**http-port** *port-number*
6.	Optionally specify web host sites for which HTTP requests are not redirected to the cache servers.	For IPv4: **bypass-list** {**range** *begin-ip-address end-ip-address* | **aclName** *access-list*} For IPv6: **bypass-list aclName** *access-list*
7.	Optionally permit or deny redirection of HTTP requests for the list of clients to this web cache.	**host** {**permit** | **deny** | **aclName** *access-list*} **redirect** {**range** *begin-ip-address end-ip-address*}
8.	Optionally configure destination NAT addresses for this web cache.	**destination ip** *access-list*
9.	Optionally configure source NAT addresses for this web cache. The IPv4 NAT pool specifies an IPv4 overloaded address.	**source nat pool** {*ipv4-nat-pool* | *ipv6-address/prefix-len*}
10.	Optionally, configure a description for this web cache.	**description** *description*
11.	Place this web cache in service.	**inservice**

Configuring the Outbound Interface

Configuring an HTTP outbound interface consists of setting the redirection of outbound HTTP traffic from this interface to the cache servers. The outbound interface can be a VLAN, L3 tunnel, or L2 tunnel. The syntax for this command in each interface configuration context is:

ip twcb *webcache-name* **redirect out**

ipv6 twcb *webcache-name* **redirect out**

Displaying TWCB Statistics/Information

Table 47-2 describes how to display TWCB statistics/information.

Table 47-2 Displaying TWCB Statistics

Task	Command(s)
Display server farm configuration data.	**show ip twcb wcserverfarms** [*serverfarm-name* | **detail**] **show ipv6 twcb wcserverfarms** [*serverfarm-name* | **detail**]
Display web cache configuration data.	**show ip twcb webcaches** [*webcache-name* | **detail**] **show ipv6 twcb webcaches** [*webcache-name* | **detail**]
Display TWCB bindings.	**show ip twcb bindings** {**summary** | **id** *id* | **match** {*sip* | **} {*dip* | **} [**detail**]} **show ipv6 twcb bindings** {**summary** | **id** *id* | **match** {*sip* | **} {*dip* | **} [**detail**]}

Table 47-2 Displaying TWCB Statistics (continued)

Task	Command(s)
Display TWCB caches.	**show ip twcb caches** [*serverfarm-name*] [**detail**] **show ipv6 twcb caches** [*serverfarm-name*] [**detail**]
Display TWCB configuration information. TWCB information displays as a combined IPv4 and IPv6 value regardless of the command entered.	**show ip twcb info** **show ipv6 twcb info**
Display cache server statistical data. TWCB statistics display as a combined IPv4 and IPv6 counter value regardless of the command entered.	**show ip twcb statistics** [**-all_vrfs**] [**-interesting**] **show ipv6 twcb statistics** [**-all_vrfs**] [**-interesting**]

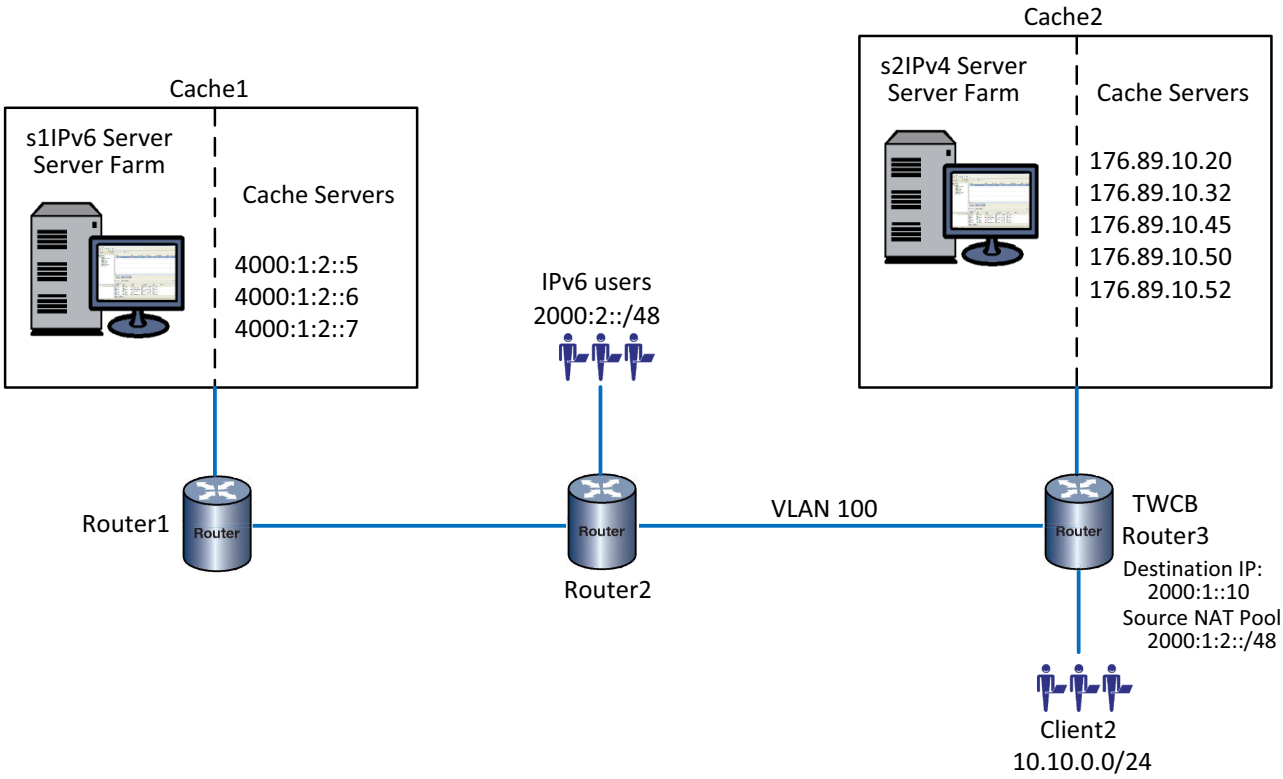
TWCB Configuration Example

In this TWCB configuration example we will step through the configuration of two TWCB webcaches:

- An IPv6 webcache named **Cache1** configured with the **s1IPv6Server** server farm for TWCB source and destination NAT
- An IPv4 webcache named **Cache2**, configured with the **s2IPv4Server** server farm for standard TWCB

See Figure 47-4 for a presentation of the example setup.

Figure 47-4 TWCB Configuration Example Overview



The IPv6 Webcache and Server Farm

The IPv6 user clients and the webcache they access for HTTP requests are not directly attached to the TWCB router and can not assure a path from the cache servers back to the user clients that transit the TWCB router. Therefore, the IPv6 **Cache1** webcache requires that source and destination NAT be configured.

The destination IP address for Cache1 is **2000:1::10** and is contained in the IPv6 standard **dest1_acl** access list. The source NAT pool address range is **2000:1:2::/48**.

Cache1 uses the default HTTP port: **80**. A bypass list denies TWCB functionality for web requests to web host sites **3000:1::/48** because these sites require IP address authentication for user access. Cache1 configuration denies TWCB functionality to end-users **2000:2::1:2:0/64** as specified by IPv6 access list **deny1_acl**.

The **s1IPv6Server** server farm is configured with cache servers **4000:1:2::5-7**. All cache servers are configured the same and our example shows the CLI input for cache server **4000:1:2::5**.

The predictor method for the **s1IPv6Server** is the default predictor destination IP hash.

s1IPv6Server cache servers will use an ICMP ping probe with parameter values changed to an interval of **4** seconds and the number of retries to **5**. The maximum number of connections per cache server will be configured for 800.

The IPv4 Webcache and Server Farm

The IPv4 user clients and the webcache they access for HTTP requests are directly attached to the TWCB router. Therefore, the IPv4 **Cache2** webcache is configured for standard TWCB.

Cache2 uses the non-default HTTP port **8080**. A bypass list denies TWCB functionality for web requests to web hosts **50.10.10.30** to **50.10.10.43** because these sites require IP address authentication for user access. **Cache2** configuration denies TWCB functionality to end-users **10.10.10.25** to **10.10.10.30**.

The **s2IPv4Server** server farm is configured with cache servers **176.89.10.20,32,45,50,52**. All cache servers are configured the same and our example shows the CLI input for cache server **176.89.10.20**.

The predictor method for the **s2IPv4Server** is **round-robin** because these hosts have an expectation of heavy web-site access requirements.

The **s2IPv4Server cache server**s will use the default **$twcb_default** ICMP probe and a TCP probe for port verification with parameter values changed to a faildetect interval of **12** seconds and the number of retries to **5**. The maximum number of connections per cache server will be configured for **800** for both server farms.

Configure the s1IPv6Server Server Farm

Configure the ICMP probe:

```
Router3(rw)->configure
Router3(su-config)->probe s1-ICMP icmp
Router3(su-config-probe)->faildetect count 5 interval 4
Router3(su-config-probe)->exit
```

Create the server farm:

```
Router3(rw-config)->ipv6 twcb wcserverfarm s1IPv6Server
Router3(rw-config-twcb-wcsfarm)->
```

Configure cache server 4000:1:2::5

```
Router3(rw-config-twcb-wcsfarm)->cache 4000:1:2::5
Router3(rw-config-twcb-cache)->faildetect probe one s1-ICMP
Router3(rw-config-twcb-cache)->maxconns 800
Router3(rw-config-twcb-cache)->inservice
Router3(rw-config-twcb-cache)->exit
Router3(rw-config-twcb-wcsfarm)->
```

Configure the s2IPv4Server Server Farm

Configure the TCP probe:

```
Router3(rw)->configure
Router3(su-config)->probe s2-TCP tcp
Router3(su-config-probe)->faildetect count 5 interval 12
Router3(su-config-probe)->exit
```

Configure server farm s2IPv4Server:

```
Router3(rw-config)->ip twcb wcserverfarm s2IPv4Server
```

```
Router3(rw-config-twcb-wcsfarm)->predictor roundrobin 176.89.10.20 176.89.10.52
Router3(rw-config-twcb-wcsfarm)->
```

Configure cache server 176.89.10.20:

```
Router3(rw-config-twcb-wcsfarm)->cache 176.89.10.20
Router3(rw-config-twcb-cache)->faildetect probe two s2-TCP
Router3(rw-config-twcb-cache)->maxconns 800
Router3(rw-config-twcb-cache)->inservice
Router3(rw-config-twcb-cache)->exit
Router3(rw-config-twcb-wcsfarm)->exit
Router3(rw-config)->
```

Configure the cache1 Web Cache

Configure the access lists used by Cache1:

```
Router3(rw-config)->ipv6 access-list standard dest1_acl
Router3(su-cfg-ipv6-std-acl)->permit 2000:1::10
Router3(su-cfg-ipv6-std-acl)->exit
Router3(rw-config)->ipv6 access-list standard bypass1_acl
Router3(su-cfg-ipv6-std-acl)->deny ipv6 3000:1::/48
Router3(su-cfg-ipv6-std-acl)->exit
Router3(rw-config)->ipv6 access-list standard deny1_acl
Router3(su-cfg-ipv6-std-acl)->deny ipv6 2000:2::1:2:0/64
Router3(su-cfg-ipv6-std-acl)->exit
Router3(rw-config)->
```

Configure the web cache cache1:

```
Router3(rw-config)->ip twcb webcache cache1
Router3(rw-config-twcb-webcache)->serverfarm s1IPv6Server
Router3(rw-config-twcb-webcache)->destination ip dest1_acl
Router3(rw-config-twcb-webcache)->source nat pool 2000:1:2::/48
Router3(rw-config-twcb-webcache)->bypass-list aclName bypass1_acl
Router3(rw-config-twcb-webcache)->host deny redirect aclName deny1_acl
Router3(rw-config-twcb-webcache)->exit
Router3(rw-config)->
```

Configure the outbound interface that connects with the internet:

```
Router3(rw-config)->interface vlan 100
Router3(rw-config-intf-vlan.0.100)->ipv6 twcb cache1 redirect out
Router3(rw-config-intf-vlan.0.100)->end
Router3(rw)->
```

Configure the cache2 Web Cache

Configure the web cache cache2:

```
Router3(rw-config)->ip twcb webcache cache2
Router3(rw-config-twcb-webcache)->http-port 8080
```

```
Router3(rw-config-twcb-webcache)->serverfarm s2IPv4Server
Router3(rw-config-twcb-webcache)->bypass-list range 50.10.10.30 50.10.10.43
Router3(rw-config-twcb-webcache)->hosts redirect deny redirect range 10.10.10.25
10.10.10.30
Router3(rw-config-twcb-webcache)->exit
Router3(rw-config)->
```

Configure the outbound interface that connects with the internet:

```
Router3(rw-config)->interface vlan 100
Router3(rw-config-intf-vlan.0.100)->ip twcb cache2 redirect out
Router3(rw-config-intf-vlan.0.100)->end
Router3(rw)->
```

This completes the TWCB configuration example.

48

Virtual Router Redundancy Protocol (VRRP) Configuration

This document describes the Virtual Router Redundancy Protocol (VRRP) feature and its configuration on Extreme Networks S-Series devices.

Using VRRP in Your Network

Virtual Router Redundancy Protocol (VRRP) is an election protocol capable of dynamically assigning responsibility for a virtual router to one of the VRRP routers on a LAN. A virtual router is an abstract object managed by VRRP that acts as a default router for hosts on a shared LAN. It consists of a Virtual Router Identifier (VRID) and a set of associated IP addresses across a common LAN that define virtual router members. A VRRP router is a router with the VRRP protocol running on it. A VRRP router may participate in and backup one or more virtual routers.

VRRP specifies an election protocol that dynamically assigns responsibility for a virtual router to one of the VRRP routers on a LAN. The elected VRRP router is called the master. The router master controls the IP addresses associated with a virtual router. The master forwards packets sent to these IP addresses. The VRRP election process provides dynamic fail over of forwarding responsibility to another VRRP router should the current master become unavailable. This allows any of the virtual router IP addresses on the LAN to be used as the default first hop router by end-hosts. In this way, VRRP provides a higher availability default path without requiring configuration of dynamic routing or router discovery protocols on every end-host.

Note: When configuring an IPv6 VRRP link local address, all link local addresses must match on all routers running the same VRRP instance in a LAN segment. Only one link local address on a VRRP instance will be active at any given time.

Statically configured default routes can represent a single point of failure that can result in a catastrophic event, isolating all end-hosts that are unable to detect any alternate available path. VRRP is designed to eliminate the single point of failure inherent in the static default routed environment.

A critical-IP address defines an interface that will prevent the master router from functioning properly if the interface were to fail. When a critical IP interface goes down, its operational priority can be set to decrement to a value lower than the priority set for the backup router. In this case, the backup router becomes the master.

Fabric route mode can be enabled on the VRRP router allowing a VRRP instance in the backup state to forward IPv4 and IPv6 packets destined for the VRRP MAC address

Implementing VRRP in Your Network

To implement VRRP:

- Create a virtual router instance
- Configure all VRRP IP addresses associated with this virtual router
- Optionally change the VRRP router priority for this virtual router
- Optionally change the advertise interval for this virtual router
- Optionally set the VRID state interface down transition to interface up delay
- Configure a critical-IP interface, the failure of which will decrement the operational priority of the router causing the backup router to take over as master.
- Optionally configure this virtual router for VRRP authentication (Only applies to VRID's that have been created as version 2 of the protocol)
- Optionally enable accept-mode for this virtual router allowing the master for this virtual router to accept IP packets for the configured associated IP address list
- Optionally enable fabric route mode on the VRRP router.
- Optionally change the master preemption setting for this VRRP router
- Verify configuration and statistics using the VRRP display command

VRRP Overview

This section provides an overview of VRRP configuration.

Basic VRRP Topology

Figure 48-1 A Basic VRRP Topology

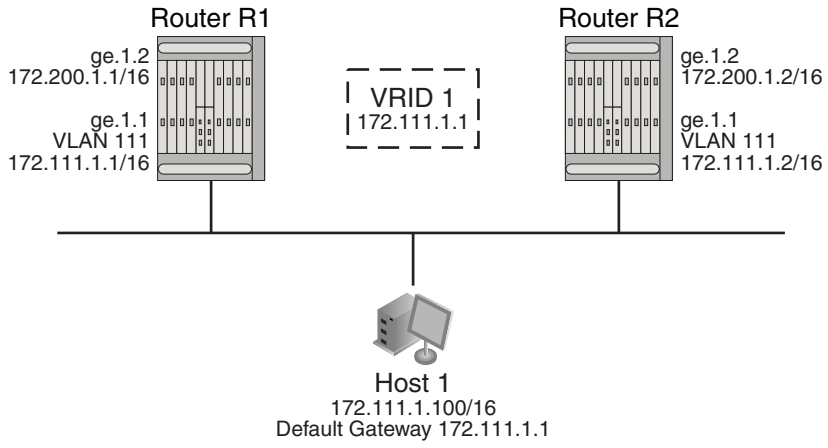


Figure 48-1 shows a basic VRRP topology with a single virtual router. Routers R1 and R2 are both configured with one virtual router (VRID 1). Router R1 serves as the master and Router R2 serves as the backup. The hosts are configured to use 172.111.1.1/16 as the default route.

If Router R1 should become unavailable, Router R2 would take over virtual router VRID 1 and its associated IP addresses. Packets sent to 172.111.1.1/16 would go to Router R2. When Router R1 comes up again, it would take over as master, and Router R2 would revert to backup.

VRRP Virtual Router Creation

Each virtual router has its own instance. Create a VRRP virtual router instance using the **vrrp create** command in interface configuration command mode specifying the VRID for this instance. The virtual router instance must be created on a routing interface before any other VRRP settings can be configured.

VRRP Master Election

After the virtual router instance has been created, assign the IP addresses associated with this virtual router using the **vrrp address** command in interface configuration command mode, specifying the IP address and the VRID this address is to be associated with. A virtual router IP address can be either an address configured on the routing interface or an address that falls within the range of any networks configured on this routing interface.

If the virtual router IP address is the same as the routing interface (VLAN) address owned by a VRRP router, then the router owning the address becomes the master. The master sends an advertisement to all other VRRP routers declaring its status and assumes responsibility for forwarding packets associated with its VRID.

If the virtual router IP address is not owned by any of the VRRP routers, then the routers compare their priorities and the higher priority owner becomes the master. VRRP router priority is set using the **vrrp priority** command in interface configuration command mode. If priority values are the same, then the VRRP router with the highest IP address is selected master.

VRRP advertisements are sent by the master router to other routers participating in the VRRP master selection process, informing them of its configured values. Once the master is selected, then advertisements are sent every advertising interval to let other VRRP routers in this VRID know the router is still acting as master of the VRID. All routers with the same VRID should be configured with the same advertisement interval. Use the **vrrp advertise-interval**, in interface configuration command mode, to change the advertise-interval for this VRID.

Configuring a VRRP Critical-IP Address

A critical-IP address defines an interface that will prevent the master router from functioning properly if the interface were to fail. A critical-IP address is typically an internet facing interface, but can be any IP address that does not include the VRRP configured interface between hosts and a VRRP master or backup first-hop router. An IP address of an interface connecting a master router to a router configured for internet access would be considered a critical-IP address for VRRP routing. Critical-IP addresses can be both local or remote.

Use the **vrrp critical-ip** command in interface configuration command mode to configure an internet facing IP address as a VRRP critical-IP address, specifying the affected IP address, the associated VRID, and an optional decrement priority setting. A default ICMP probe is auto-configured to monitor remote critical-IP addresses. An administratively configured ICMP probe can be applied to override the default ICMP probe. See “Preset Default ICMP Probes” on page 13-7 for default ICMP probe details. Probes are configured in the tracked object manager. See Chapter 13, **Tracked Object Manager Configuration** for details.

If the critical-IP interface goes down with priority configured and enabled, the operational priority for the VRID to which this critical-IP address is associated is decremented by the value of the priority specified in this command. If the operational priority of the VRID falls below that of a backup router, the backup router becomes the master and the VRID assumes the priority value of the new master. Should the critical-IP interface come back up, the priority of the router associated with this critical-IP address is increased by the priority set for the critical-IP address. If preempt is enabled on the critical-IP address associated router, the router will once again become master and the VRID assumes the priority of the new master.

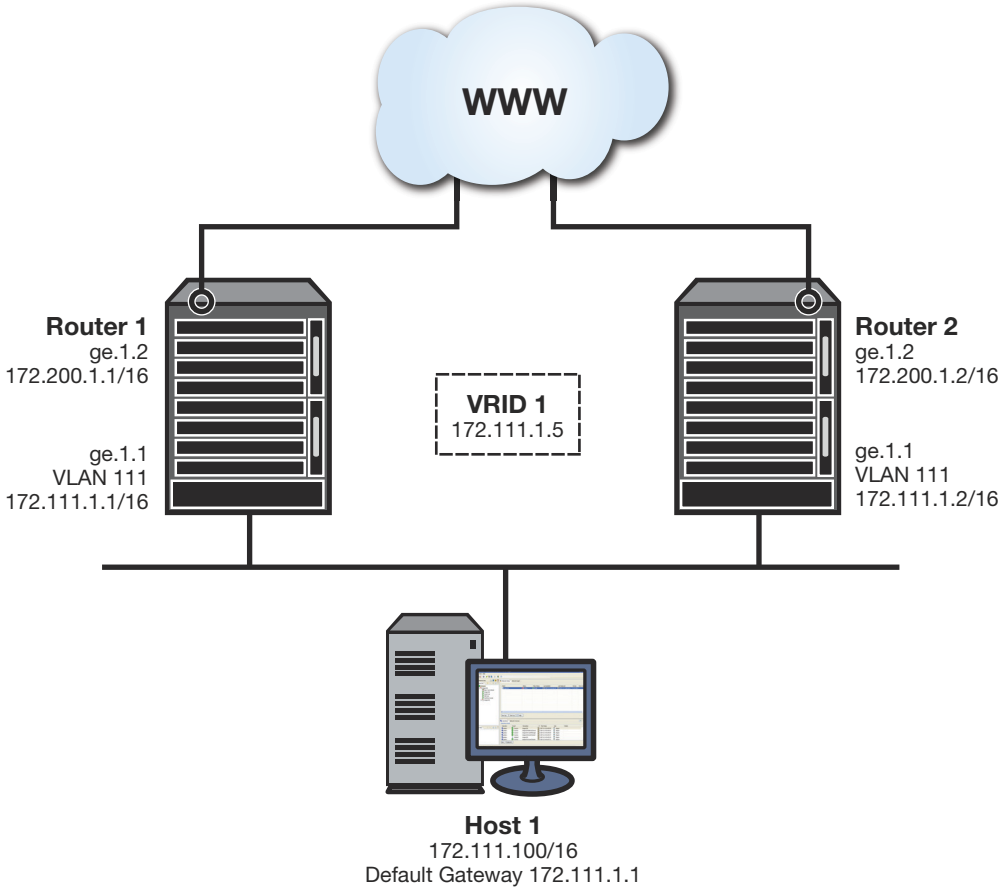
The default priority setting is enabled with a value of **10**. Setting the critical-IP address priority to enabled signals that the critical-IP will affect the operational priority for the VRID. Setting the priority to disabled signals the critical-IP interface state will have no effect on the operational priority for the VRID.

Up to 2048 critical-IP addresses can be configured on a device. Up to 10 per VRID.

If the critical-IP address is configured on a router where the VRID IP address is owned by that router, the critical-IP configuration is ignored. When a router owns the IP address configured for the VRID, that router is automatically made the master with a hard-coded priority of 255. Only the failure of the interface with the VRID IP address can cause the router to move to backup status.

Figure 48-2 presents a typical critical-IP address configuration.

Figure 48-2 Critical-IP Address Configuration



The VRRP configuration is entered as follows:

- On both router 1 and router 2, in VLAN 111 configuration command mode, VRID 1 is created using the **vrrp create** command.

- On both router 1 and router 2, in VLAN 111 configuration command mode, the IP address 172.111.1.5 is assigned to VRID 1 using the **vrrp address** command.
- On router 1, in VLAN 111 configuration command mode, the VRRP priority is set to 105 using the **vrrp priority** command.
- On router 2, in VLAN 111 configuration command mode, the VRRP priority is set to 100 using the the **vrrp priority** command.
- On router 1, in VLAN 200 configuration command mode, configure IP address 172.200.1.1/16 as critical-IP address, enabling a priority of 10, using the **vrrp critical-ip** command.

In this example, should the critical-IP address 172.200.1.1/16 go down, the VRID 1 priority would decrement by 10, the value of the down critical-IP address, to 95. Router 2, with a priority set to 100 would take over as master. Should the critical-IP address 172.200.1.1/16 come back up, the priority for router 1 would increment by 10 from 95 to 105. Router 1 would now have a priority higher than the current priority 100 for VRID 1 and would become master once again.

Configuring VRRP Authentication

A version 2 VRRP VRID can be configured for a simple clear text or encrypted MD5 authentication password. For MD5 authentication, 128-bit encryption is used unless the hmac-96 option is specified, in which case 96-bit encryption is used.

Use the **vrrp authentication** command in interface configuration command mode, specifying the authentication type, a password, and in the case of MD5 optionally specifying 96-bit encryption.

Enabling Master Preemption

By default, a router is enabled to preempt a lower priority master for the configured virtual router. If the router owns the virtual router IP address, it can not be preempted and always preempts other routers regardless of the priority setting or this preemption setting. Use the **vrrp preempt** command to enable or disable master preemption on this VRRP router.

It may be desirable to set a delay that a higher priority backup router should wait before preempting a lower priority master. By default there is no delay. To set a delay between 1 and 900 seconds, use the vrrp preempt-delay command in interface configuration command mode.

Enabling Fabric Route Mode on the VRRP Backup Router

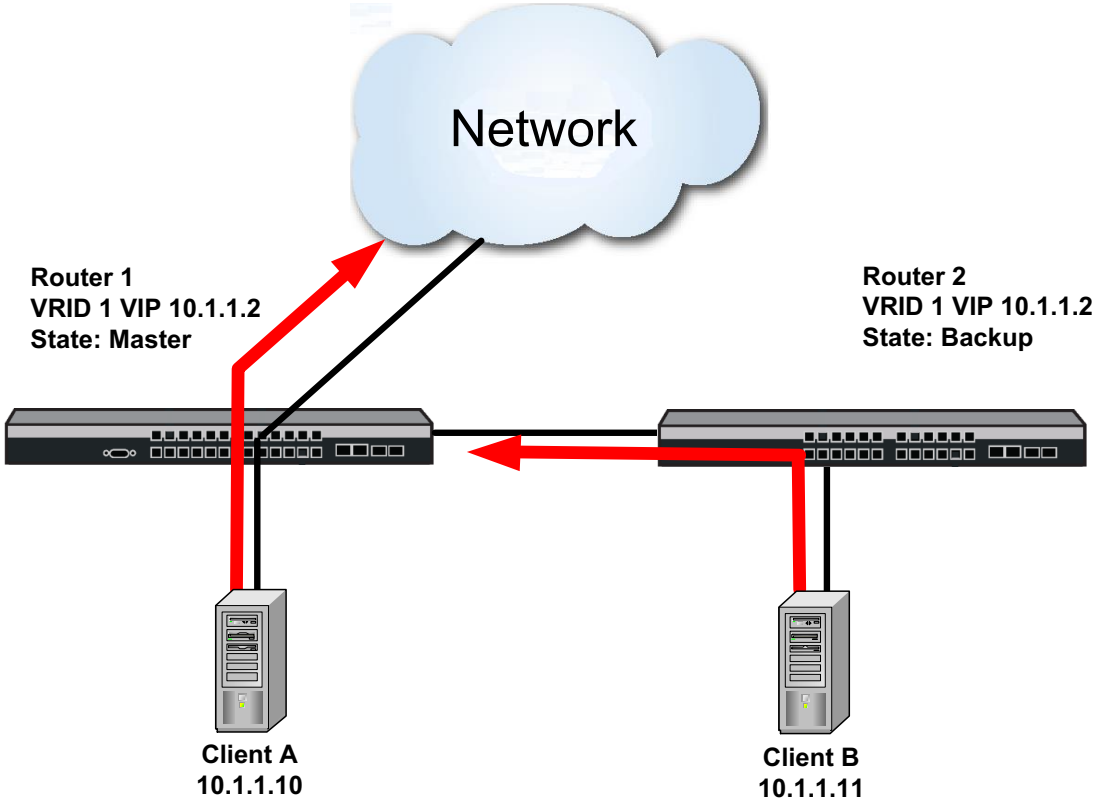
The fabric route mode allows a VRRP instance in the backup state to forward IPv4 and IPv6 packets destined for the VRRP MAC address. This feature provides for sharing of the traffic load across VRRP routers.

When fabric route mode is enabled on a VRRP router, the router behavior is exactly as detailed in RFC 5798 with these exceptions:

- **Accept-Mode** – A VRRP backup router with fabric route mode enabled will respond to ICMP packets it receives that are destined for the virtual gateway. A VRRP instance that wants to allow a non-owner virtual IP address to be pingable must enable the VRRP instance in fabric route mode to respond to ICMP packets using the **vrrp accept-mode** command.

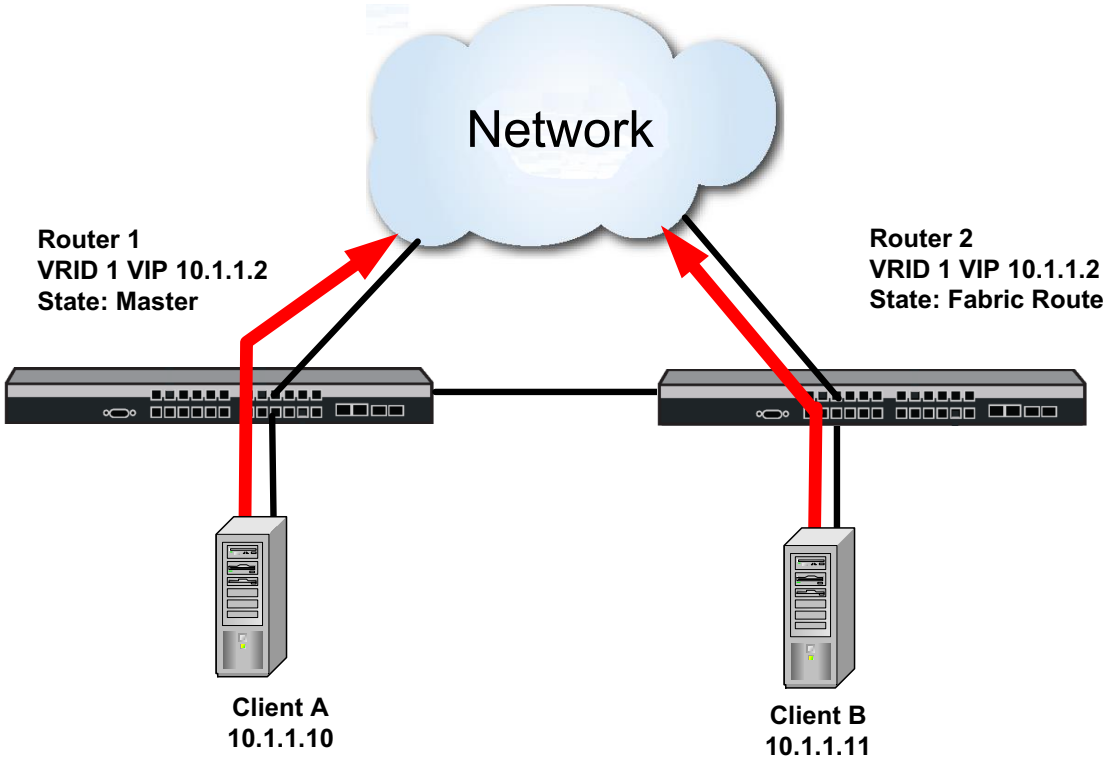
- **Forwarding** – The VRRP standard forwarding defines a single VRRP master router and one or more VRRP backup router. Packets arriving on the backup router, destined for the VRRP MAC address, are cross bridged to the VRRP master router. The VRRP master router forwards the packet to the network (see Figure 48-3 on page 6). The VRRP backup router remains in a dormant state and is not allowed to forward any traffic destined to the virtual gateway MAC address.

Figure 48-3 Standard VRRP Forwarding



With VRRP fabric route mode enabled on the backup router, a VRRP master and a backup router are started. Packets arriving on the backup router, destined to the VRRP MAC address, are no longer cross bridged. All IPv4 or IPv6 traffic destined for the VRRP MAC address and received directly on the VRRP backup router enabled for fabric route mode is now forwarded to the network by the backup router (See Figure 48-4 on page 7). All ACL, policy rules, router application services, and routing is applied to the incoming packet.

Figure 48-4 Fabric Route VRRP Forwarding



- **IPV4 ARP** – All broadcast ARP requests are still handled by the VRRP master router except for unicast ARP requests destined for the VRRP MAC address received by the VRRP backup router with fabric route mode enabled. The VRRP backup fabric route mode enabled router processes these requests.
- **IPV6 Neighbor Discovery** – All neighbor solicitation requests are handled by the VRRP master router, except for unicast neighbor solicitation requests destined for the VRRP MAC address received by the VRRP fabric route mode enabled backup router. The VRRP fabric route mode enabled backup router will process these requests.

Use the **vrrp fabric-route-mode** command, in interface configuration mode, specifying the VRID the fabric mode is applied to, to enable fabric route mode on the VRRP router. This command can be applied to both the VRRP master and backup routers, but will only be operational on the VRRP backup router.

Use the **show running-config** command, specifying the interface the VRRP router is configured on, to display determine whether VRRP fabric route mode has been configured on the router.

Enabling Fabric Route Host Mobility

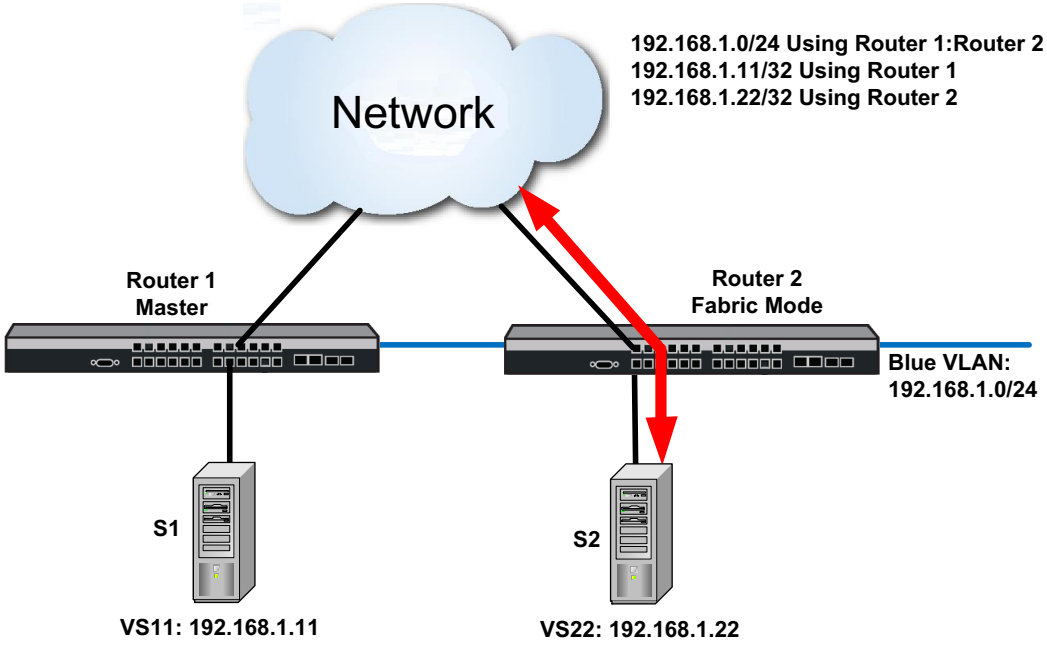
VRRP fabric route mode provides for sharing of traffic load for VRRP routers by allowing a VRRP instance in backup state to forward IPv4 and IPv6 traffic that is destined for the VRRP Gateway MAC address. Directly connected routes that should not be fully advertised can be optionally restricted using the host mobility ACL command. VRRP fabric route host mobility supports the alternate host route age out mechanism.

Should the directly attached device be moved to another VRRP router, upon timing out, the full address is advertised by the new VRRP router enabled for fabric route host mobility. By default, when a connected device is moved from one VRRP router to another, the original route continues to advertise until the age out timer expires. When a host mobility enabled device is moved to another router, initially both the old and new router continue to advertise their route for the

device. When the original VRRP router receives an advertisement of the moved route, it will ignore the remaining age out time and no longer advertise the associated route.

Figure 48-5 displays Router 2 in fabric mode allowing Router 2 to advertise virtual server 22 (VS22) on physical server S2 out to the network. Should VS22 be moved to physical server S1 with host mobility disabled, Router 2 will continue to advertise VS22 out to the network until the age out time expires. At the same time, Router 1 will begin advertising VS22.

Figure 48-5 Forwarding Prior to Virtual Server Move



This can cause asymmetric traffic flows to occur as shown in Figure 48-6 on page 9. With VRRP host mobility enabled, as soon as Router 2 receives Router 1's VS22 route advertisement, Router 2 ignores any remaining ageout time and stops advertising the VS22 route. All VS22 traffic flows use Router 1, as shown in Figure 48-7 on page 9.

Figure 48-6 Asymmetric Traffic Flows During Timeout Period

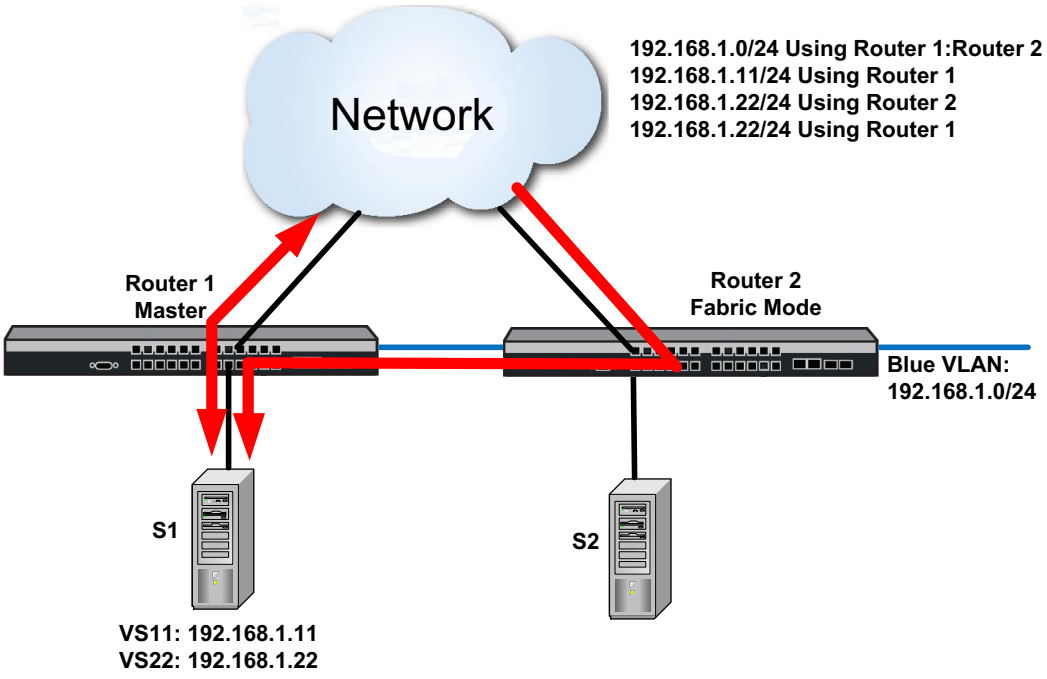
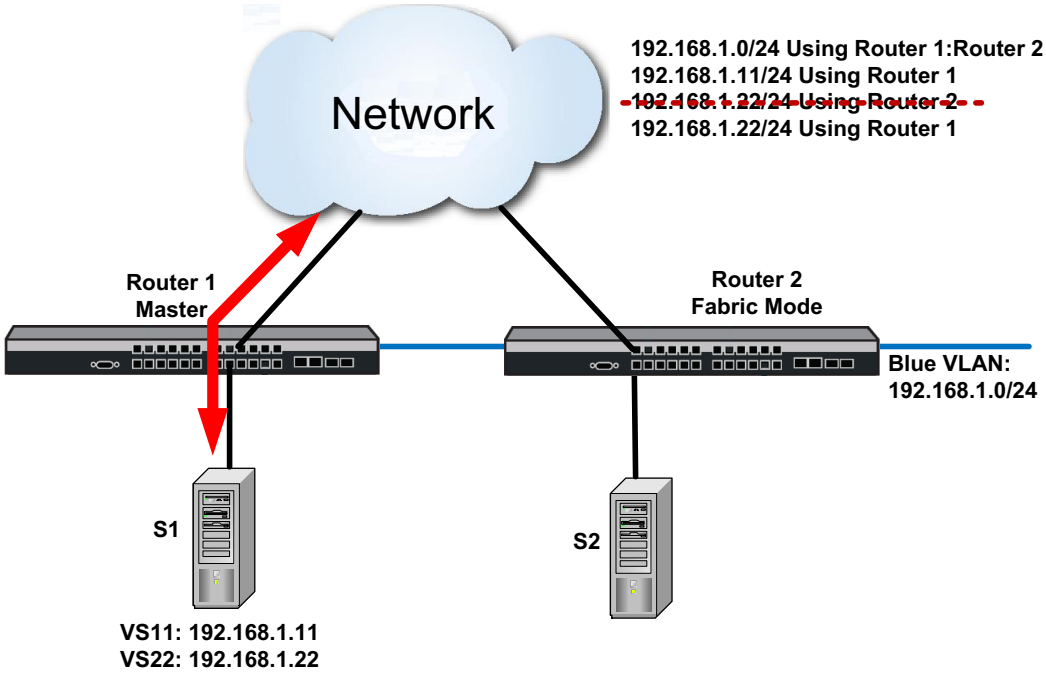


Figure 48-7 New Traffic Flows With Fabric Route Host Mobility Enabled



The actual age out time can be up to twice the time specified by the **set router host-mobility timeout** command.

Use the **vrrp host-mobility** command to enable fabric route host mobility on the VRRP router.

Use the **vrrp host-mobility-acl** command to specify an ACL containing those routes that should or should not take part in fabric route host mobility for this router.

Directly connected routes are not advertised until redistributed by OSPF using the OSPF **redistribute connected** command.

The following example configures Router 2 for fabric route host mobility and redistributes connected routes into OSPF. Fabric route and host mobility may also be configured on the VRRP master, but will have no affect while in master mode.

```
Router 2(rw)->configure
Router 2(rw-config)->interface vlan 200
Router 2(rw-config-intf-vlan.0.200)->ip address 198.168.1.0 255.255.255.0 primary
Router 2(rw-config-intf-vlan.0.200)->vrrp create 1 v3-IPv4
Router 2(rw-config-intf-vlan.0.200)->vrrp address 1 198.168.1.10
Router 2(rw-config-intf-vlan.0.200)->vrrp enable 1
Router 2(rw-config-intf-vlan.0.200)->vrrp fabric-route-mode 1
Router 2(rw-config-intf-vlan.0.200)->vrrp host-mobility 1
Router 2(rw-config-intf-vlan.0.200)->no shutdown
Router 2(rw-config-intf-vlan.0.200)->exit
Router 2(rw-config)->router ospf 1
Router 2(rw-config-ospf-1)->redistribute connected
```

Enabling the VRRP Virtual Router

All other VRRP options must be set before enabling a VRRP virtual router on the routing interface. Once enabled, you can not make any configuration changes to VRRP without first disabling the interface, using the **no vrrp enable** command.

Use the **vrrp enable** command in interface configuration command mode, specifying the VRID of the virtual router to be enabled.

Configuring VRRP

This section provides details for the configuration of VRRP on the S-Series products.

Table 48-1 lists VRRP parameters and their default values.

Table 48-1 Default VRRP Parameters

Parameter	Description	Default Value
accept-mode	Enables the master of this virtual router to accept IP packets for the configured IP address list, even if the device is not the owner.	disabled
advertise-interval	Specifies the interval between the advertisement the master sends to other routers participating in the selection process.	1 second
fabric route mode	A VRRP feature that allows a VRRP instance in the backup state to forward IPv4 and IPv6 packets destined for the VRRP MAC address.	disabled
interface-up delay	Specifies the delay in seconds for the VRID state transition from interface down to interface up	0 seconds (no delay)

Table 48-1 Default VRRP Parameters (continued)

Parameter	Description	Default Value
priority	Specifies the router priority for the master election for this virtual router.	100
VRRP preemption	Specifies whether higher priority backup VRRP routers can preempt a lower priority master VRRP router and become master.	enabled

Procedure 48-1 describes how to configure VRRP. All VRRP configuration commands are entered in interface configuration command mode.

Procedure 48-1 Configuring VRRP

Step	Task	Command(s)
1.	Create a virtual router instance. Supported VRRP Versions: • **v2-IPv4** - RFC 2338 • **v3-IPv4** - draft-ietf-vrrp-unified-spec-03 • **v3-IPv6** - draft-ietf-vrrp-unified-spec-03	**vrrp create** *vrid version*
2.	Configure all VRRP IP addresses associated with this virtual router.	**vrrp address** *vrid ip-address* [**enable** | **disable**]
3.	Configure a VRRP primary address for this virtual router.	**vrrp primary-address** *vrid ip-address* [**enable** | **disable**]
4.	Optionally, change the VRRP router priority for this virtual router.	**vrrp priority** *vrid priority*
5.	Optionally, change the advertise interval for this virtual router.	**vrrp advertise-interval** *vrid* {**seconds** *interval* | **centiseconds** *interval*}
6.	Optionally, set the VRID state interface down to interface up delay.	**vrrp interface-up-delay** *vrid seconds*
7.	Configure any critical-IP interfaces for this virtual router.	**vrrp critical-ip** *vrid ip-address* [*priority*] [**enable** | **disable**] [**remote** [**probe-name** *probe-name*]]
8.	Optionally, configure this router for VRRP authentication	**vrrp authentication** {**simple** *password* | **md5** *password* [**hmac-96**]
9.	Optionally, enable accept-mode for this virtual router, allowing the master to accept IP packets for the configured associated IP address list.	**vrrp accept-mode** *vrid*
10.	Optionally change the master preemption setting for this VRRP router.	**vrrp preempt** *vrid*
11.	Optionally, set the amount of time that will elapse before a backup VRRP router takes control from the current master when preemption is enabled.	**vrrp preempt-delay** *vrid delay*
12.	Optionally, in interface configuration mode, enable VRRP fabric route mode on the VRRP router.	**vrrp fabric-route-mode** *vrid*

Procedure 48-1 Configuring VRRP (continued)

Step	Task	Command(s)
13.	Optionally, configure an interface for VRRP fabric route host mobility.	**vrrp host-mobility** *vrid*
14.	Optionally, assign an ACL to the host mobility configuration to restrict routes that should or should not take part in this host mobility context.	**vrrp host-mobility-acl** *vrid acl-name*
15.	Optionally, in global configuration mode, set the age out timer for a directly connected host mobility route that is no longer available.	**host-mobility timeout**

Table 48-2 describes how to display VRRP information and statistics.

Table 48-2 Displaying VRRP Information and Statistics

Task	Command
Display VRRP configuration information for this system.	**show ip vrrp**
Display VRRP statistics for this system.	**show ip vrrp statistics**
Display VRRP configuration information for a specified interface.	**show ip vrrp** *interface* [*vrid*] [**verbose**]
Display detailed VRRP configuration information.	**show ip vrrp verbose**

VRRP Configuration Examples

This section presents a two VRRP configuration examples:

- A basic VRRP configuration example with a single virtual router configured
- A multiple backup VRRP configuration with three virtual routers configured

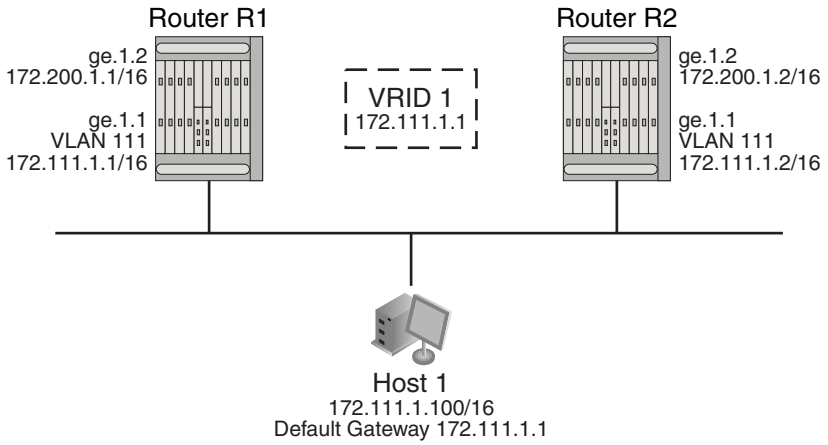
Basic VRRP Configuration Example

Figure 48-8 shows a basic VRRP configuration with a single virtual router. Routers R1 and R2 are both configured with one virtual router (VRID 1). Router R1 serves as the master because the VRRP router owns the IP address for this virtual router. Router R2 serves as the backup. The hosts are configured to use 172.111.1.1/16 as the default route.

The master advertise-interval is changed to 1.5 seconds for VRID 1.

If Router R1 should become unavailable, Router R2 would take over virtual router VRID 1 and its associated IP addresses. Packets sent to 172.111.1.1/16 would go to Router R2. When Router R1 comes up again, it would take over as master, and Router R2 would revert to backup.

Figure 48-8 Basic Configuration Example



Router R1 Configuration of VRRP Instance 1

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 111
S Chassis(rw-config-intf-vlan.0.111)->ip address 172.111.1.1 255.255.255.0
primary
S Chassis(rw-config-intf-vlan.0.111)->vrrp create 1 v3-IPv4
S Chassis(rw-config-intf-vlan.0.111)->vrrp advertise-interval 1 centiseconds 150
S Chassis(rw-config-intf-vlan.0.111)->vrrp address 1 172.111.1.1
S Chassis(rw-config-intf-vlan.0.111)->vrrp enable 1
S Chassis(rw-config-intf-vlan.0.111)->no shutdown
S Chassis(rw-config-intf-vlan.0.111)->exit
S Chassis(rw-config)->show ip vrrp verbose
Interface: vlan.0.111
 VRID: 1
  Version: 3, State: Master
  Master IP Address : 172.111.1.1
  Primary IP Address: 172.111.1.1
  Virtual MAC Address: 00:00:5E:00:01:01
  Advertisement Interval:   1.50 seconds
  Operational Priority:   255,  Configured Priority: 100
  Accept: no ,  Preempt: yes,  Preempt time:   0 seconds
  Virtual IP Count:   1,  Critical IP Count:    0
  Virtual IP Addresses:
   172.111.1.1
  Critical IP Addresses:
   Interface                                  Critical Priority     State
S Chassis(rw-config)->
```

Router R2 Configuration of VRRP Instance 1

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 111
```

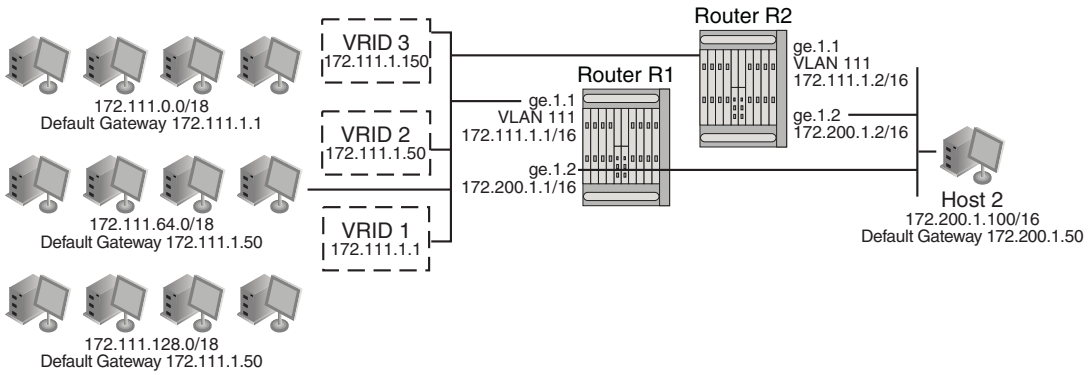
```
S Chassis(rw-config-intf-vlan.0.111)->ip address 172.111.1.2 255.255.255.0
primary
S Chassis(rw-config-intf-vlan.0.111)->vrrp create 1 v3-IPv4
S Chassis(rw-config-intf-vlan.0.111)->vrrp address 1 172.111.1.1
S Chassis(rw-config-intf-vlan.0.111)->vrrp advertise-interval 1 centiseconds 150
S Chassis(rw-config-intf-vlan.0.111)->vrrp enable 1
S Chassis(rw-config-intf-vlan.0.111)->no shutdown
S Chassis(rw-config-intf-vlan.0.111)->exit
S Chassis(rw-config)->show ip vrrp verbose
Interface: vlan.0.111
 VRID: 1
  Version: 3, State: Backup
  Master IP Address : 172.111.1.1
  Primary IP Address: 172.111.1.2
  Virtual MAC Address: 00:00:6A:00:03:01
  Advertisement Interval:   1.50 seconds
  Operational Priority:   100,  Configured Priority: 100
  Accept: no ,  Preempt: yes,  Preempt time:   0 seconds
  Virtual IP Count:   1,  Critical IP Count:    0
  Virtual IP Addresses:
   172.111.1.1
  Critical IP Addresses:
   Interface                                    Critical Priority     State
S Chassis(rw-config)->
```

In this configuration, if an interface on VLAN 111 for Router R1 fails, the interface on Router R2 will take over for forwarding outside the local LAN segment.

Multiple Backup VRRP Configuration Example

Figure 48-9 shows a multi-backup sample configuration.

Figure 48-9 Multi-Backup VRRP Configuration Example



Three VRRP instances are configured on VLAN 111 for both S-Series devices on Router R1's interface, 172.111.1.1, and Router R2's interface, 172.111.1.2. Each virtual router is given a different virtual IP address that is used as a default gateway by a subset of hosts that reside of the LAN

segment. Because interfaces on Router R1 and Router R2 for VLAN 111 are configured as belonging to VRID 1, 2, and 3, VRRP will support resiliency between these interfaces if one interface becomes in-operational.

To load balance traffic generated from the hosts on the 172.111.0.0/16 network, the hosts are partitioned into being configured with default gateways matching the virtual IP address of the VRRP virtual routers, and the VRRP Master for each VRRP instance is configured for distribution across Router R1 and Router R2. It is known that Router R1's interface, 172.111.1.1, will become Master for VRID 1 because it is the IP address owner for the virtual router. This interface is also configured to be Master for VRID 3 by raising its VRRP priority in VRRP instance 3 to 200. Therefore, Router R1's interface 172.111.1.1 will be Master for VRID 1 and VRID 3 handling traffic on this LAN segment sourced from subnets 172.111.0.0/18 and 172.111.128.0/18. Router R2's interface is configured to be the Master for VRID 2 by raising its VRRP priority in VRRP instance 2. Therefore, Router R2's interface 172.111.1.2 will be Master for VRID 2 handling traffic on this LAN segment sourced from subnets 172.111.64.0/18.

In this configuration, an interface on VLAN 111 for Router R1 or Router R2, or VRID 1, 2, or 3 fails, the interface on the other router will take over for forwarding outside the local LAN segment.

Router R1

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 111
S Chassis(rw-config-intf-vlan.0.111)->vrrp create 1 v2-IPv4
S Chassis(rw-config-intf-vlan.0.111)->vrrp address 1 172.111.1.1
S Chassis(rw-config-intf-vlan.0.111)->vrrp enable 1
S Chassis(rw-config-intf-vlan.0.111)->vrrp create 2 v2-IPv4
S Chassis(rw-config-intf-vlan.0.111)->vrrp address 2 172.111.1.50
S Chassis(rw-config-intf-vlan.0.111)->vrrp enable 2
S Chassis(rw-config-intf-vlan.0.111)->vrrp create 3 v2-IPv4
S Chassis(rw-config-intf-vlan.0.111)->vrrp address 3 172.111.1.150
S Chassis(rw-config-intf-vlan.0.111)->vrrp priority 3 200
S Chassis(rw-config-intf-vlan.0.111)->vrrp enable 3
S Chassis(rw-config-intf-vlan.0.111)->no shutdown
S Chassis(rw-config-intf-vlan.0.111)->exit
S Chassis(rw-config)->
```

Router R2

```
S Chassis(rw)->configure
S Chassis(rw-config)->interface vlan 111
S Chassis(rw-config-intf-vlan.0.111)->vrrp create 1 v2-IPv4
S Chassis(rw-config-intf-vlan.0.111)->vrrp address 1 172.111.1.1
S Chassis(rw-config-intf-vlan.0.111)->vrrp enable 1
S Chassis(rw-config-intf-vlan.0.111)->vrrp create 2 v2-IPv4
S Chassis(rw-config-intf-vlan.0.111)->vrrp address 2 172.111.1.50
S Chassis(rw-config-intf-vlan.0.111)->vrrp priority 2 200
S Chassis(rw-config-intf-vlan.0.111)->vrrp enable 2
S Chassis(rw-config-intf-vlan.0.111)->vrrp create 3 v2-IPv4
S Chassis(rw-config-intf-vlan.0.111)->vrrp address 3 172.111.1.150
S Chassis(rw-config-intf-vlan.0.111)->vrrp enable 3
S Chassis(rw-config-intf-vlan.0.111)->no shutdown
```

```
S Chassis(rw-config-intf-vlan.0.111)->exit
S Chassis(rw-config)->
```

Terms and Definitions

Table 48-3 lists terms and definitions used in this VRRP configuration discussion.

Table 48-3 VRRP Configuration Terms and Definitions

Term	Definition
Accept Mode	When enabled, it allows the master for this virtual router to accept IP packets for the configured associated IP address list.
Backup	The set of VRRP routers available to assume forwarding responsibility for a virtual router should the current Master fail.
IP Address owner	The VRRP router that has the virtual router's IP address(es) as real interface address(es). This is the router that, when up, will respond to packets addressed to one of these IP addresses for ICMP pings, TCP connections, etc.
Master	The VRRP router that is assuming the responsibility of forwarding packets sent to the IP address(es) associated with the virtual router, and answering ARP requests for these IP addresses.
Priority	The priority field specifies the sending VRRP router's priority for the virtual router. Higher values equal higher priority. This field is an 8 bit unsigned integer field. The priority value for the VRRP router that owns the IP address(es) associated with the virtual router MUST be 255 (decimal). VRRP routers backing up a virtual router MUST use priority values between 1-254 (decimal). The default priority value for VRRP routers backing up a virtual router is 100 (decimal). The priority value zero (0) has special meaning indicating that the current Master has stopped participating in VRRP. This is used to trigger Backup routers to quickly transition to Master without having to wait for the current Master to timeout.
Virtual Router	An abstract object managed by VRRP that acts as a default router for hosts on a shared LAN. It consists of a Virtual Router Identifier and a set of associated IP address(es) across a common LAN. A VRRP Router may backup one or more virtual routers.
VRID	Virtual Router ID — a unique number associated with each virtual router.
VRRP fabric route mode	A VRRP feature that allows a VRRP instance in the backup state to forward IPv4 and IPv6 packets destined for the VRRP MAC address.
VRRP Router	A router running the Virtual Router Redundancy Protocol. It may participate in one or more virtual routers. A VRRP router may associate a virtual router with its real addresses on an interface, and may also be configured with additional virtual router mappings and priority for virtual routers it is willing to backup.

49

Security Configuration

This document provides the following information about configuring security features on the Extreme Networks S-Series platforms.

Using Security Features in Your Network

The S-Series platform supports the following security features.

Note: The security feature Flow Setup Throttling (FST) is also supported by the S-Series platform (see Chapter 50, **Flow Setup Throttling Configuration** for a detailed discussion of the FST feature).

MAC Locking

The MAC locking security feature provides for limiting access to a port to specified MAC addresses or a maximum number of MAC addresses on a first come first serve basis. In the first case, a device with a MAC address that is not specifically configured will not be allowed access to a port. This provides the network administrator with confidence that only known devices will gain access to a port. The second case provides a means of controlling the maximum number of unique devices that will have access at any given time to the port configured for this mode of MAC locking.

Secure Shell

The Secure Shell (SSH) security feature provides a secure encrypted communications method between a client and the switch providing data privacy and integrity that is an alternative to the unsecure Telnet protocol. Using SSH the entire session is encrypted, including the transmission of user names and passwords, and negotiated between a client and server both configured with the SSH protocol. Telnet sessions are insecure. All data is sent unencrypted. Use SSH instead of Telnet when the security of login and data transmission is a concern.

The S-Series device supports both public key and password authentication methods.

TACACS+

The TACACS+ security feature provides an alternative to RADIUS for the authentication of devices desiring access to the network. TACACS+ provides device authentication, session authorization, and per-command authorization, as well as accounting on a session and per command basis.

Host Denial of Service (DoS)

The Host DoS security feature provides protection from all known attack vectors commonly used to deny service to the management entity of an Extreme Networks S-Series switch router. TCP, UDP and ICMP communications are monitored and reported on. Each attack type can be individually enabled and provides feed back in the form of display counters and SYSLOG messages when attacks are detected and prevented.

Implementing Security

Take the following steps to implement supported S-Series security features in your network:

- To implement MAC locking:
  - Enable MAC locking both globally and on the ports to be configured for MAC locking
  - For ports that you are going to restrict access based upon a device's MAC address, set the port to MAC lock static and specify the maximum number of configure MAC addresses for that port
  - For ports you are going to restrict on a first come first serve for a set number of MAC addresses, enable dynamic MAC locking specifying the maximum number of MAC addresses allowed for that port
  - Optionally move all current dynamically enabled MAC locking MAC addresses to a static MAC locking configuration
  - Optionally allow dynamic MAC addresses to age based upon the configured MAC agetime
  - Optionally enable MAC lock trap messaging
- To implement Secure Shell:
  - Enable the SSH server
  - Set or reinitalize the host key
  - Verify the SSH state
- To implement TACACS+:
  - Enable TACACS+ on the client
  - Configure the TACACS+ server to be used by the client
  - Optionally enable TACACS+ session accounting
  - Optionally configure the TACACS+ session authorization service or privilege level
  - Optionally enable per command authorization
  - Optionally enable the TCP single connection feature for this device
- To implement Host DoS:
  - Enable one or more DoS attack mitigation types

- Optionally set a logging event rate per a specified amount of time
- Optionally enable logging
- Verify the Host DoS configuration

Security Overview

MAC Locking

MAC Locking, sometimes referred to as MAC-based port locking, port locking, or port security, helps prevent unauthorized access to the network by limiting access based on a device's MAC address. MAC locking locks a port to one or more MAC addresses, preventing connection of unauthorized devices via a port. With MAC locking enabled, the only frames forwarded on a MAC locked port are those with the configured or dynamically selected MAC addresses for that port.

There are two different types of MAC locking:

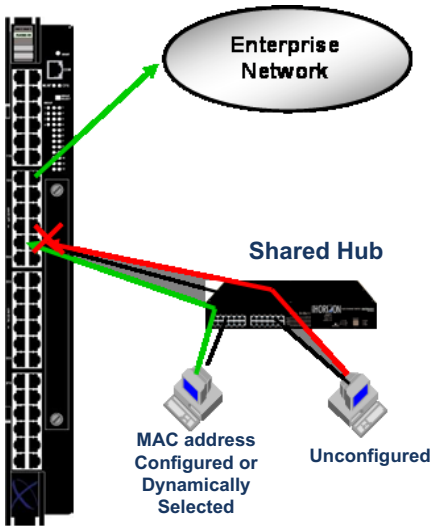
- Static MAC Locking - Locking one or more specified MAC addresses to a port.
- Dynamic MAC Locking - Locking one or more MAC addresses to a port based on first arrival of received frames after dynamic MAC locking is enabled. The configuration specifies the maximum number of end users that will be allowed. As each new end user is identified, it is MAC locked up to the maximum number of users. Once the maximum number of users have been MAC locked, all other users will be denied access to the port until a MAC locked address is either aged, if aging is configured, or the session for that user ends.

MAC Locking is disabled by default. MAC locking must be both globally enabled and enabled on the desired ports. When globally enabling MAC lock you can optionally specify the port or ports to enable, or enable MAC locking on all ports. Once enabled, ports can be configured for either static or dynamic MAC locking. When configuring static MAC locking, specify the user MAC address and the port string for that user. When configuring dynamic MAC locking, specify the port and the maximum number of users that will be dynamically MAC locked. MAC addresses that are currently dynamically active can be auto reconfigured as static using the **set maclock move** command for the specified port.

Dynamic MAC lock address aging can be enabled per port. If the Filter DataBase (FDB) entry ages out for this station, the corresponding dynamic MAC locked stations will no longer be MAC locked. The age time for the FDB is set by the **set mac agetime** command and is displayed using the **show mac agetime** command. Dynamic MAC lock address aging is disabled by default.

Figure 49-1 displays two users on a shared hub connected to an S-Series switch port. Data from the MAC locked user is forwarded on to the enterprise network. Data from the unconfigured user is dropped.

Figure 49-1 Blocking Unauthorized Access with MAC Locking



Secure Shell

Secure Shell (SSH) is a protocol for secure remote login over an insecure network. SSH provides a secure substitute for Telnet by encrypting communications between two hosts.

The S-Series SSHv2 implementation includes:

- Data privacy
- Communication integrity

An SSH server resides on the S-Series platform and listens for client connection requests. Once a request is authenticated, a secure connection is formed through which all subsequent traffic is sent. All traffic is encrypted across the secure channel, which ensures data integrity. This prevents someone from seeing clear text passwords or file content, as is possible with the Telnet application.

Once SSH has been enabled and the S-Series has at least one valid IP address, you can establish an SSH session from any TCP/IP based node on the network, by using SSH to connect to an IP address, and entering your user name and password. Refer to the instructions included with your SSH application for information about establishing a session.

SSH is activated by enabling the SSH server on the device. Enabling the server automatically generates a host key for the server, used during the life of the client to server connection. The SSH server can be reinitialized. Reinitializing the server clears all current client to server connections. Reinitializing the server does not reinitialize the host key. Should you believe the host key has been compromised, or otherwise wish to change it, the host key can be reinitialized with a separate command.

During the handshake between an SSH client and an SSH server, each side sends a proposal of cartographic Ciphers and Message Authentication Code (MAC)s. SSH Ciphers and MACs are applied to all new inbound (SSH server) and outbound (SSH client) SSH sessions. Existing sessions remain unchanged. Ciphers and MACs are entered in order of precedence from high to low.

Applied SSH Ciphers default to all supported ciphers in the following order of precedence: aes128-cbc, aes192-cbc, aes256-cbc, 3des-cbc, blowfish-cbc, and cast128-cbc.

When in FIPS mode, only the following FIPS compliant Ciphers are allowed (listed in the default order of precedence from high to low): aes128-cbc, aes192-cbc, aes256-cbc, and 3des-cbc. If non-FIPS Ciphers are configured when booting in FIPS mode, SSH uses the default Cipher list.

Use the **set ssh ciphers** command to administratively change the applied SSH Ciphers list. When using this command the order of precedence is modified to the order the Ciphers are entered. Any supported Cipher not entered is no longer allowed.

Applied MACs default to all supported MACs in the following order of precedence: hmac-sha1-etm @openssh.com, hmac-md5-etm @openssh.com, hmac-ripemd160-etm@openssh.com, hmac-sha1-96-etm @openssh.com, hmac-md5-96-etm @openssh.com, hmac-sha1, hmac-md5, hmac-ripemd160, hmac-ripemd160 @openssh.com, hmac-sha1-96, and hmac-md5-96.

When in FIPS mode, only the following FIPS compliant MACs are allowed (listed in the default order of precedence from high to low): hmac-sha1 and hmac-sha1-96. If non-FIPS MACs are configured when booting in FIPS mode, SSH uses the default MACs list.

Use the **set ssh macs** command to administratively change the applied SSH MACs list. When using this command the order of precedence is modified to the order the MACs are entered. Any supported MAC not entered is no longer allowed.

TACACS+

TACACS+ (Terminal Access Controller Access Control System Plus) is a security protocol that can be used as an alternative to the standard RADIUS security protocol. The client function is implemented on the S-Series device to control access to this device in conjunction with a remote server. TACACS is defined in RFC 1492, and TACACS+ is defined in an un-published and expired Internet Draft draft-grant-tacacs-02.txt, “The TACACS+ Protocol Version 1.78", January, 1997.

TACACS+ client functionality falls into four basic capabilities: authentication and session authorization, per-command authorization, session accounting, and per-command accounting.

When the single connect feature is enabled, the TACACS+ client will use a single TCP connection for all requests to a given TACACS+ server.

Session Authorization and Accounting

The TACACS+ client is disabled by default. When the TACACS+ client is enabled on the S-Series, using the **set tacacs enable** command, the session authorization parameters configured with the **set tacacs session authorization** command are sent by the client to the TACACS+ server when a session is initiated. The parameter values must match a service and access level attribute-value pairs configured on the server for the session to be authorized. If the parameter values do not match, the session will not be allowed. The service name and attribute-value pairs can be any character string, and are determined by your TACACS+ server configuration.

When session accounting is enabled, using the **set tacacs session accounting** command, the TACACS+ server will log accounting information, such as start and stop times, IP address of the remote user, and so forth, for each authorized client session. Once session accounting has been enabled, you can disable it with this command.

The S-Series device is informed of the TACACS+ server properties using the **set tacacs server** command. You can configure the timeout value for all configured servers or a single server, or you can configure the IP address, TCP port, and secret for a single server, specifying a server index value for this server.

Per-Command Authorization and Accounting

In order for per-command accounting or authorization by a TACACS+ server to take place, the **set tacacs** command must be executed within an authorized session.

When per-command accounting is enabled, using the **set tacacs command accounting** command, the TACACS+ server will log accounting information, such as start and stop times, IP address of the client, and so forth, for each command executed during the session.

When per-command authorization is enabled, using the **set tacacs command authorization** command, the TACACS+ server will check whether each command is permitted for that authorized session and return a success or fail. If the authorization fails, the command is not executed.

Single TCP Connection for All TACACS+ Requests

The S-Series device can be configured to use a single TCP connection for all TACACS+ client requests to a TACACS+ server. Use the **set tacacs singleconnect** command to enable this feature on the S-Series device.

Host DoS

The Host DoS feature provides protection against all known DoS attack mitigation types.

Table 49-2 lists the configurable Host DoS mitigation types.

Table 49-1 Host DoS Mitigation Types

Threat	Description	Action
Excessive Arp or ND	Reception of an excessive number of ARP or ND frames from a single host.	Frames are discarded.
Bad SIP	Frames with a source IP address equal to multicast or broadcast.	Frames are discarded.
Spoof	Frames with a source IP address that is same as this router's interface address.	Frames are discarded.
Christmas Tree	Frames with an invalid TCP flag combination.	SYN+FIN and SYN+RST frames are discarded.
Fragmented ICMP	ICMP packets are fragmented.	All ICMP fragmented packets are discarded.
ICMP Flood	Excessive number of ICMP packets received.	Receipt of ICMP packets is limited to a user configurable limit of packets per second.
Large ICMP	ICMP packets exceed the configured maximum ICMP size.	ICMP packets exceeding the configured maximum ICMP size are discarded.
Multicast/Broadcast Source address	Packets with a Multicast or Broadcast source IP address.	Packets with a Multicast or Broadcast source IP address are discarded.
LANd	Packets with the destination IP address equal to the source IP address.	Packets with the destination IP address equal to the source IP address are discarded.
Smurf	A vulnerability due to ICMP directed broadcast packets.	ICMP directed broadcast packets are discarded.
Fraggle Attack	A vulnerability due to UDP directed broadcast packets.	UDP directed broadcast packets are discarded.
SYN Flood	Packets exceeding the maximum value and maximum establishment rate per source IP address or regardless of source.	Packets beyond established rates are discarded.
Port Scan	Packets exceeding the maximum value and maximum establishment rate.	Packets beyond established rates are discarded.

Globally enable host DoS for this device using the **hostdos enable** command. Host DoS is globally enabled by default. Entering a command line for each threat, **s**pecify the mitigation-type, in the **hostdos** command in global configuration command mode, to enable the specific DoS attack type to be mitigated.

The ICMP maximum allowed length can be set using the **hostdos** command **icmp-maxlength** option.

Configuring Security

Table 49-2 lists Security parameters and their default values.

Table 49-2 Default Security Parameters

Parameter	Description	Default Value
MAC locking status	Specifies whether MAC locking is enabled or disabled both globally and on a specific port.	disabled
maximum number of dynamic MAC addresses	Specifies the maximum number of MAC addresses that will be locked on a port configured for dynamic MAC locking.	600
first arrival MAC address aging	Specifies that dynamic MAC locked addresses will be aged after the time set by the MAC agetime configuration.	disabled
MAC lock traps	Specifies whether traps associated with MAC locking will be sent.	disabled
maximum number of static MAC addresses	Specifies the maximum number of static MAC addresses allowed on a port.	64
SSH state	Specifies whether the SSH protocol is enabled or disabled on this device.	disabled
TACACS+ state	Specifies whether the TACACS+ protocol is enable or disabled on this device.	disabled
TACACS+ server timeout	Specifies the TACACS+ server timeout for the all TACACS+ servers.	10 seconds
session privilege level	Specifies the attribute value for the TACACS+ session management privilege level.	read-only = 0 read-write = 1 super-user = 15
TACACS+ single connect state	Specifies whether the TACACS+ single connect feature is enabled or disabled.	disabled

Configuring MAC Locking

Procedure 49-1 describes how to configure MAC locking on an S-Series device. All MAC locking commands can be entered in any command mode.

Procedure 49-1 MAC Locking Configuration

Step	Task	Command(s)
1.	Globally enable MAC locking, optionally specifying the port(s) to be enabled. If no port is specified, all ports on the device are enabled. If one or more ports are specified, all unspecified ports remain disabled.	**set maclock enable** [*port_string*]

Procedure 49-1 MAC Locking Configuration (continued)

Step	Task	Command(s)
2.	Optionally, enable static MAC locking configuration on the specified port for the maximum number of MAC addresses specified by *value*.	**set maclock static** *port_string value*
3.	Optionally, create static MAC locking entries for the specified MAC address and port.	**set maclock** *mac_address port_string* {**create** | **enable** | **disable**}
4.	Optionally, create a dynamic MAC locking configuration, specifying the maximum number MAC addresses allowed for the specified port.	**set maclock firstarrival** *port_string value*
5.	Optionally, move all current dynamic MAC locking configured MACs to static entries.	**set maclock move** *port-string*
6.	Optionally, enable or disable first arrival MAC address aging on the specified port(s).	**set maclock agefirstarrival** *port_string* {**enable** | **disable**}
7.	Optionally, enable or disable MAC lock trap messaging.	**set maclock trap** *port_string* {**enable** | **disable**}

Table 49-3 describes how to manage MAC locking on an S-Series port. All MAC locking commands can be entered in any command mode.

Table 49-3 Managing MAC Locking

Step	Task	Command(s)
1.	Display MAC locking configuration information for dynamic configurations, static configurations or by port.	**show maclock** [**stations** [**firstarrival** | **static**]] [*port_string*]
2.	Clear dynamic MAC locking configuration by port.	**clear maclock firstarrival** *port-string*
3.	Clear static MAC locking configuration by port.	**clear maclock static** *port_string*
4.	Clear MAC locking from one or more static MAC addresses.	**clear maclock** {**all** | *mac-address*} *port-string*

MAC Locking Configuration Example

The following command line enables MAC locking both globally for the device and at the port level for ports **ge.1.1** through **5**:

```
S Chassis(rw)->set maclock enable ge.1.1-5
S Chassis(rw)->
```

The following command lines enable **port ge.1.1** for a maximum of **3** static MAC address entries. This is followed by four static MAC address creation entries. The fourth entry fails because the maximum allowed has been set to 3:

```
S Chassis(rw)->set maclock static ge.1.1 3
S Chassis(rw)->set maclock 00-10-a4-e5-08-4e ge.1.1 create
S Chassis(rw)->set maclock 08-00-20-7c-e0-db ge.1.1 create
S Chassis(rw)->set maclock 00-60-08-14-4b-15 ge.1.1 create
S Chassis(rw)->set maclock 00-01-f4-2c-ad-b4 ge.1.1 create
Set failed for ge.1.1.
```

```
S Chassis(rw)->show maclock stations static

Port Number   MAC Address         Status          State           Aging
-----------   -----------------   -------------   -------------   -----
ge.1.1        00-10-a4-e5-08-4e   active          static          false
ge.1.1        00-60-08-14-4b-15   active          static          false
ge.1.1        08-00-20-7c-e0-db   active          static          false
S Chassis(rw)->
```

The following command lines configure ports **ge.1.2** through **5** for dynamic MAC locking with a maximum of **15** users on each port. This line is followed by a line enabling MAC locking trap messaging on ports **ge.1.1** through **5**:

```
S Chassis(rw)->set maclock firstarrival ge.1.2-5 15
S Chassis(rw)->set maclock trap ge.1.1-5 enable
S Chassis(rw)->
```

Configuring Secure Shell

Procedure 49-2 describes how to configure Secure Shell on an S-Series device. Secure Shell commands can be entered in any command mode.

Procedure 49-2 SSH Configuration

Step	Task	Command(s)
1.	Enable, disable, or reinitialize the SSH server.	**set ssh** {**enable** | **disable** | **reinitialize**}
2.	Set or reinitialize the host key on the SSH server.	**set ssh hostkey** [**reinitialize**]
3.	Modify the SSH Ciphers list for all future sessions on this system.	**set ssh ciphers** {**aes128-cbc** | **aes192-cbc** | **aes256-cbc** | **3des-cbc** | **blowfish-cbc** | **cast128-cbc** | **rijndael-cbc@lysator.liu.se**}
4.	Modify the SSH MACs list for all future sessions on this system.	**set ssh macs** {**hmac-sha1-etm@openssh.com** | **hmac-md5-etm@openssh.com** | **hmac-ripemd160-etm@openssh.com** | **hmac-sha1-96-etm@openssh.com** | **hmac-md5-96-etm@openssh.com** | **hmac-sha1** | **hmac-md5** | **hmac-ripemd160** | **hmac-ripemd160@openssh.com** | **hmac-sha1-96** | **hmac-md5-96**}
5.	Verify the SSH state.	**show ssh state**

SSH Configuration Example

The following commands enable and verify SSH:

```
S Chassis(rw)->set ssh enable
S Chassis(rw)->show ssh state
SSH Server state: Enabled
S Chassis(rw)->
```

The following command reinitializes the host key on the SSH server:

```
S Chassis(rw)->set ssh hostkey reinitialize
```

Configuring TACACS+

Procedure 49-3 describes how to configure TACACS+ on an S-Series device. TACACS+ commands can be entered in any command mode.

Procedure 49-3 TACACS+ Configuration

Step	Task	Command(s)
1.	Enable or disable the TACACS+ client.	**set tacacs** {**enable** | **disable**}
2.	Configure the TACACS+ server(s) to be used by the TACACS+ client.	**set tacacs server** {*index* [*ipaddress port* [*secret*]] | **all timeout** *timeout*}
3.	Optionally, enable TACACS+ session accounting	**set tacacs session accounting enable**
4.	Optionally, configure the TACACS+ session authorization service or privilege level. The attribute for privilege level is: **priv-lvl**.	**set tacacs session** {**authorization service** *name* | **read-only** *attribute value* | **read-write** *attribute value* | **super-user** *attribute value*}
5.	Optionally, enable per command accounting within an authorized session.	**set tacacs command accounting enable**
6.	Optionally, enable per command authorization.	**set tacacs command authorization enable**
7.	Optionally, enable the TCP single connection feature for this device.	**set tacacs singleconnect enable**

Table 49-4 describes how to manage TACACS+ on an S-Series device. All TACACS+ commands can be entered in any command mode.

Table 49-4 Managing TACACS+

Task	Command(s)
Display TACACS+ configuration or state.	**show tacacs** [**state**]
Display the current TACACS+ server configuration.	**show tacacs server** {*index* | **all**}
Clear the TACACS+ server configuration or reset the server timeout to the default value.	**clear tacacs server** {**all** | *index*} [**timeout**]
Display the current TACACS+ client session settings.	**show tacacs session** {**authorization** | **accounting**} [**state**]
Reset TACACS+ session authorization settings to their default values.	**clear tacacs session authorization** { [**service**] [**read-only**] [**read-write**] [**super-user**] }
Display the current TACACS+ single connect state.	**show tacacs singleconnect** [**state**]

TACACS+ Configuration Example

The following command enables TACACS+ on the TACACS+ client for this device:

```
S Chassis(rw)->set tacacs enable
```

The following commands configure and verify two TACACS servers for this device to indexes **1** and **2**. Index **1** has an IP address of **10.10.10.20** on port **49** with a secret **mysecret1**. Index **2** has an IP address of **10.10.10.30** on port **49** with a secret of **mysecret2**. The server timeout value will remain at the default of 10 seconds.

```
S Chassis(rw)->set tacacs server 1 10.10.10.20 49 mysecret1
S Chassis(rw)->set tacacs server 2 10.10.10.30 49 mysecret2
S Chassis(rw)->show tacacs server all
```

```
 TACACS+ Server  IP Address       Port   Timeout  Status
 --------------  ---------------  -----  -------  -------
 1               10.10.10.20      49     10       Active
 2               10.10.10.30      49     10       Active
S Chassis(rw)->
```

The following command enables and verifies session authorization for the exec service:

```
S Chassis(rw)->set tacacs session authorization service exec
S Chassis(rw)->show tacacs session authorization
 TACACS+ service:                      exec
 TACACS+ session authorization A-V pairs:
         access level attribute                          value
         read-only   'priv-lvl'                          '0'
         read-write  'priv-lvl'                          '1'
         super-user  'priv-lvl'                          '15'
S Chassis(rw)->
```

The following commands enable and verify session accounting, followed by commands that enable both accounting and authorization on a per command basis, for this device:

```
S Chassis(rw)->set tacacs session accounting enable
S Chassis(rw)->show tacacs session accounting
 TACACS+ session accounting state:     enabled
S Chassis(rw)->set tacacs command accounting enable
S Chassis(rw)->set tacacs command authorization enable
S Chassis(rw)->
```

The following command enables the TCP single connection feature for this device:

```
S Chassis(rw)->set tacacs singleconnect
S Chassis(rw)->
```

Configuring Host DoS

Procedure 49-4 describes how to configure Host DoS on an S-Series device. Host DoS configuration commands are entered in global configuration command mode.

Procedure 49-4 Host DoS Configuration

Step	Task	Command(s)
1.	Enable host DoS globally for this device. Threats must be specifically enabled for mitigation to occur for that threat as specified in the following step.	**hostdos enable**
2.	Enable a mitigation type, and optionally set the rate at which events will be acted upon.	**hostdos** {*mitigation-type* | **enable** | **icmp-maxlength** *icmp-maxlength*} [**rate** *count* [**per-second** | **per-minute** | **per-hour** | **per-day**]] [**nolog**]
3.	Optionally, disable logging for the specified DoS attack types.	**hostdos** *mitigation-type* **nolog**
4.	Optionally, specify an ICMP maximum packet size for **icmpsize** mitigation.	**hostdos icmpsize maxlength** *length*

Table 49-4 describes how to display Host DoS configuration state and counters on an S-Series device.

Table 49-5 Displaying Host DoS

Step	Task	Command(s)
1.	Display configuration state for one or all Host DoS attack mitigation types.	**show hostdos** [*mitigation-type*]
2.	Display statistic counters for one or all Host DoS attack mitigation types.	**show hostdos** [*mitigation-type*] [**stats**]

Host DoS Configuration Example

This example shows how to:

- Globally enables host Dos on this device
- Enable the checkSpoof mitigation type, with a log display rate of 5 per-minute
- Enable the XmasTree mitigation type and disable logging for this threat

```
S Chassis(rw-config)->hostdos enable
S Chassis(rw-config)->hostDoS spoof rate 5 per-minute
S Chassis(rw-config)->hostdos xmastree nolog
S Chassis(rw-config)->show hostDoS
 hostDoS is globally enabled
 hostDoS icmp-maxlength is 1024
 hostDoS Spoof is enabled , logging is enabled , rate is   5 per-minute
 hostDoS XmasTree   is enabled , logging is disabled, rate is   0 per-second
 hostDoS IcmpFrag   is disabled, logging is enabled , rate is   0 per-second
 hostDoS IcmpFlood  is disabled, logging is enabled , rate is   0 per-second
 hostDoS IcmpSize   is disabled, logging is enabled , rate is   0 per-second
 hostDoS BadSIP     is disabled, logging is enabled , rate is   0 per-second
 hostDoS LANd       is disabled, logging is enabled , rate is   0 per-second
 hostDoS Smurf      is disabled, logging is enabled , rate is   0 per-second
 hostDoS Fraggle    is disabled, logging is enabled , rate is   0 per-second
 hostDoS SynFlood   is disabled, logging is enabled , rate is   0 per-second
 hostDoS PortScan   is disabled, logging is enabled , rate is   0 per-second
 hostDoS TearDrop   is disabled, logging is enabled , rate is   0 per-second
S Chassis(rw-config)->
```

50

Flow Setup Throttling Configuration

This document provides the following information about configuring flow setup throttling on the Extreme Networks S-Series platforms.

Using Flow Setup Throttling in Your Network

Flow Setup Throttling (FST) is a proactive feature designed to mitigate zero-day threats and Denial of Service (DoS) attacks before they can wreak havoc on the network. FST directly combats the effects of zero-day and DoS attacks by limiting the number of new or established flows that can be programmed on any individual switch port. This feature, combined with other Extreme Networks security solutions, can slow down and even stop viruses before the available network bandwidth is saturated. This is achieved by monitoring the new flow arrival rate and controlling the maximum number of allowable flows. The FST processes are defined and administered by means of the enterasys-flow-limiting-mib.

FST lets you define port behaviors using a set of up to 10 port classification types. Each port classification type is configured for a low- and high-limit flow threshold. When the number of active flows on a port reaches a threshold, the action associated with that threshold is taken. Actions include sending SNMP traps, dropping flows that exceed a threshold, and disabling interfaces.

Implementing Flow Setup Throttling

To configure FST for a given port classification:

1. Determine an appropriate flow baseline from which flow limits can be set for each port classification type by monitoring the ports associated with each port classification.
2. Set the low- and high-limit actions to be taken for the specified port classification.
3. Set the ports that will use the configured port classification.
4. Enable FST on all ports configured for flowlimiting.

5. Optionally, enable the sending of SNMP traps action globally on the device.
6. Optionally, enable the disable port action globally on the device.
7. Enable FST on the device.
8. Verify the configuration or monitor baseline configurations using the FST show commands.

Flow Setup Throttling Overview

What is a Flow?

A flow is a stream of packets that has not yet met an expiration criteria, in which the value of a subset of L2, L3, and L4 fields appropriate to the communication exchange are the same for each packet in the stream. ASIC technology implemented on S-Series devices provides for line-rate packet field investigation for the setup and tracking of flows. A flow is unidirectional, and is defined after the first packet is encountered. A network conversation consists of two separate flows, one in each direction. Upon inactivity, a given flow times out after a product-specific timer expires.

Where is Flow Setup Throttling Configured?

FST is used to monitor flows throughout the network, providing notification when flow limits are exceeded. Because issues tend to originate on ingress at the user edge, FST is ideally used to actively limit flows on user edge ports only. Actions taken on Inter-Switch Link (ISL) ports can be difficult to recover from. Creating too many flow monitors at the network core, and dropping flows, or disabling ports in the core, is not an optimal design strategy, and should be avoided.

Determining a Port Classification Flow Baseline

In a well-managed network, begin by measuring normal flow levels to determine the proper limits for a given port classification. The firmware tracks flows regardless of whether FST is enabled. Before configuring and enabling a set of FST limits, use the **show flowlimit stats** command to form a baseline over time for the ports you wish to configure FST on. This baseline is defined as the highest level of flows seen on a port classification type under normal operating conditions: a port not under DoS or zero-day threat. Set the flow limits for each port classification by:

- Adjusting the high-level limit to be perhaps 50 - 100% higher than the determined baseline for the port classification
- Adjusting the low-level limit to be just above the baseline for the port classification

The idea is to only involve flow management when an event worthy of examination occurs. This baseline will vary according to how the port is used in the network. That is why each port should be set to a traffic classification with appropriate associated limits and actions.

Once the baselines for an FST port classification are determined, implement FST as defined in “Implementing Flow Setup Throttling” on page 50-1 and fully described below.

Setting the Port Classification

Each FST enabled port is classified based upon its position in the network. Each port enabled for FST can be classified as either a:

- User defined classification - a classification other than the pre-defined classifications, represented by a numeric value.

- User port - an edge port with one user attached to it.
- Server port - a port with a server attached to it. This class may encompass a wide range of server types from a small workgroup print server to an enterprise exchange server. Alternately, an administrator may choose to configure an interface with a small print server as a user port given that its flow setup needs may be similar to that of a user port.
- Aggregated user port - a port likely to have multiple end stations attached either through a wireless access point or an unmanaged low cost hub or switch. It is expected that this class may also be used instead of the Inter-Switch Link class when switches are interconnected using a lower speed link.
- Inter-Switch Link - a port that is used as a high-speed interconnect between two intelligent switches or routers.
- Unspecified port - a port in which nothing can be assumed about its intended use.

Note: Port classifications function only as traffic classification guidelines. Each port classification can be configured with any set of limits, and any interface can be associated with any port classification.

Setting Flow Limits and Associated Actions

FST provides for the setting of two limits and an associated action per flow. The first limit sets a low-level flow threshold and an associated action. The second limit sets a high-level flow threshold and an associated action. Setting a limit to **0** disables that limit.

Note: The command to set the flowlimit action is additive in that it adds the specified action to the current list of actions for the specified port classification. To remove an action already in the actions list for the current context, use the clear command.

Associated actions when the flow limit is reached can be set to:

- Notify – This option sends out an SNMP trap notification when the associated threshold is exceeded. If the flowlimit threshold is exceeded, a single notification is sent out. The notification action is reset when the number of flows drops below the flowlimit threshold. In order for SNMP traps to be sent as a result of this option, the notify action must be both associated with one or more port classifications and globally enabled on the device.

  When globally enabling notification on the device, a notification interval option can be set. The specified interval sets the number of seconds to wait before generating another notification of the same type for the same interface. This allows notification generation to be throttled in the case of a flow counter or rate that is repeatedly transitioning across a threshold. A value of **0** indicates that the device should not suppress any notifications related to the flowlimiting.
- Drop – This action drops flow setup requests in excess of the configured limit and discards the associated packets. The use of this option could cause the device to repetitively process setup requests for the dropped flows. The process of dropping flow setup requests and their associated packets could cause end stations attached to this interface to behave in an indeterminate manner. The use of this option may also prevent the device from being able to count additional flows and from reaching any additional configured limits.
- Disable – This option operationally disables the interface. The interface operational status is set to the down state. The interface remains in the down state until the associated FST interface status is set to operational using the **set flowlimit port** command, the FST feature is disabled, or the device is reset. In order for a port to be disabled as a result of this option, the disable action must be associated with one or more port classifications and globally enabled on the device using the **set flowlimit shutdown** command.

Sending out an SNMP trap notification is often times used as the low-level limit action. Dropping excess flows or even disabling the port can be appropriate high-level limit actions.

Flowlimit Action Precedence

If the notify action is a configured action, globally enabled, and does not exceed the global rate limit for notifications, the SNMP trap notification will always be sent, and is not subject to precedence. The notification is sent out after other actions have been performed and indicates the condition on the interface after any other actions have taken place.

If one or more other actions are configured, only the one with the highest precedence will be performed. The order of precedence, from highest to lowest, is disable and drop.

Configuring Flow Setup Throttling

This section provides details for the configuration of FST on the S-Series product.

Table 50-1 lists FST parameters and their default values.

Table 50-1 Default Flow Setup Throttling Parameters

Parameter	Description	Default Value
action1	Specifies the action associated with the low-limit (limit1) for a given port classification	notify
action2	Specifies the action associated with the high-limit (limit2) for a given port classification.	disable and notify
flowlimit global state	Specifies whether FST is enabled or disabled globally on the device.	disabled
flowlimit interface state	Specifies whether FST is enabled or disabled on a specified interface	enabled
interface disable global state	Specifies whether the disable interface action is enabled or disabled globally on the device.	disabled
notification global state	Specifies whether notification is enabled or disabled globally on the device.	enabled
notification interval	Specifies the number of seconds to wait before generating another notification of the same type on the same interface.	120 seconds
port classification	Specifies the type of port for a given flowlimit and action.	unspecified

Procedure 50-1 describes how to configure FST.

Procedure 50-1 Configuring FST

Step	Task	Command(s)
1.	Set the low- and high-limit values for each traffic classification to be applied to network ports. • **limit1** – The low-limit option to which the specified limit is applied. • **limit2** – The high-limit option to which the specified limit is applied. • *limit* – specifies flows threshold for each limit type. • *class-index* – Specifies a numeric value for the class user classification type to assign to this action. • **userport** – Specifies the configured limit will be applied to an edge port with a single attached user. Default values: limit1 = 800, limit2 = 1000. • **serverport** – Specifies the configured limit will be applied to a port with a server attached to it. Default values: limit1 = 5000, limit2 = 6000. • **aggregateduser** – Specifies the configured limit will be applied to an edge port with multiple users attached to it. Default values: limit1 = 5000, limit2 = 6000. • **interswitchlink** – Specifies the configured limit will be applied to a high speed interconnect port between switches or routers. Default values: limit1 = 14000, limit2 = 16000. • **unspecified** – Specifies the configured limit will be applied to a port for which the intended usage is unknown. Default values: limit1 = 0, limit2 = 0 (disabled). • If no port classification type is specified, the limit is applied to all classifications.	**set flow limit** {**limit1** *limit* | **limit2** *limit*} [*class-index* | **userport** | **serverport** | **aggregateduser** | **interswitchlink** | **unspecified**]

Procedure 50-1 Configuring FST (continued)

Step	Task	Command(s)
2.	Add the low- and high-limit action to be taken for the specified classification to the current list of actions. • **action1** - The action associated with the low-limit option, to which the specified action is applied. • **action2** - The action associated with the high-limit option, to which the specified action is applied. • **notify** - Specifies that an SNMP trap notification will be sent for this action. • **drop** - Specifies that flow setup requests and packets associated with flows in excess of configured limits should be dropped for this action. • **disable** - Specifies that the interface should be disabled for this action. • *class-index* – Specifies a numeric value for the class user classification type to assign to this action. • If no action is specified then the default precedence of disable, drop, and notify is applied. • If a port classification is specified, the configured action is added to that port classification list. The actual action applied depends upon port classification precedence for the list. See Step 1 of this procedure for port classification definitions. • If no port classification is specified, the specified action is applied to all port classifications.	**set flowlimit** {**action1** | **action2**} [**notify** | **drop** | **disable**] [*class-index* | **userport** | **serverport** | **aggregateduser** | **interswitchlink** | **unspecified**]
3.	Set the ports to be used by the specified port classification. • See step 1 on page 50-5 for port classification definitions. If no port-string is specified, the specified port classification is applied to all ports.	**set flowlimit port class** {*class-index* | **userport** | **serverport** | **aggregateduser** | **interswitchlink** | **unspecified**} [*port-string*]
4.	Optionally, enable or disable FST on the specified port or all ports. • *port-string* - Specifies the port to which FST is enabled. If no port-string is specified, all ports are enabled for FST.	**set flowlimit port** {**enable | disable**} [*port-string*]
5.	Optionally enable or disable SNMP trap notifications globally on the device. Configured notify port actions will not occur until notification is globally enabled on the device. • *interval* - Specifies the number of seconds to wait before generating another notification of the same type for the same interface.	**set flowlimit notification** {**enable** | **disable**} | *interval*}

Procedure 50-1 Configuring FST (continued)

Step	Task	Command(s)
6.	Optionally enable or disable port shutdown globally on the device. Configured disable-port actions will not occur until port shutdown is globally enabled on the device.	**set flowlimit shutdown** {**enable** | **disable**}
7.	Enable FST on the device.	**set flowlimit enable**
8.	Optionally set to the operational state an administratively flowlimit disabled port. • *port-string* - Specifies the port to be manually set to the operational state. If no port-string is specified, all ports are set to the operational state.	**set flowlimit port status operational** *port-string*

Table 50-2 describes how to manage link aggregation.

Table 50-2 Managing FST

Task	Command
Clear the specified limit configuration for the specified port classification or for all port classifications. • **limit1** - The low-limit option to be cleared. • **limit2** - The high-limit option to be cleared. • *class-index* – Specifies a numeric value for the class user classification type to assign to this action. • **userport** - Clears the user port classification. • **serverport** - Clears the server port classification. • **aggregateduser** - Clears the multi-user port classification. • **interswitchlink** - Clears the ISL port classification. • **unspecified** - Clears the unspecified port classification. • If no port classification is specified, the specified limit is cleared for all port classifications.	**clear flowlimit** {**limit1** | **limit2**} [*class-index* | **userport** | **serverport** | **aggregateduser** | **interswitchlink** | **unspecified**]

Table 50-2 Managing FST (continued)

Task	Command
Clear the specified action configured for the specified port classification or for all port classifications. • **action1** - The low-limit action option to be cleared. • **action2** - The high-limit action option to be cleared. • **userport** - Clears the user port classification. • *class-index* – Specifies a numeric value for the class user classification type to assign to this action. • **serverport** - Clears the specified action for the server port classification. • **aggregateduser** - Clears the specified action for the multi-user port classification. • **interswitchlink** - Clears the specified action for the ISL port classification. • **unspecified** - Clears the specified action for the unspecified port classification. • If no port classification is specified, the specified action is cleared for all port classifications.	**clear flowlimit** {**action1** | **action2**} [**notify**] [**drop**] [**disable**] [*class-index* | **userport** | **serverport** | **aggregateduser** | **interswitchlink** | **unspecified**]
Clear the port classification for the specified port or for all ports. The port classification is reset to unspecified (the default). • *port-string* - Specifies the port for which to clear the port classification. If no port-string is specified, the port classification is cleared on all ports.	**clear flowlimit port class** [*port-string*]
Clear the flowlimit notification interval to the default value.	**clear flowlimit notification interval**
Clear all FST statistics associated with one or more ports. • *port-string* - Specifies the port for which to clear the show display statistics. If no port-string is specified, the statistics are cleared on all ports.	**clear flowlimit stats** [*port-string*]

Table 50-3 describes how to display link aggregation information and statistics.

Table 50-3 Displaying FST Information and Statistics

Task	Command
Display FST port configuration for one or more ports. *port-string* - Specifies the port for the display of port configuration. If no port-string is specified, configuration is displayed for all ports.	**show flowlimit port** [*port-string*]
Display FST statistics for one or more ports. • *port-string* - Specifies the port for the display of FST statistics. If no port-string is specified, statistics are displayed for all ports.	**show flowlimit stats** [*port-string*]
Display FST port classification configuration. If a port classification is not specified, configuration for all port classifications is displayed.	**show flowlimit class** [*class-index* | **userport** | **serverport** | **aggregateduser** | **interswitchlink** | **unspecified**]

Flow Setup Throttling Configuration Example

The FST configuration example presented here will provide a single port setup example for each port classification type. The baseline has been determined for each port as described in section "Determining a Port Classification Flow Baseline" on page 50-2. To determine the low-limit, the baseline is increased by 15%. To determine the high-limit, the baseline is increased by 60%.

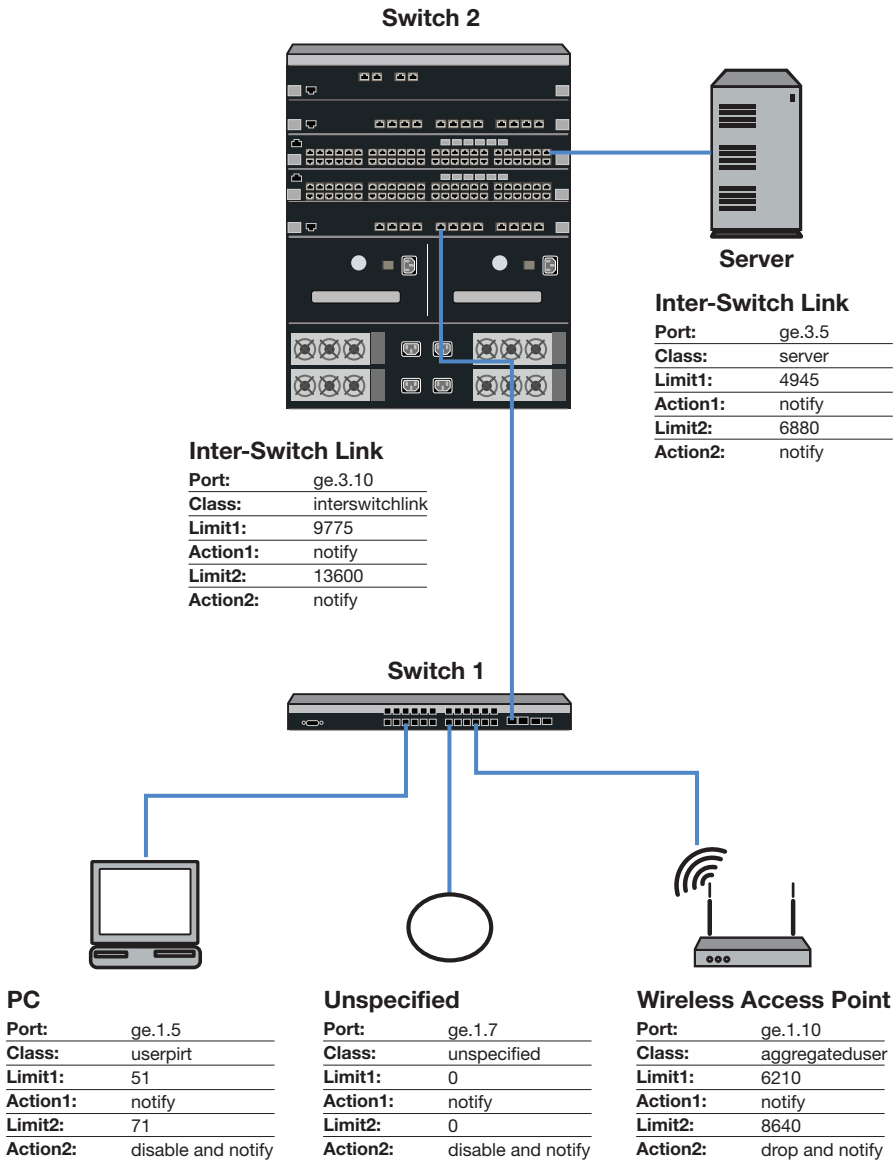
All limit1 actions will be configured for notification only. Limit2 actions are:

- The PC user: disable the port and send notification
- The wireless access point: drop excess packets associated with flows above the limit and send notification
- The unspecified port connection: disable interface and send notification
- The server port, ISL, and unspecified port connections: send notification only

The configuration components used in this example are two S-Series chassis, a PC, a wireless access point, and a server.

See Figure 50-1 on page 50-10 for an overview of this FST configuration example.

Figure 50-1 FST Configuration Example Overview



The configuration example assumes the default action configuration list of notify only for action1 and disable and notify for action2. Therefore:

- There is no need to make any configuration changes for action1 since action1 is always set to notify and that is the default.
- For action2, when either notification or disable are configured actions, there is no need to set these actions. For notification only actions, disable will be cleared. When drop is the configured action, drop is added and disable is cleared.

Switch 1 Configuration

The switch 1 chassis has ports with a single PC, a wireless access point, and an unspecified device.

Single User PC Configuration

The single user PC port was determined to have a flow baseline of 44 flows and is configured for:

- Port name and classification: ge.1.5, userport
- Limit1 and action1: 51, notification only (default)
- Limit2 and action2: 71, disable interface and notification (default)

```
S1(rw)->set flowlimit port class userport ge.1.5
S1(rw)->set flow limit1 51 userport
S1(rw)->set flow limit2 71 userport
S1(rw)->set flowlimit port enable ge.1.5
```

Wireless Access Point Configuration

The wireless access point was determined to have a flow baseline of 5400 flows. Because disable is the default action, you must clear the disable option for action2 before adding the drop action. The wireless access point is configured for:

- Port name and classification: ge.1.10, aggregateduser
- Limit1 and action1: 6210, notification only (default)
- Limit2 and action2: 8640, drop and notification

```
S1(rw)->set flowlimit port class aggregateduser ge.1.10
S1(rw)->set flow limit1 6210 aggregateduser
S1(rw)->set flow limit2 8640 aggregateduser
S1(rw)->clear flowlimit action2 disable aggregateduser
S1(rw)->set flowlimit action2 drop aggregateduser
S1(rw)->set flowlimit port enable ge.1.10
```

Unspecified Port Configuration

The unspecified port by definition has an undetermined baseline and is configured for:

- Port name and classification: ge.1.7, unspecified
- Limit1 and action1: 0 (default), notification only (default)
- Limit2 and action2: 0 (default), disable and notification (default)

```
S1(rw)->set flowlimit port class unspecified ge.1.7
S1(rw)->set flowlimit port enable ge.1.7
```

Switch 1 Global Configuration

Once the port classifications are associated with flow limits and actions, the following global configuration occurs:

- Notification is enabled on the device by default with an interval of 120 seconds
- Enable port shutdown on the switch 1 to globally allow PC and unspecified port action2 actions to occur
- Enable FST on the switch 1

```
S1(rw)->set flowlimit shutdown enable
S1(rw)->set flowlimit enable
```

Switch 2 Chassis Configuration

Server Configuration

The server port was determined to have a flow baseline of 4300 flows, and is configured for:

- Port name and classification: ge.3.5, serverport
- Limit1 and action1: 4945, notification only (default)
- Limit2 and action2: 6880, notification only

```
S2(rw)->set flowlimit port class serverport ge.3.5
S2(rw)->set flow limit1 4945 serverport
S2(rw)->set flow limit2 6880 serverport
S2(rw)->clear flowlimit action2 disable serverport
S2(rw)->set flowlimit port enable ge.3.5
```

Inter-Switch Link Configuration

The inter-switch link was determined to have a flow baseline of 8500 flows, and is configured for:

- Port name and classification: ge.3.10, interswitchlink
- Limit1 and action1: 9775, notification only (default)
- Limit2 and action2: 13600, notification only

```
S2(rw)->set flowlimit port class interswitchlink ge.3.10
S2(rw)->set flow limit1 9775 interswitchlink
S2(rw)->set flow limit2 13600 interswitchlink
S2(rw)->clear flowlimit action2 disable interswitchlink
S2(rw)->set flowlimit port enable ge.3.10
```

Switch 2 Global Configuration

Once the port classifications are associated with flow limits and actions, the following global configuration occurs:

- Notification is enabled on the device by default with an interval of 120 seconds
- Port shutdown is disabled by default. Since there is no disable action associated with a flowlimit on the N5, do not enable port shutdown on the this device.
- Enable FST on the switch 2

```
S2(rw)->set flowlimit enable
```

Terms and Definitions

Table 50-4 lists terms and definitions used in this link aggregation configuration discussion.

Table 50-4 Flow Setup Throttling Terms and Definitions

Term	Definition
action	The FST behavior that will occur when a limit threshold is exceeded for an associated port classification. Possible FST actions are: disable, drop, and notification.

Table 50-4 Flow Setup Throttling Terms and Definitions (continued)

Term	Definition
disable interface	An action that will be applied when an associated limit threshold for this ports configured port classification is reached. The disable interface action operationally disables the interface by placing the interface in a down state. The interface remains in the down state until the associated FST interface status is manually set to operational, the FST feature is disabled, or the device is reset.
drop	An action that will be applied when an associated limit threshold for this ports configured port classification is reached. The drop action drops any current or new flows that are in excess of the associated limit threshold.
Flow Setup Throttling (FST)	A proactive feature designed to mitigate zero-day threats and Denial of Service (DoS) attacks by defining ports by their placement in the network and setting low- and high-limit flow thresholds that trigger configured notification or flowlimiting actions.
Inter-Switch Link (ISL)	A high speed link connecting switches and routers.
limit threshold	Specifies the number of flows for the associated port classification that must be reached to trigger a configured FST action.
notification	An action that will be applied when an associated limit threshold for this ports configured port classification is reached. The notification action sends out an SNMP trap notification of the exceeded threshold. If the flowlimit threshold is exceeded, a single notification is sent out. The notification action is reset when the number of flows drops below the flowlimit threshold.
notification interval	A configured interval that throttles the sending of FST notifications by assuring that the configured period in seconds has expired before the sending of another notification.
operational state	An FST interface state that indicates the interface is fully FST operational. A down interface can be manually reset to operational status.
port classification	Provides for the configuring of separate limits and actions to different ports based upon the position of the port in the network or a numeric user defined classification. Configurable port types are: user defined (numeric value), single user, multiple user, server, ISL, and unspecified.
precedence	The order in which actions will be taken from highest precedence to lowest, when multiple actions are configured. Default precedence is disable and drop. If notification is configured, notification is always sent after any other configured action and takes into account that action in the information provided.

51

Route-Map Manager Configuration

This document describes the route-map manager feature and its configuration on Extreme Networks S-Series devices.

Using Route-Map Manager in Your Network

The route-map manager supports four distinct types of route-maps:

- Redistribution route-maps provide for the filtering of routes redistributed from one routing domain to another via the OSPF protocol
- Policy based route-maps filter learned routes and support the calculation of the next hop forwarding decisions in a policy based routing context
- Filter route-maps provide for the denial of routes into the OSPF route table
- BGP route-maps provide for the permit and denial of BGP packets

A named route-map consists of a set of permit or deny entries. Entries are sequenced by unique sequence numbers per named route-map. A route-map can contain multiple route-map sequences. Route-map entries are not unlike the permit and deny statements in an ACL with one very important exception: unlike the ACL, all route-map entries must be successful for this route-map's action to occur.

Each route-map sequence may contain one or more match and set clauses. A match clause contains the criteria that determines whether the permit or deny action for this entry should be taken. All route-map entries for a given sequence must be successful for a route-map action to occur. If multiple sequences are configured, the first one that matches all entries will "pass" and return the set actions for that sequence. If a sequence does not pass, the next sequence is processed until a sequence in which all entries match is found. If no entries match for all sequences, then the route-map is not used.

A set clause defines the action for this route-map. Depending on the route-map type and permit/deny setting of the route-map sequence, zero or more set clauses are supported per route-map sequence.

Policy Based Route-Maps

For policy based route-maps, if a match clause is configured, a match of the packet's source IP address against the contents of the specified ACL is required. A set entry specifies up to 5 next hop IP addresses for the forwarding of this packet. Multiple set clauses can be configured.

Policy based route-maps must be associated with an interface before route-mapping occurs. When assigning a route-map to an interface, the next hop load-policy behavior, which configures the algorithm used to select the next hop, and prioritization, which determines whether the priority based or routing table next hop is used, or whether the packet is dropped.

Default next hops can be configured and are only used when:

- No next hop configuration exists or the configured next hop IP addresses are not available
- The destination IP lookup results in the default route being returned

If both criteria are true, the next hop will be chosen from the default-next hop IP address list, using the configured load-policy setting.

If the next hop of a policy IP address match belongs to a different VRF, you can set the next hop VRF to perform the route lookup.

The route-map probe feature provides for the configuration of an ICMP probe to monitor next hops.

Redistribution Route-Maps

For redistribution route-maps, if a match clause is configured, a match of the packet source IP address against either a specified VLAN or the contents of one or more specified ALCs is required. A configured set entry specifies a route tag, metric, metric increment or decrement, or metric type to be used for redistribution by the ACLs matched in this route-map.

Redistribution route-maps, with a set entry specifying a route tag, must be assigned to the **redistribute** command within the OSPF router configuration command mode, for redistribution based upon this route-map to occur.

OSPF Filter Route-Maps

For OSPF filter route-maps, if a match clause is configured, a match on a deny route-map will deny the matched route from being installed into the OSPF route table based upon IP network address, next hop, source router-ID, outbound interface, OSPF tag, metric cost, or route-type.

OSPF filter route-maps must be assigned to the **distribution-list route-map in** command within OSPF configuration command mode for OSPF route table filtering to occur.

BGP Route-Maps

For BGP permit route-maps, all match clauses within a sequence must match for set clauses to be performed. For BGP deny route-maps, all match clauses within a sequence must match for the packet to be dropped. There is an exception to the all match clauses rule: in the case of multiple match prefix entries, only a single prefix entry needs to match. BGP route-maps support match clauses for:

- Address Family Indicator (AFI) and Subsequent Address Family Indicator (SAFI) attributes
- AS-Path attribute
- Community name
- Extended-community name
- Prefix list
- Multi-Exit Discriminator (MED)

- Autonomous System (AS)

BGP route-maps support set clauses for:

- Autonomous System (AS)
- Maximum length of the AS path attribute
- Community name
- Extended community attributes:
  - IP route target
  - AS and 4-octet AS route target
  - IP site of origin
  - AS and 4-octet AS site of origin
  - OSPF domain and router ID
  - OSPF route type
- Local preference
- Multi-Exit Discriminator (MED)
- IP next hop
- Origin
- Local Outbound Rate Filtering (ORF) association
- Weight
- Flap table

Implementing Route-Maps

Implementing a Policy Based Route-Map

To implement a policy based route-map:

- Create a policy based route-map and one or more entries for this route map
- For each sequence in the route-map, optionally configure match clauses to filter the packet based upon the specification of up to five ACLs per match clause
- For each sequence in the route-map, optionally configure a set clause specifying up to 5 next hops or default next hops per command line (system maximum of 128)
- Optionally configure the route-map probe feature to monitor each specified next hop in the route-map
  - If the next hop of a policy IP address match belongs to a different VRF, set the next hop VRF to perform the route lookup
- Assign the configured route-map to the interface for which policy-based routing is to be performed (a route-map can be assigned to multiple interfaces)
- Optionally, change the policy priority settings for this interface
- Optionally, change the load-policy settings for this interface

Implementing a Redistribution Route-Map

To implement a redistribution route-map:

- Create a redistribution route-map and one or more entries for this route-map
- For each sequence in the route-map, optionally configure match clauses to filter the packet source IP address based upon the specification of up to five ACLs per match clause
- For each sequence in the route-map, optionally configure match clauses to filter the packet source IP address based upon a specified interface
- For each sequence in the route-map, optionally configure match clauses to filter the packet based upon route cost or a route cost range.
- For each sequence in the route-map, optionally configure a set clause containing an OSPF route tag or range of route tags for this route-map
- In router configuration command mode, assign the route-map to the redistribute feature

Implementing an OSPF Filter Route-Map

To implement an OSPF filter route-map:

- Create a filter route-map and one or more entries for this route-map
- For each sequence in the route-map, optionally configure match clauses to filter routes for this OSPF route table
- In OSPF router configuration command mode, apply the route map filter using the distribute-list route-map in command

Implementing a BGP Route-Map

To implement a BGP route-map:

- Create a BGP route-map and one or more entries for this route-map.
- For each sequence in the route-map, configure match clauses. See "BGP Route-Maps" on page 51-2 for a listing of supported match clauses.
- For each sequence in the route-map, optionally configure set clauses. See "BGP Route-Maps" on page 51-2 for a listing of supported set clauses.

Route-Map Manager Overview

This section provides an overview of route-map manager configuration.

Creating a Route-Map

When creating a route-map, specify:

- Whether it is a policy based, redistribution, or filter route-map
- The name of the route-map using up to 32 alpha-numeric characters
- Whether this sequence is a permit or deny (defaults to permit)
- A sequence number for this entry (defaults to 10)

Currently, up to 100 each of filter, redistribution, BGP, and policy route-maps are permitted.

Multiple sequences can be input for a single named route-map. Configuring a route-map sequence places you in route-map configuration command mode for the configuration of route-map match and set clauses. The system-wide maximum number of both match and set route-map clauses is 1000.

Policy route-maps must have at least one IP address match clause and at least one next hop or default next hop clause. An ACL that has not yet been created can be specified in an IP address match clause. If a route-map is applied to an interface, any ACLs that have not been created will be ignored. Policy based route-maps must be assigned to an interface using the **ip policy route-map** command in interface configuration mode.

Redistribution route-maps must be associated with the redistribution of OSPF routes within the OSPF routing protocol using the **redistribute** command in OSPF router configuration command mode.

Filter route-maps must be associated with the filtering of OSPF routes from the OSPF route table (FIB) using the **distribute-list route-map in** command.

Use the **route-map policy** command in configuration command mode to create a policy based route-map.

Use the **route-map redistribution** command in configuration command mode to create a redistribution route-map.

Use the **route-map filter** command in configuration command mode to create an OSPF filter route-map.

Configuring Match and Set Clauses

Upon entering a route-map sequence, you are placed in route-map configuration command mode. Match and set clauses are configured in route-map configuration command mode.

A route-map sequence's match clause specifies the criterion that determines whether the action for this route-map will occur. The following types of match clauses are supported:

Redistribution Match Clauses

- A match clause that matches packets egressing on this interface with the statements in up to five specified ACLs. Multiple clauses may be used. At least one of the ACLs in each clause must match the packet in order for the route-map to redirect the packet. The only limit on the number of ACLs supported is the system limit of 1000 route-map clauses. Use the **match ip address** command in redistribution route-map configuration command mode to specify up to five ACLs for this match clause.
- An interface match clause that matches the source IP address of a packet egressing on this interface against a specified VLAN interface. Use the **match interface** command in redistribution route-map configuration command mode to specify a VLAN interface for this match clause.
- A metric match clause that matches the specified or a range of cost against the route cost specified in the packet. Use the **match metric** command in redistribution route-map configuration command mode to specify the metric cost for this match clause.
- An OSPF tag match clause that matches the specified OSPF tag or range of tags against the OSPF tag ID specified in the packet. Use the match tag command is redistribution route-map configuration command mode to specify the OSPF tag ID for this match clause.

Policy Match Clauses

An IP address match clause that matches the source IP address of a packet egressing on this interface with the statements up to five specified ACLs. The IP address match clause can be entered for both a policy based route-map and a redistribution route-map.

Use the **match ip address** command in policy-based route-map configuration command mode to specify up to five ACLs to be associated with this match clause. Multiple clauses may be used. At least one of the ACLs in each clause must match the packet in order for the route-map to redirect the packet.

OSPF Filter Match Clauses

- An IP match clause that matches a route network address, next hop or source router ID against the route to be entered into the OSPF routing table. Use the **match ip** command in filter-based route-map configuration command mode to specify up to five ACLs to be associated with this match clause. Multiple clauses may be used. At least one of the ACLs in each clause must match the packet in order for the route-map to redirect the packet.
- An interface match clause that matches the outgoing interface of the route to be installed in the OSPF routing table. Use the **match interface** command in filter-based route-map configuration command mode to specify an outgoing interface for this match clause.
- A OSPF tag match clause that matches the OSPF tag for this route. Use the **match tag** command in filter-based route-map configuration command mode to specify an OSPF tag or range of tags for this match clause.
- A metric match clause that matches the OSPF cost metric for this route. Use the **match metric** command in filter-based route-map configuration command mode to specify an OSPF route cost metric or range of cost metrics for this match clause.
- A route-type match clause that matches the internal or external route type for this route. Use the **match route-type** command in filter-based route-map configuration command mode to specify an OSPF route-type for this match clause.

There can be multiple match clauses associated with a single route-map sequence.

A route-map sequence's set clause determines the action the route-map will take when a successful match for this sequence occurs. The action configurable for a set clause depends upon the route-map type. For a policy based route-map, the set clause specifies one or more next hops for this route. For the redistribution route-map, the set clause specifies an OSPF route tag for this route.

Policy Based Set Clauses

Policy based set clauses determine the next hop for this route if a match clause for this route-map sequence is successful. If a nexthop clause is specified, any default next hop clauses are ignored unless all next hops are unavailable and the destination IP lookup results in the default route being returned.

Use the **set next-hop** command in policy based route-map configuration mode to specify the next hop(s) available for this route-maps action.

Use the **set default-next-hop** command in policy based route-map configuration mode to specify the default next hop(s) available for this route-maps action.

Use the **set vrf** command in policy based route-map configuration mode to set the next hop VRF to perform the route lookup, if the next hop of a policy IP address match belongs to a different VRF.

Route-Map Probe

The route-map manager supports the assigning of an ICMP probe to monitor a next hop IP address. Tracked object manager uses the route-map facility to monitor the IP address, but you do not assign the ICMP probe to a specific route-map. If a next hop IP address is declared down, it is removed from the next hop selection process for all route-maps specifying this address as a next hop, until it is declared up again. The assigned ICMP probe will ping port 0 of the specified IPv4 or IPv6 address.

A route-map probe entry is configurable for each configured next hop address. Currently a combination of up to 128 standard or default next hop addresses are configurable on a system. If the same next hop is referenced in multiple route-maps, only a single route-map probe instance is created.

See Chapter 13, **Tracked Object Manager Configuration** for tracked object manager details.

Use the **route-map probe** command in router configuration mode to assign an ICMP probe to monitor the specified next hop IP address. A predefined policy based routing ICMP probe named **$pbr_default** can be used, or you can create a probe, using the **probe** command. Predefined ICMP probes can not be specified by name. Use the **default** keyword when configuring the default route-map probe.

This example shows how to create the ICMP probe **ICMP-PBR** and assign it to a route-map probe to monitor next hop IP addresses **101.10.1.252** and **2000::1301:0:21f:45ff:fe4d:8722**. The fail detection count is set to **5** attempts, and the fail detection interval is set to **5** seconds. The two assigned sessions are displayed:

```
S Chassis(su-config)->probe ICMP-PBR icmp
S Chassis(su-config-probe)->faildetect count 5 interval 5
S Chassis(su-config-probe)->inservice
S Chassis(su-config-probe)->exit
S Chassis(su-config)->route-map probe 101.10.1.252 probe-name ICMP-PBR
S Chassis(su-config)->route-map probe 2000::1301:0:21f:45ff:fe4d:8722 probe-name
ICMP-PBR
S Chassis(su-config)->show probe sessions

Client Codes: P-policy based routing, S-SLB, V-VRRP, W-TWCB
              T-tracked object probe
...
Probe: ICMP-PBR, icmp
IP Address                      Port  Status    StChngs Last Change   Clients
------------------------------- ----- --------- ------- ------------- -------
101.10.1.252                    0     Up        1            0h0m30s  P
2000::1301:0:21f:45ff:fe4d:8722 0     Up        1            0h0m40s  P
Displayed 2 sessions
...
S Chassis(su-config)->
```

This example shows how to create the ICMP probe **ICMP-PBR** and assign it to a route-map probe to monitor next hop IP address **101.10.1.252**. The fail detection count is set to **5** attempts, and the fail detection interval is set to **5** seconds. The assigned session is displayed:

```
S Chassis(su-config)->probe ICMP-PBR icmp
S Chassis(su-config-probe)->faildetect count 5 interval 5
S Chassis(su-config-probe)->inservice
```

```
S Chassis(su-config-probe)->exit
S Chassis(su-config)->route-map probe 101.10.1.252 probe-name ICMP-PBR
S Chassis(su-config)->show probe sessions

Client Codes: P-policy based routing, S-SLB, V-VRRP, W-TWCB
              T-tracked object probe
...
Probe: ICMP-PBR, icmp
IP Address                      Port  Status    StChngs Last Change   Clients
------------------------------- ----- --------- ------- ------------- -------
101.10.1.252                    0     Up        1            0h0m30s  P
Displayed 1 sessions
...
S Chassis(su-config)->
```

The Redistribution Match Clauses

The redistribution route-map entry allows the specifying of both IP address and interface match clauses. Up to five ACLs can be configured in an IP address match clause. A single interface can be configured for an interface match clause.

The Redistribution Set Clause

The redistribution set clause determines the OSPF route tag, metric cost, along with the ability to increment or decrement the current metric cost, and metric type for this route if a match clause for this route-map entry is successful.

Use the **set tag** command in redistribution route-map configuration command mode to specify the OSPF route to be used for redistributing non-OSPF routes that match for this route-map.

Use the **set metric** command in redistribution route-map configuration command mode to specify the metric cost of routes that match for this route-map. Use the **set metric increment** command to increase the current metric cost or **set metric decrement** command to decrement the current metric cost of routes that match for this route-map.

OSPF route tag is a 32-bit numeric value that is attached to redistributed routes into OSPF. The route tag is not used by OSPF, but can be used by other routers for making policy decisions. OSPF route tags are displayed in the **show ip ospf database external** command. See the *Extreme Networks S-Series CLI Reference* for command details.

Assigning a Policy Route-Map to an Interface

Route-map filtering does not occur until the configured route-map is assigned to an interface. Once assigned to an interface the route-map is operational.

Next hop load-policy and priority can also be configured at the interface level. Load-policy determines the load balancing algorithm that will be used in the next hop selection process. The three configurable options are:

- **first-available** - The first available next hop from the list of next hops is used (default)
- **round-robin** - The selection process moves through the list in a sequential circular fashion repeating the sequence when it comes to the end of the list
- **ip-hash** - The selection is based on an exclusive-or (XOR) hash of the IP source address, IP destination address, or both

Priority allows the user to specify whether the route-map lookup or the route table lookup will have priority in the next hop selection process as follows:

- **only** - Uses the priority based routing next hop and drops the packet if the priority based routing next hop is not available
- **first** - Uses priority based routing next hop or uses the route table next hop if the priority based next hop is not available
- **last** - Uses the route table if the route exists there, otherwise the priority based routing next hop is used

Use the **ip policy route-map** command in interface configuration command mode to assign a route-map to an interface.

Use the **ip policy load-policy** command in interface configuration command mode to determine the load balancing algorithm that will be used in the next hop selection process.

Use the **ip policy priority** command in interface configuration command mode to specify whether the route-map lookup or route table lookup will determine the next hop for this route.

Configuring Route-Map Manager

This section provides details for the configuration of route-map manager on the S-Series products.

Table 51-1 lists route-map manager parameters and their default values.

Table 51-1 Default Route-Map Manager Parameters

Parameter	Description	Default Value
entry	A route-map's sequenced container for match and set clauses specifying a permit or deny behavior.	permit
sequence number	A numeric value specifying the ordering of route-map entries.	10
next hop priority	Specifies whether the priority based lookup or the routing table lookup will be used to select the next hop.	priority based lookup, then route table lookup
next hop load-policy	Specifies the algorithm that will be used to select the next hop.	first-available

Procedure 51-1 describes how to configure a policy based route-map.

Procedure 51-1 Configuring a Policy Based Route-Map

Step	Task	Command(s)
1.	In configuration command mode, create a policy based route map, optionally specifying whether this entry is a permit or deny, and the sequence number for this entry. This command provides access to policy based route-map configuration command mode. Use this command to create multiple entries if required.	**route-map policy** *name* [**permit** | **deny**] [*sequence-number*]

Procedure 51-1 Configuring a Policy Based Route-Map (continued)

Step	Task	Command(s)
2.	In policy based route-map configuration command mode, specify one or more match clauses for this route-map, specifying up to five ACLs that will be matched against the packet source IP address. Though not necessary, it is recommended that all ACLs be configured before assigning them to an IP address match clause.	**match ip address** *access-list*
3.	In policy based route-map configuration command mode, specify a set clause containing up to five next hop IP addresses for this route-map. One or more of these commands can be specified.	**set next-hop** {*next-hop1*} [*next-hop2 .... next-hop5*]
4.	In policy based route-map configuration command mode, specify a set clause containing up to five default next hop IP addresses for this route-map to be used when next hops are not specifically configured or available using the **set next-hop** command. One or more of these commands can be specified.	**set default-next-hop** {*next-hop1*} [*next-hop2....next-hop5*]
5.	In policy based route-map configuration command mode, specify the VRF that will perform the next hop lookup, when the next hop of a policy IP address match belongs to a different VRF.	**set vrf** *vrf-name*
6.	Optionally, in configuration command mode, configure the route-map probe feature to monitor the configured next hops.	**route-map probe** *ip-address* **probe-name** {*name* | **default**}
7.	In interface configuration command mode, prioritize the priority based lookup to route table lookup behavior for this interface.	**ip policy priority** {[**only**] [**first**] [**last**]}
8.	In interface configuration command mode, configure the load policy for this route-map's next hop selection method.	**ip policy load-policy** {**first-available** | **round-robin** | **ip-hash** {**source** | **destination** | **both**}}
9.	In interface configuration command mode, assign the configured route-map to the interface.	**ip policy route-map** *name*

Procedure 51-2 describes how to configure a redistribution route-map.

Procedure 51-2 Configuring a Redistribution Route-Map

Step	Task	Command(s)
1.	In configuration command mode, create a redistribution route map, optionally specifying whether this entry is a permit or deny, and the sequence number for this entry. This command provides access to redistribution route-map configuration command mode. Use this command to create multiple entries if required.	**route-map redistribution** *name* [**permit** | **deny**] [*sequence-number*]

Procedure 51-2 Configuring a Redistribution Route-Map (continued)

Step	Task	Command(s)
2.	In redistribution route-map configuration command mode, specify one or more match clauses for this route-map, specifying up to five ACLs that will be matched against the packet source IP address.	**match ip address** *access-list*
3.	In redistribution route-map configuration command mode, specify a VLAN interface to match a packet source IP address against.	**match interface** {**vlan** *vlan* | *string*}
4.	In redistribution route-map configuration command mode, specify one or a range of metric costs that will be matched against the packet metric cost.	**match metric** {*cost* | **range** *min-cost max-cost*}
5.	In redistribution route-map configuration command mode, specify an OSPF tag ID or range of IDs that will be matched against the packet OSPF tag ID.	**match tag** {*tag-id* | **range** *min-tag-id max-tag-id*}
6.	In redistribution route-map configuration command mode, specify a set clause containing an OSPF route tag for this route-map.	**set tag** *tag*
7.	In redistribution route-map configuration command mode, specify a set clause containing a metric cost for this route-map. A single metric cost can be configured per sequence.	**set metric** *cost*
8.	In redistribution route-map configuration command mode, specify a set clause containing the amount to decrement the current metric cost for this route-map. A single metric decrement can be configured per sequence.	**set metric decrement** *cost*
9.	In redistribution route-map configuration command mode, specify a set clause containing the amount to increment the current metric cost for this route-map. A single metric increment can be configured per sequence.	**set metric increment** *cost*
10.	In redistribution route-map configuration command mode, specify a set clause containing the OSPF metric type to be used when redistributing a source packet matched by this route-map. A single metric type can be configured per sequence.	**set metric-type** {**type-1** | **type-2**}
11.	In OSPF router configuration mode, assign this route-map to the redistribute command.	**redistribute** {**rip** | **static** | **connected**} [**route-map** *name*] [**metric** *metric value*] [**metric-type** *type-value*] [**tag** *tag*]

Procedure 51-3 describes how to configure an OSPF filter route-map.

Procedure 51-3 Configuring a Filter Route-Map

Step	Task	Command(s)
1.	In configuration command mode, create an OSPF filter route map, optionally specifying whether this entry is a permit or deny, and the sequence number for this entry. This command provides access to filter route-map configuration command mode. Use this command to create multiple entries if required.	**route-map filter** *name* [**permit** | **deny**] [*sequence-number*]
2.	In filter route-map configuration command mode, specify one or more IP network address, next hop, or source router ID match clauses for this route-map, specifying up to five ACLs that will be matched against specified IP type. • **address** - network address • **next-hop** - next hop • **route-source** - source router ID	**match ip** {**address** | **next-hop** | **route-source**} *access-list*
3.	In filter route-map configuration command mode, specify one or more outbound interface match clauses that will be matched against the route outbound interface.	**match interface** {*interface-name* | *alias*}
4.	In filter route-map configuration command mode, specify one or more OSPF tag match clauses that will be matched against the route OSPF tag or a range of OSPF tags.	**match tag** {*tag* | **range** *min-tag max-tag*}
5.	In filter route-map configuration command mode, specify one or more OSPF cost metric match clauses that will be matched against the route metric cost or a range of metric cost values.	**match metric** {*cost* | **range** *min-cost max-cost*}
6.	In filter route-map configuration command mode, specify one or more OSPF route type match clauses that will be matched against the route's route type. • **internal** - Internal route type • **external-t1** - External route type 1 • **external-t2** - External route type 2 • **nssa-external** - External NSSA route type	**match route-type** {**internal** | **external-t1** | **external-t2** | **nssa-external**}
7.	In OSPF router configuration command mode, apply the filter route-map to the OSPF distribution-list.	**distribute-list route-map** *name* **in**

Procedure 51-4 describes how to configure a BGP route-map.

Table 51-2 describes how to display route-map manager information. Display commands can be

Procedure 51-4 Configuring a BGP Route-Map

Step	Task	Command(s)
1.	In configuration command mode, create a BGP route map, optionally specifying whether this entry is a permit or deny, and the sequence number for this entry. This command provides access to BGP route-map configuration command mode. Use this command to create multiple entries if required.	**route-map bgp** *name* [**permit** | **deny**] [*sequence-number*]
2.	In BGP route-map configuration command mode, configure a match clause to match a packet against its IPv4 or IPv6 Address Family Indicator (AFI) attribute.	**match afi** {**ipv4** | **ipv6**}
3.	In BGP route-map configuration command mode, configure a match clause to match a packet against its Subsequent Address Family Indicator (SAFI) attribute, specifying whether the attribute is unicast or multicast.	**match safi** {**unicast** | **multicast**}
4.	In BGP route-map configuration command mode, configure a match clause to match a packet against its AS path attribute.	**match as-path** *as-path-string*
5.	In BGP route-map configuration command mode, configure a match clause to match a packet against the specified community name.	**match community** *name*
6.	In BGP route-map configuration command mode, configure a match clause to match a packet against the specified extended community.	**match extended-community** *name*
7.	In BGP route-map configuration command mode, configure a match clause to match a packet against the specified prefix list. Multiple prefix-list match entries are allowed.	**match prefix-list** *prefix-list*
8.	In BGP route-map configuration command mode, configure a match clause to match a packet against the specified MED value.	**match med** *value*
9.	In BGP route-map configuration command mode, configure a match clause to specify the number of times to prepend the AS number of this router to the AS path for this route map context.	**set as** *num*
10.	In BGP route-map configuration command mode, configure a match clause to set a maximum length of the AS path attribute allowed when all match clauses match for this route map.	**set as-path-limit** *limit*
11.	In BGP route-map configuration command mode, configure a match clause to set the community when all match clauses match for this route map.	**set community** {*as*:*community* | *defined-community*}

Procedure 51-4 Configuring a BGP Route-Map (continued)

Step	Task	Command(s)
12.	In BGP route-map configuration command mode, configure a match clause to specify an action for an extended community IP route target when all match clauses match for this route map.	**set extended-community ip-route-target** *set-value* {**remove-all** | **remove-specific** | **set-specific** | **remove-all-and-set**}
13.	In BGP route-map configuration command mode, configure a match clause to specify an action for an extended community AS route target when all match clauses match for this route map.	**set extended-community as-route-target** *set-value* {**remove-all** | **remove-specific** | **set-specific** | **remove-all-and-set**}
14.	In BGP route-map configuration command mode, configure a match clause to specify an action for an extended community IP site of origin when all match clauses match for this route map.	**set extended-community ip-site-of-origin** *set-value* {**remove-all** | **remove-specific** | **set-specific** | **remove-all-and-set**}
15.	In BGP route-map configuration command mode, configure a match clause to specify an action for an extended community AS site of origin when all match clauses match for this route map.	**set extended-community as-site-of-origin** *set-value* {**remove-all** | **remove-specific** | **set-specific** | **remove-all-and-set**}
16.	In BGP route-map configuration command mode, configure a match clause to specify an action for an extended community AS4 route target when all match clauses match for this route map.	**set extended-community as4-route-target** *set-value* {**remove-all** | **remove-specific** | **set-specific** | **remove-all-and-set**}
17.	In BGP route-map configuration command mode, configure a match clause to specify an action for an extended community AS4 site of origin when all match clauses match for this route map.	**set extended-community as4-site-of-origin** *set-value* {**remove-all** | **remove-specific** | **set-specific** | **remove-all-and-set**}
18.	In BGP route-map configuration command mode, configure a match clause to specify an action for an extended community OSPF domain ID when all match clauses match for this route map.	**set extended-community ospf-domain-id** *set-value* {**remove-all** | **remove-specific** | **set-specific** | **remove-all-and-set**}
19.	In BGP route-map configuration command mode, configure a match clause to specify an action for an extended community OSPF domain ID when all match clauses match for this route map.	**set extended-community ospf-router-id** *set-value* {**remove-all** | **remove-specific** | **set-specific** | **remove-all-and-set**}
20.	In BGP route-map configuration command mode, configure a match clause to specify an action for an extended community OSPF route type when all match clauses match for this route map.	**set extended-community ospf-route-type** *area* **route-type** *type* [**type2-metric**] {**remove-all** | **remove-specific** | **set-specific** | **remove-all-and-set**}
21.	In BGP route-map configuration command mode, configure a match clause to specify the local preference to be set when all match clauses in the route map match.	**set local-preference** *value*

Procedure 51-4 Configuring a BGP Route-Map (continued)

Step	Task	Command(s)
22.	In BGP route-map configuration command mode, configure a match clause to specify the MED to be set when all match clauses in the route map match.	**set med** *value*
23.	In BGP route-map configuration command mode, configure a match clause to specify the next hop IP address to be set when all match clauses in the route map match.	**set ip next-hop** *ip-address*
24.	In BGP route-map configuration command mode, configure a match clause to specify the origin code to be set when all match clauses in the route map match.	**set origin** *code*
25.	In BGP route-map configuration command mode, configure a match clause to set local ORF association when all match clauses in the route map match.	**set orf-association local**
26.	In BGP route-map configuration command mode, configure a match clause to specify the weight to be set when all match clauses in the route map match.	**set weight** *value*
27.	In BGP route-map configuration command mode, configure a match clause to specify the flap table to be set when all match clauses in the inbound route map match.	**set flap-table** *name*

entered in any command mode.

Table 51-2 Displaying Route-Map Manager Information and Statistics

Task	Command
To display configured route-maps:	**show route-map** [*name*] [**brief**] [**probe**]
To display the policy applied to a routing interface:	**show ip policy**

Route-Map Manager Configuration Examples

This section presents a route-map manager configuration examples for a policy based and a redistribution route-map.

Policy Based Route-Map Example

The following example:

- Creates a policy based route-map name **rmP1** that filters IP packets with source addresses on the **60.10.0.0** subnet destined for hosts **50.10.0.1-2**.
- Packets that pass this filter will be routed using one of three next hops: **30.10.0.10**, **30.10.0.20**, or **30.10.0.30**.
- The route-map probe feature is configured to monitor these three next hops for availability using the default policy based routing probe **$pbr_default**.

- The route-map is assigned to VLAN **110**.
- Policy priority is set such that only the policy route lookup will determine the route, and if not available, the packet will be dropped.
- The load-policy is set to round-robin.

```
S Chassis(rw)->configure
S Chassis(rw-config)->ip access-list extended 101
S Chassis(rw-cfg-ext-acl)->permit ip 60.10.0.0 0.0.255.255 host 50.10.0.1
S Chassis(rw-cfg-ext-acl)->permit ip 60.10.0.0 0.0.255.255 host 50.10.0.2
S Chassis(rw-cfg-ext-acl)->deny ip any any
S Chassis(rw-cfg-ext-acl)->show access-lists 101
Extended IP access list 101  (4 entries)
  1 permit ip  60.10.0.0  0.0.255.255  host 50.10.0.1
  2 permit ip  60.10.0.0  0.0.255.255  host 50.10.0.2
  3 deny   ip  any  any
  -- implicit deny all --
S Chassis(rw-cfg-ext-acl)->exit
S Chassis(rw-config)->route-map policy rmP1 permit 10
S Chassis(rw-config-route-map-pbr)->match ip address 101
S Chassis(rw-config-route-map-pbr)->set next-hop 30.10.0.10 30.10.0.20 30.10.0.30
S Chassis(rw-config-route-map-pbr)->exit
S Chassis(rw-config)->show route-map rmP1
 route-map policy rmP1 permit 10
  match ip address 101
  set next-hop 30.10.0.10 30.10.0.20 30.10.0.30
 Policy matches: 0 packets
S Chassis(rw-config)->route-map probe 30.10.0.10 default
S Chassis(rw-config)->route-map probe 30.10.0.20 default
S Chassis(rw-config)->route-map probe 30.10.0.30 default
S Chassis(rw-config)->interface vlan 110
S Chassis(rw-config-intf-vlan.0.110)->ip policy priority only
S Chassis(rw-config-intf-vlan.0.110)->ip policy load-policy round-robin
S Chassis(rw-config-intf-vlan.0.110)->ip policy route-map rmP1
S Chassis(rw-config-intf-vlan.0.110)->show ip policy
Interface    Route map                       Priority Load policy  Match count
-----------  ------------------------------- -------- ------------ ----------
vlan.0.110   rmP1                            Only     Round Robin  0
S Chassis(rw-config-intf-vlan.0.110)->exit
S Chassis(rw-config)->
```

Redistribution Route-Map Example

The following example:

- Creates a redistribution route-map named **rmR1** for the redistribution of RIP routes with a permit entry, sequence 10 that filters IP packets with source addresses on the 40.0.0.0 and 40.0.10.0 subnets

- Packets that pass the filter have the OSPF route tag set to 65432
- Redistribute in OSPF router 1 is assigned the rmR1 route-map

```
S Chassis(rw)->configure
S Chassis(rw-config)->ip access-list standard OSPF
S Chassis(rw-cfg-std-acl)->permit 40.0.0.0 0.0.0.255
S Chassis(rw-cfg-std-acl)->permit 40.0.10.0 0.0.0.255
S Chassis(rw-cfg-std-acl)->show access-lists OSPF
Standard IP access list OSPF  (3 entries)
  1 permit 40.0.0.0  0.0.0.255
  2 permit 40.0.10.0  0.0.0.255
  -- implicit deny all --
S Chassis(rw-cfg-std-acl)->exit
S Chassis(rw-config)->route-map redistribution rmR1 permit 10
S Chassis(rw-config-route-map)->match ip address OSPF
S Chassis(rw-config-route-map)->set tag 65432
S Chassis(rw-config-route-map)->exit
S Chassis(rw-config)->show route-map rmR1
 route-map redistribution rmR1 permit 10
  match ip address OSPF
  set tag 65432
S Chassis(rw-config)->router ospf 1
S Chassis(rw-config-ospf-1)->redistribute rip route-map rmR1
S Chassis(rw-config-ospf-1)->exit
S Chassis(rw-config)->
```

BGP Route-Map Example

The following BGP route-map example:

- Creates a BGP route-map named **bgprm1** as a permit entry
- Specifies that the packet prefix should match a prefix listed in prefix list **pfxlist1**
- Specifies that the packet AS path string should match **^20313.*13$**
- Specifies the setting of the IP next hop to **152.50.25.10** as the action if all match clauses match

```
S Chassis(su)->configure
S Chassis(su-config)->route-map bgp bgprm1 permit
S Chassis(su-config-route-map-bgp)->match prefix-list permit100
S Chassis(su-config-route-map-bgp)->match prefix-list pfxlist1
S Chassis(su-config-route-map-bgp)->match as-path ^20313.*13$
S Chassis(su-config-route-map-bgp)->set ip next-hop 152.50.25.10
S Chassis(su-config-route-map-bgp)->
```

Terms and Definitions

Table 51-3 lists terms and definitions used in this route-map manager configuration discussion.

Table 51-3 Route-Map Manager Terms and Definitions

Term	Definition
entry	A logical container within the named route-map that specifies a permit or deny behavior for the configured match and set clauses it contains.
filter route-map	A route filtering container that provides for the denial of routes into the OSPF route table.
load-policy	The ability to configure the algorithm that will be used for the next hop selection for this route-map.
match clause	A clause that specifies the criteria for filtering routes for a route-map.
route-map probe	A tracked object manager object of protocol type ICMP that tracks the availability of a next hop IP address, by actively pinging the address.
policy route-map	A route filtering container that permits or denies routes based upon an ACL entry match, optionally allowing for the specification of up to five next hops for routes that pass the filter.
priority	The ability to configure whether the priority route lookup or the route table lookup will determine the next hop for this route.
redistribution route-map	A route filtering container that permits or denies routes based upon an ACL entry match for purposes of redistribution over the OSPF protocol
set clause	A clause that specifies the action that will occur for routes matched by the route-map match clause.

52

Access Control List Configuration

This document provides the following information about configuring Layer 3 (both IPv4 and IPv6) and Layer 2 Access Control Lists (ACLs) on the Extreme Networks S-Series platforms.

Using Access Control Lists (ACLs) in Your Network

This section details two types of ACLs:

- **Layer 3 ACL**s (L3 ACL) which allow the configuration of permit and denial of IPv4 and IPv6 packet forwarding based upon IP address, protocol, port matching (depending upon the ACL type) and provides an all traffic option allowing ingress packet filtering on all traffic instead of just routed traffic.
- **Layer 2 ACL**s (L2 ACL) which allow the configuration of permit and denial packet restrictions based upon the MAC address, VLAN tag, Drop Eligibility Indicator (DEI), and Ethernet II type

The S-Series firmware supports configuration of both standard and extended L3 ACLs and L2 ACLs. Standard L3 ACLs allow the packet source IP address to be configured, while extended L3 ACLs allow both source and destination IP addresses, protocol and TCP or UDP port matching, as well as the optional specifying of a DSCP, ToS, or IP precedence value. L3 ACLs are also used to match addresses or traffic by client applications such as route map (for policy-based routing and route redistribution), NAT, and IP Directed Broadcast.

ACLs can be applied to VRF access groups to provide a more granular control of traffic between VRFs. One IPv4 and one IPv6 ACL inbound to each VRF and one IPv4 and one IPv6 ACL outbound from each VRF can be applied.

Implementing ACLs

To implement an ACL on your network:

- Create the L3 or L2 ACL
- Enter the rules and comments for the ACL:

  - For a L3 ACL the rules determine which packets will be forwarded or not forwarded on the routing interface this ACL will be applied to
  - For the L2 ACL the rules determine which packets will be restricted on the VLAN interface
- Optionally manage ACLs of the same type by:
  - Copying a preexisting ACL to a non-existing ACL
  - Appending a preexisting ACL to another preexisting ACL
  - Entering an ACL comment entry
  - Deleting an ACL rule entry
  - Inserting a new ACL rule entry into an ACL
  - Moving an ACL rule to a new location in an ACL
- Apply the L3 ACL to a routing interface or the L2 ACL to a VLAN interface
- Optionaly apply an IPv4 and IPv6 L3 ACL in both an inbound and outbound direction to a VRF

ACL Overview

This section describes ACL creation, rule entry, and application of the ACL to a routing or VLAN interface required to implement an ACL, as well as, the features available for managing ACL rules and displaying ACLs.

Note: An "implicit deny" is hard coded at the end of all ACLs. The implicit deny blocks anything not explicitly permitted within the ACL, including routing protocols and management connections.

L3 ACL Creation

There are two types of ACLs: standard and extended. The type of ACL you need depends exclusively upon the packet field(s) that will generate a hit for the rules specified in the ACL. For a standard ACL, only the source IP address is configurable. For an extended ACL, the protocol, source IP address, destination IP address, and in the case of the TCP or UDP protocols, matching source and destination ports are configurable.

There are two ways to identify the new ACL: a number or a name. The use of a number is for IPv4 ACLs only. Standard IPv4 ACL numbers range from **1** to **99**. Extended IPv4 ACL numbers range from **100** to **199**. Both IPv4 and IPv6 allow alphanumeric names that must start with an alpha character. A name may be quoted, as the quotes are stripped, but spaces are not supported in the quoted string. A name cannot be one of the **show access-lists** keywords **brief** or **applied**, or any prefix thereof such as ?br? or ?app?. Names can be up to 64 characters in length.

Once you have determined the appropriate ACL type, use the:

- **ip access-list standard** command to create an IPv4 standard access-list and **ipv6 access-list standard** command to create an IPv6 standard access-list
- **ip access-list extended** command to create an IPv4 extended access-list and **ipv6 access-list extended** command to create an IPv6 extended access-list

In each case, specifying the access-list number or name for the ACL.

An existing L3 ACL can be copied to a non-existing L3 ACL of the same IP type (IPv4 or IPv6). An existing L3 ACL can be appended to the end of another existing L3 ACL of the same IP type, but a standard L3 ACL may not be appended to an extended L3 ACL nor vice versa.

Upon creating the L3 ACL, you are placed in the access-list configuration command mode where you can enter rules or comment entries for this L3 ACL.

IPv4 ACL Creation Examples

The following example creates a standard IPv4 ACL with the access-list number **1** as its identifier:

```
S Chassis(rw-config)->ip access-list standard 1
S Chassis(rw-cfg-std-acl)->
```

The following example creates an extended IPv4 ACL with the access-list number **100** as its identifier:

```
S Chassis(rw-config)->ip access-list extended 100
S Chassis(rw-cfg-ext-acl)->
```

The following example creates a standard ACL with the name **ipv4acl1** as its identifier:

```
S Chassis(rw-config)->ip access-list standard ipv4acl1
S Chassis(rw-cfg-std-acl)->
```

IPv6 ACL Creation Examples

The following example creates a standard IPv6 ACL with the access-list number **acl1** as its identifier:

```
S Chassis(rw-config)->ipv6 access-list standard acl1
S Chassis(rw-cfg-ipv6-std-acl)->
```

The following example creates an extended IPv6 ACL with the access-list number **acl100** as its identifier:

```
S Chassis(rw-config)->ipv6 access-list extended 100
S Chassis(rw-cfg-ipv6-ext-acl)->
```

The following example creates a standard IPv6 ACL with the name **ipv6acl1** as its identifier:

```
S Chassis(rw-config)->ipv6 access-list standard ipv6acl1
S Chassis(rw-cfg-ipv6-std-acl)->
```

L2 ACL Creation

Create an L2 ACL using the **l2 access-list** command specifying an ACL name of up to 64 alpha-numeric characters. A name cannot be one of the **show access-lists** keywords **brief** or **applied**, or any prefix thereof such as ?br? or ?app?.

An existing L2 ACL can be copied to a non-existing L2 ACL. An existing L2 ACL can be appended to the end of another existing L2 ACL.

Upon creating the L2 ACL, you are placed in the access-list configuration command mode where you can enter rules or comment entries for this L2 ACL.

The following example creates the L2 ACL **list1**, if it does not already exist, and enters L2 ACL **list1** configuration mode:

```
S Chassis(rw-config)->l2 access-list list1
S Chassis(rw-cfg-l2-acl-list1)->
```

Creating ACL Rules

ACL rules define the basis upon which a hit will take place for the ACL. Rules in an ACL are order-dependent. A packet is either forwarded (a **permit** rule) or not forwarded (a **deny** rule) according to the first rule that is matched. The matching criteria available is determined based upon whether the ACL is a standard ACL, extended ACL, or L2 ACL. As soon as a rule is matched, processing of the access list stops. There is an implicit "deny all" rule at the end of every ACL. If all rules are missed, the packet is not forwarded.

L3 Standard ACL Rule Options

For a standard ACL, a source IPv4 address and an optional wildcard or IPv6 address and length are specified for the rule. For an extended ACL a source and destination IP address and wildcard are specified for the rule. In the case of an IPv4, Source and destination wildcards provide an inverted mask (specifies the don't care bits as 1s). 0.0.0.0 specifies an exact match. An **any** option is available for. The any option is short hand for 0.0.0.0 255.255.255.255.

L3 Extended ACL Rule Protocols and Other Options

For an extended ACL, the following protocols can be specified in a rule:

- A specific or all internet protocols
- Authentication Header protocol
- Encapsulation Security Payload
- Generic Router Encapsulation protocol
- An established TCP connection
- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP)
- Internet Control Message Protocol (ICMP or ICMPv6)

TCP and UDP rules can match source and destination ports against the following values: equal to, not equal to, greater than, less than, or a specified range. TCP rules can also distinguish established connections for new connection requests.

ICMP can be set for message type and code. See the details for the **permit** and **deny** commands in the *Extreme Networks S-Series CLI Reference* for supported ICMP message types and codes.

Extended ACLs can optionally be set for a Diffserv codepoint (DSCP), IP precedence, or IP Type of Service (ToS) value for both IPv4 and IPv6. IPv6 provides additional support for routing header match against source-routed packet, and the packet's routing extension header, mobility extension header, and mobility-type extension header.

ACL Rule Logging and Comments

Logging of ACL configuration activity is supported via syslog messages. This logging can be enabled for a specified entry, all entries, or the final implicit deny rule using the **log** entry command in access list configuration mode. Logging format can be in either a verbose or summary format.

Comments can be entered at the next available entry location, and, once entered, can be moved to a desired location.

ACL Rule Creation

Use the **permit** command to create a rule that forwards L3 packets or allows the packet at L2 based upon the defined rule.

Use the **deny** command to create a rule that prevents the forwarding of L3 packets or drops the packet at L2 based upon the defined rule.

IPv4 ACL examples

The following example creates a standard ACL **1**, and specifies an entry 1 permit rule with a source IP address of 10.0.0.1 and a wild card of 0.0.255.255. The explicit deny all rule denies all other traffic for this ACL:

```
S Chassis(rw-config)->ip access-list standard 1
S Chassis(rw-cfg-std-acl)->permit 10.0.0.1 0.0.255.255
S Chassis(rw-cfg-std-acl)->show access-lists 1
Standard IP access list 1  (2 entries)
  1 permit 10.0.0.1  0.0.255.255
  -- implicit deny all --
```

The following example creates an extended access-list 120 and configures a deny entry for the IP protocol with a source address 20.0.0.1 and source wildcard of 0.0.255.255 and a destination address of any. Syslog messaging is enabled to log any hit for this rule. This rule is followed by a permit rule for any other source or destination IP protocol traffic:

```
S Chassis(rw-config)->ip access-list extended 120
S Chassis(rw-cfg-ext-acl)->deny ip 20.0.0.1 0.0.255.255 any log
S Chassis(rw-cfg-ext-acl)->permit ip any any
S Chassis(rw-cfg-ext-acl)->show access-lists 120
Extended IP access list 120  (3 entries)
  1 deny   ip  20.0.0.1  0.0.255.255  any
  2 permit ip  any  any
  -- implicit deny all --
S Chassis(rw-cfg-ext-acl)->
```

IPv6 ACL Examples

This example enters configuration mode for standard IPv6 access list acl2 and configures a permit entry for source address 2001:1234:50:0:21f:45ff:fe3d:21be/64:

```
S Chassis(rw-config)->ipv6 access-list standard acl2
S Chassis(rw-cfg-ipv6-ext-acl)->permit 2001:1234:50:0:21f:45ff:fe3d:21be/64
S Chassis(rw-cfg-ipv6-ext-acl)->
```

This example enters configuration mode for extended IPv6 access list **acl120** and configures a permit entry for the IP protocol with a source address **2001:1234:50:0:21f:45ff:fe3d:21aa/64** and a destination address of any:

```
S Chassis(rw-config)->ipv6 access-list extended acl120
S Chassis(rw-cfg-ipv6-ext-acl)->permit ipv6 2001:1234:50:0:21f:45ff:fe3d:21aa/64
any
S Chassis(rw-cfg-ipv6-ext-acl)->
```

L2 ACL Examples

This example enters configuration mode for the **list1** L2 ACL and configures a permit entry for packets containing (verbose logging is enabled for this entry):

- Any source address
- A destination host with a MAC address of 00:11:88:fd:8e:f0
- VLANs 11 through 13
- An Ethernet II type 800

```
S Chassis(rw-config)->l2 access-list list1
S Chassis(rw-cfg-l2-acl)->permit any host 00:11:88:fd:8e:f0 vlan 11 13 ethertype
800 log-verbose
S Chassis(rw-cfg-l2-acl)->
```

Managing ACL Rules

Existing ACL rules can be deleted, moved, or replaced. New rules can be inserted at a specified location, otherwise rules are placed at the next available entry value. Comments can be entered into an ACL to provide useful information about the ACL. The contents of one or all ACLs can be displayed.

Deleting an ACL Rule

An ACL rule or range of rules can be deleted using the **delete** command.

The following example displays an extended L3 ACL **120** and deletes and deletes entries **2** and **3**:

```
S Chassis(rw-config)->ip access-list extended 120
S Chassis(rw-cfg-ext-acl)->show access-lists 120
Extended IP access list 120  (5 entries)
  1 deny   ip  20.0.0.1  0.0.255.255  any
  2 deny   ip  30.0.0.1  0.0.255.255  any
  3 deny   ip  40.0.0.1  0.0.255.255  any
  4 permit ip  any  any
  -- implicit deny all --
S Chassis(rw-cfg-ext-acl)->delete from 2 to 3
S Chassis(rw-cfg-ext-acl)->show access-lists 120
Extended IP access list 120  (3 entries)
  1 deny   ip  20.0.0.1  0.0.255.255  any
  2 permit ip  any  any
  -- implicit deny all --
```

The following example enters configuration mode for standard IPv6 access list **acl2** and deletes rule entry **10** - **12**:

```
S Chassis(rw-config)->ipv6 access-list standard acl2
S Chassis(rw-cfg-ipv6-std-acl)->delete from 10 to 12
S Chassis(rw-cfg-ipv6-std-acl)->
```

The following example enters configuration mode for the L2 ACL **list2** and deletes rule entry 10:

```
S Chassis(rw-config)->l2 access-list list2
S Chassis(rw-cfg-l2-acl)->delete 10
```

```
S Chassis(rw-cfg-l2-acl)->
```

Moving an ACL Rule

An ACL rule or range of rules can be moved to a different location in the ACL using the **move before** command.

The following example displays an extended ACL **121** and moves entries **3** and **4** to before entry **2**:

```
S Chassis(rw-config)->ip access-list extended 121
S Chassis(rw-cfg-ext-acl)->show access-lists 121
Extended IP access list 121 (5 entries)
  1 deny   ip  20.0.0.1  0.0.255.255  any
  2 permit ip  any  any
  3 deny   ip  30.0.0.1  0.0.255.255  any
  4 deny   ip  40.0.0.1  0.0.255.255  any
  -- implicit deny all --
S Chassis(rw-cfg-ext-acl)->move before 2 from 3 to 4
S Chassis(rw-cfg-ext-acl)->show access-lists 121
Extended IP access list 121 (5 entries)
  1 deny   ip  20.0.0.1  0.0.255.255  any
  2 deny   ip  30.0.0.1  0.0.255.255  any
  3 deny   ip  40.0.0.1  0.0.255.255  any
  4 permit ip  any  any
  -- implicit deny all --
```

This example enters configuration mode for standard IPv6 access list **acl2** and moves rule entries **10** - **12** before rule entry **5**:

```
S Chassis(rw-config)->ipv6 access-list standard acl2
S Chassis(rw-cfg-ipv6-std-acl)->move before 5 from 10 to 12
S Chassis(rw-cfg-ipv6-std-acl)->
```

This example enters configuration mode for L2 ACL **list2** and moves rule entry 20 before rule entry 10:

```
S Chassis(rw-config)->l2 access-list list2
S Chassis(rw-cfg-l2-acl)->move before 10 from 20 to 20
S Chassis(rw-cfg-l2-acl)->
```

Replacing an ACL Rule

An ACL rule or range of rules can be replaced by a specified permit, deny, or remark using the **replace** command.

The following example displays an extended ACL 121 and replaces entry 1 with a deny rule for source IP address 10.0.0.1 and destination IP address any:

```
S Chassis(rw-config)->ip access-list extended 121
S Chassis(rw-cfg-ext-acl)->show access-lists 121
Extended IP access list 121 (5 entries)
  1 deny   ip  20.0.0.1  0.0.255.255  any
  2 deny   ip  30.0.0.1  0.0.255.255  any
  3 deny   ip  40.0.0.1  0.0.255.255  any
```

```
  4 permit ip  any  any
  -- implicit deny all --
S Chassis(rw-cfg-ext-acl)->replace 1 deny ip 10.0.0.1 0.0.255.255 any
S Chassis(rw-cfg-ext-acl)->show access-lists 121
Extended IP access list 121 (5 entries)
  1 deny   ip  10.0.0.1  0.0.255.255  any
  2 deny   ip  30.0.0.1  0.0.255.255  any
  3 deny   ip  40.0.0.1  0.0.255.255  any
  4 permit ip  any  any
  -- implicit deny all --
```

This example replaces entry **1** of IPv6 access list **acl10** with a permit any source address :

```
S Chassis(rw-config)->ipv6 access-list standard acl10
S Chassis(rw-cfg-ipv6-std-acl)->replace 1 permit any
S Chassis(rw-cfg-ipv6-std-acl)->
```

This example replaces the current entry at sequence 17 with the remark “I am a remark entry at sequence number 17” in the L2 ACL **list1**:

```
S Chassis(rw-config)->l2 access-list list1
S Chassis(rw-cfg-l2-acl)->replace 17 remark “I am a remark entry at sequence
number 17”
```

Inserting an ACL Rule

When entering an ACL rule, the new rule is appended to the end of the ACL by default. A new ACL rule can be inserted into a specified entry location using the **insert before** command.

The following example displays an extended ACL 121 and inserts a new entry 2 with a deny rule for source IP address 20.0.0.1 and destination IP address any:

```
S Chassis(rw-config)->ip access-list extended 121
S Chassis(rw-cfg-ext-acl)->show access-lists 121
Extended IP access list 121 (5 entries)
  1 deny   ip  10.0.0.1  0.0.255.255  any
  2 deny   ip  30.0.0.1  0.0.255.255  any
  3 deny   ip  40.0.0.1  0.0.255.255  any
  4 permit ip  any  any
  -- implicit deny all --
S Chassis(rw-cfg-ext-acl)->insert before 2 deny ip 20.0.0.1 0.0.255.255 any
S Chassis(rw-cfg-ext-acl)->show access-lists 121
Extended IP access list 121 (6 entries)
  1 deny   ip  10.0.0.1  0.0.255.255  any
  2 deny   ip  20.0.0.1  0.0.255.255  any
  3 deny   ip  30.0.0.1  0.0.255.255  any
  4 deny   ip  40.0.0.1  0.0.255.255  any
  5 permit ip  any  any
  -- implicit deny all --
```

This example enters configuration mode for extended IPv6 access list **acl10** and inserts a rule before entry **10** that permits packets with a source address for host **2002:100::50** and a destination address of **2001:100::100:25/64** with a ToS value of **6**:

```
S Chassis(rw-config)->ipv6 access-list standard acl10
S Chassis(rw-cfg-ipv6-ext-acl)->insert before 10 permit host 2002:100::50
2001:100::100:25/64 traffic-class 6
S Chassis(rw-cfg-ipv6-ext-acl)->
```

This example enters configuration mode for the list1 L2 ACL and inserts at list sequence 5 a permit entry for packets containing (verbose logging is enabled for the inserted entry):

- Any source address
- A destination host with a MAC address of 00:11:88:fd:8e:f0
- VLANs 11 through 13
- An Ethernet II type 800

```
S Chassis(rw-config)->l2 access-list list1
S Chassis(rw-cfg-l2-acl)->insert before 5 permit any host 00:11:88:fd:8e:f0 vlan
11 13 ethertype 800 log-verbose
S Chassis(rw-cfg-l2-acl)->
```

Applying L3 and L2 ACLs

Once you have defined an ACL, the L3 ACL can be applied per routing interface and the L2 ACL can be applied per VLAN interface. An ACL can be applied to host access or an interface before it is created. The association of the name of the L2 or L3 ACL or number of the L3 ACL to the host or interface is persistent. You can use ACLs to filter traffic on individual interfaces, with a directional context (inbound, outbound, or both).

Use the **ip access-group** command to apply an IPv4 access-list, the **ipv6 access-group** command to apply an IPv6 access-list, or the **l2 access-group** command to apply a L2 access-list to an interface, in interface configuration command mode, specifying the access-list name, or in the case of L3 ACLs the number, followed by the directional context to which this ACL will be applied.

Use the **ip host-access** command for an IPv4 access-list and the **ipv6 host-access** command for an IPv6 access-list in configuration command mode, specifying the access-list number or name, to apply an ACL to host services for this device.

Use the **show access-lists applied** to display access-lists that have been applied to an interface.

The following example applies the extended ACL 121 to both the inbound and outbound direction on VLAN 2.

```
S Chassis(su-config)->interface vlan 2
S Chassis(su-config-intf-vlan.0.2)->ip access-group 121 in
S Chassis(su-config-intf-vlan.0.2)->ip access-group 121 out
S Chassis(su-config-intf-vlan.0.2)->show access-lists applied
Extended IP access list 121, applied inbound on interface 2  (5 entries)
Extended IP access list 121, applied outbound on interface 2  (5 entries)
S Chassis(su-config-intf-vlan.0.2)->
```

This example shows how to apply the standard access list acl10 for all inbound frames on VLAN 50. Based upon the definition of access list acl10, only frames with source

fe80:0:0:0:21f:45ff:fe3d:21aa/64 are routed. All the frames with other sources received on VLAN 50 are dropped:

```
S Chassis(su-config)->ipv6 access-list standard acl10
S Chassis(su-cfg-ipv6-std-acl)->permit fe80:0:0:0:21f:45ff:fe3d:21aa/64 log
S Chassis(su-cfg-ipv6-std-acl)->exit
S Chassis(su-config)->interface vlan 50
S Chassis(su-config-intf-vlan.0.50)->ipv6 access-group acl10 in
S Chassis(su-config-intf-vlan.0.50)->
```

This example shows how to apply L2 ACL **list1** for all inbound frames on VLAN 1:

```
S Chassis(rw-config)->interface vlan 1
S Chassis(rw-config-intf-vlan.0.1)->l2 access-group list1 in
```

Applying L3 ACLs to a VRF

Within VRF access configuration mode you can apply access lists to VRF access groups for the restriction of traffic to and from other VRFs. One ingress and one egress IPv4 and one ingress and one egress IPv6 access group may be applied to a VRF. The same access group may be applied to multiple VRFs.

Use the **vrf-access** command in VRF configuration mode to enter to enter VRF access configuration mode.

Once in VRF access configuration mode, you can apply:

- One ingress IPv4 access list from the specified VRF using the **ip access-group from-vrf** command or from any VRF using the **ip access-group from-any-vrf** command
- One ingress IPv6 access list from the specified VRF using the **ipv6 access-group from-vrf** command or from any VRF using the **ipv6 access-group from-any-vrf** command
- One egress IPv4 access list to the specified VRF using the **ip access-group to-vrf** command or to any VRF using the **ip access-group to-any-vrf** command
- One egress IPv6 access list to the specified VRF using the **ipv6 access-group to-vrf** command or to any VRF using the **ipv6 access-group to-any-vrf** command

Configuring ACLs

This section provides details for the configuration of ACLs on the S-Series products.

Procedure 52-1 describes how to create an IPv4 ACL and manage IPv4 ACLs at the ACL level.

Procedure 52-1 Creating and Managing IPv4 and IPv6 ACLs

Step	Task	Command(s)
1.	In global configuration command mode, create a standard or extended IPv4 or IPv6 ACL, or enter IPv4 or IPv6 ACL configuration mode for an already existing ACL.	**ipv4 access-list** {**standard** | **extended**} {*access-list-number* | *name*} **ipv6 access-list** {**standard** | **extended**} *name*
2.	In global configuration command mode, optionally, copy a preexisting IPv4 or IPv6 ACL to a non-existing IPv4 or IPv6 ACL.	**ipv4 ip access-list** {**standard** | **extended**} {*access-list-number* | *name*} **copy to** {*access-list-number* | *name*} **ipv6 ip access-list** {**standard** | **extended**} *name* **copy to** *name*

Procedure 52-1 Creating and Managing IPv4 and IPv6 ACLs (continued)

Step	Task	Command(s)
3.	In global configuration command mode, optionally, append a preexisting IPv4 or IPv6 ACL to another preexisting IPv4 or IPv6 ACL.	**ipv4 ip access-list** {**standard** | **extended**} {*access-list-number* | *name*} **append to** {*access-list-number* | *name*} **ipv6 ip access-list** {**standard** | **extended**} *name* **append to** *name*
4.	In global configuration command mode, optionally, check the efficiency of an IPv4 or IPv6 ACL.	**ipv4 ip access-list** {**standard** | **extended**} {*access-list-number* | *name*} **check** **ipv6 ip access-list** {**standard** | **extended**} *name* **check**

Procedure 52-2 describes how to create an L2 ACL and manage the L2 ACL at the ACL level.

Procedure 52-2 Creating and Managing L2 ACLs

Step	Task	Command(s)
1.	In global configuration command mode, create an L2 ACL, or enter L2 ACL configuration mode for an already existing ACL.	**l2 access-list** *name*
2.	In global configuration command mode, optionally, copy a preexisting L2 ACL to a non-existing L2 ACL.	**l2 access-list** *name* **copy to** *name*
3.	In global configuration command mode, optionally, append a preexisting L2 ACL to another preexisting L2 ACL.	**l2 access-list** *name* **append to** *name*
4.	In global configuration command mode, optionally, check the efficiency of an L2 ACL.	**l2 access-list** *name* **check**

Procedure 52-3 describes how to enter and manage standard ACL rules.

Procedure 52-3 Entering and Managing Standard IPv4 ACL Rules

Step	Task	Command(s)
1.	In IPv4 ACL configuration command mode, optionally, create a standard IPv4 ACL permit rule entry.	**permit** {*source source-wildcard* | **any** | **host** *ip-address*]} [**log** | **log-verbose**]
2.	In IPv4 ACL configuration command mode, optionally, create a standard IPv4 ACL deny rule entry.	**deny** {*source source-wildcard* | **any** | **host** *ip-address*]} [**log** | **log-verbose**]
3.	In IPv4 ACL configuration command mode, optionally, insert a new standard IPv4 ACL rule entry before the specified preexisting entry for this standard ACL.	**insert before** *entry* {**remark "***text***"** | {**permit** | **deny**} {*source source-wildcard* | **any** | **host** *ip-address*} [**log** | **log-verbose**]}
4.	In IPv4 ACL configuration command mode, optionally, replace the specified standard ACL entry with the specified new entry.	**replace** *entry* {**remark "***text***"** | **deny** {*source* [*source-wildcard*] | **any** | **host** *ip-address*] | **permit** {*source* [*source-wildcard*] | **any** | **host** *ip-address*]}

Procedure 52-4 describes how to enter and manage standard ACL rules.

Procedure 52-4 Entering and Managing Standard IPv6 ACL Rules

Step	Task	Command(s)
1.	In IPv6 ACL configuration command mode, optionally, create a standard IPv6 ACL permit rule entry.	**permit** {*source-address*/*length* | **any** | **host** *ip-address*]} [**log** | **log-verbose**]
2.	In IPv6 ACL configuration command mode, optionally, create a standard IPv6 ACL deny rule entry.	**deny** {*source-address*/*length* | **any** | **host** *ip-address*]} [**log** | **log-verbose**]
3.	In IPv6 ACL configuration command mode, optionally, insert a new standard IPv6 ACL rule entry before the specified preexisting entry for this standard ACL.	**insert before** *entry* { **remark** *text* | {**permit** | **deny**}} {*source-address*/*length* | **any** | **host** *ip-address*]} [**log** | **log-verbose**]
4.	In IPv6 ACL configuration command mode, optionally, replace the specified standard ACL entry with the specified new entry.	**replace** *entry* { **remark** *text* | {**permit** | **deny**}} {*source-address*/*length* | **any** | **host** *ip-address*]} [**log** | **log-verbose**]

Procedure 52-5 describes how to enter and manage extended IPv4 ACL rules.

Procedure 52-5 Entering and Managing Extended IPv4 ACL Rules

Step	Task	Command(s)
1.	In IPv4 ACL configuration command mode, optionally, create an extended IPv4 ACL permit rule entry.	**permit** {*protocol-num* | **ip** | **ah** | **esp** | **gre**} {*source source-wildcard* | **any** | **host** *ip-address*} {*destination destination-host wildcard* | **any** | **host** *ip-address*} [**dscp** *code*] [**precedence** *value*] [**tos** *value*] [**log** | **log-verbose**]
		permit tcp {*source source-wildcard* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *source-port*] [**range** *start-port end-port*] {*destination destination-host wildcard* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *dest-port*] [**range** *start-port end-port*] [**established**] [**dscp** *code*] [**precedence** *value*] [**tos** *value*] [**log** | **log-verbose**]
		permit udp {*source source-wildcard* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *source-port*] [**range** *start-port end-port*] {*destination destination-host wildcard* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *dest-port*] [**range** *start-port end-port*] [**dscp** *code*] [**precedence** *value*] [**tos** *value*] [**log** | **log-verbose**]
		permit icmp {*source source-wildcard* | **any** | **host** *ip-address*} {*destination destination-host wildcard* | **any** | **host** *ip-address*} [**msg** *icmp-msg*] [**dscp** *code*] [**precedence** *value*] [**tos** *value*] [**log** | **log-verbose**]

Procedure 52-5 Entering and Managing Extended IPv4 ACL Rules (continued)

Step	Task	Command(s)
2.	In IPv4 ACL configuration command mode, optionally, create an extended IPv4 ACL deny rule entry.	**deny** {*protocol-num* | **ip** | **ah** | **esp** | **gre**} {*source source-wildcard* | **any** | **host** *ip-address*} {*destination destination-host wildcard* | **any** | **host** *ip-address*} [**dscp** *code*] [**precedence** *value*] [**tos** *value*] [**log** | **log-verbose**] **deny tcp** {*source source-wildcard* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *source-port*] [**range** *start-port end-port*] {*destination destination-host wildcard* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *dest-port*] [**range** *start-port end-port*] [**established**] [**dscp** *code*] [**precedence** *value*] [**tos** *value*] [**log** | **log-verbose**] **deny udp** {*source source-wildcard* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *source-port*] [**range** *start-port end-port*] {*destination destination-host wildcard* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *dest-port*] [**range** *start-port end-port*] [**dscp** *code*] [**precedence** *value*] [**tos** *value*] [**log** | **log-verbose**] **deny icmp** {*source source-wildcard* | **any** | **host** *ip-address*} {*destination destination-host wildcard* | **any** | **host** *ip-address*} [**msg** *icmp-msg*] [**dscp** *code*] [**precedence** *value*] [**tos** *value*] [**log** | **log-verbose**]
3.	In IPv4 ACL configuration command mode, optionally, insert a new extended IPv4 ACL rule entry before the specified preexisting entry for this extended ACL. See the appropriate command syntax when entering a deny or permit rule to be inserted.	**insert before** *entry* {**remark "text"** | *deny-syntax* | *permit-syntax*}
4.	In IPv4 ACL configuration command mode, optionally, replace the specified extended IPv4 ACL entry with the specified new entry. See the appropriate command syntax when entering a deny or permit rule to be replaced.	**replace** *entry* {**remark "text"** | *deny-syntax* | *permit-syntax*}

Procedure 52-6 describes how to enter and manage extended IPv6 ACL rules.

Procedure 52-6 Entering and Managing Extended IPv6 ACL Rules

Step	Task	Command(s)
1.	In IPv6 ACL configuration command mode, optionally, create an extended IPv6 ACL permit rule entry.	**permit** {*protocol-num* | **ipv6** | **ah** | **esp** | **gre**} {*source-address/length* | **any** | **host** *ip-address*} {*destination-address/length* | **any** | **host** *ip-address*} [**dscp** *code*] [**traffic-class** *value*] [**flow-label** *value*] [**log** | **log-verbose**] [**routing**] [**routing-type** *type*] [**mobility**] [**mobility-type** *type*] **permit tcp** {*source-address/length* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *source-port*] [**range** *start-port end-port*] {*destination-address/length* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *dest-port*] [**range** *start-port end-port*] [**established**] [**dscp** *code*] [**traffic-class** *value*] [**flow-label** *value*] [**log** | **log-verbose**] [**routing**] [**routing-type** *type*] [**mobility**] [**mobility-type** *type*] **permit udp** {*source-address/length* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *source-port*] [**range** *start-port end-port*] {*destination-address/length* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *dest-port*] [**range** *start-port end-port*] [**dscp** *code*] [**traffic-class** *value*] [**flow-label** *value*] [**log** | **log-verbose**] [**routing**] [**routing-type** *type*] [**mobility**] [**mobility-type** *type*] **permit icmpv6** {*source-address/length* | **any** | **host** *ip-address*} {*destination-address/length* | **any** | **host** *ip-address*} [*icmpv6-type* [*icmpv6-code*] | **msg** *icmpv6-msg*] [**dscp** *code*] [**traffic-class** *value*] [**flow-label** *value*] [**log** | **log-verbose**] [**routing**] [**routing-type** *type*] [**mobility**] [**mobility-type** *type*]
2.		**deny** {*protocol-num* | **ipv6** | **ah** | **esp** | **gre**} {*source-address/length* | **any** | **host** *ip-address*} {*destination-address/length* | **any** | **host** *ip-address*} [**dscp** *code*] [**traffic-class** *value*] [**flow-label** *value*] [**log** | **log-verbose**] [**routing**] [**routing-type** *type*] [**mobility**] [**mobility-type** *type*] **deny tcp** {*source-address/length* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *source-port*] [**range** *start-port end-port*] {*destination-address/length* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *dest-port*] [**range** *start-port end-port*] [**established**] [**dscp** *code*] [**traffic-class** *value*] [**flow-label** *value*] [**log** | **log-verbose**] [**routing**] [**routing-type** *type*] [**mobility**] [**mobility-type** *type*]

Procedure 52-6 Entering and Managing Extended IPv6 ACL Rules (continued)

Step	Task	Command(s)
3.	In IPv6 ACL configuration command mode, optionally, create an extended IPv6 ACL deny rule entry.	**deny udp** {*source-address/length* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *source-port*] [**range** *start-port end-port*] {*destination-address/length* | **any** | **host** *ip-address*} [{**eq** | **neq** | **gt** | **lt**} *dest-port*] [**range** *start-port end-port*] [**dscp** *code*] [**traffic-class** *value*] [**flow-label** *value*] [**log** | **log-verbose**] [**routing**] [**routing-type** *type*] [**mobility**] [**mobility-type** *type*] **deny icmpv6** {*source-address/length* | **any** | **host** *ip-address*} {*destination-address/length* | **any** | **host** *ip-address*} [*icmpv6-type* [*icmpv6-code*] | **msg** *icmpv6-msg*] [**dscp** *code*] [**traffic-class** *value*] [**flow-label** *value*] [**log** | **log-verbose**] [**routing**] [**routing-type** *type*] [**mobility**] [**mobility-type** *type*]
4.	In IPv6 ACL configuration command mode, optionally, insert a new extended IPv6 ACL rule entry before the specified preexisting entry for this extended ACL. See the appropriate command syntax when entering a deny or permit rule to be inserted.	**insert before** *entry* {**remark "text"** | *deny-syntax* | *permit-syntax*}
5.	In IPv6 ACL configuration command mode, optionally, replace the specified extended IPv6 ACL entry with the specified new entry. See the appropriate command syntax when entering a deny or permit rule to be replaced.	**replace** *entry* {**remark "text"** | *deny-syntax* | *permit-syntax*}

Procedure 52-7 describes how to enter and manage L2 ACL rules.

Procedure 52-7 Entering and Managing L2 ACL Rules

Step	Task	Command(s)
1.	In L2 ACL configuration command mode, optionally, create a L2 ACL permit rule entry.	**permit** {**any** | **host** *source-macAddr* | *source-macAddr source-wildcard*} [**any** | **host** *destination-macAddr* | *destination-macAddr destination-wildcard*] [**dei**] [**cos** *cos*] [**vlan** *vlan* [*vidhi*]] [**ethertype** *data*] [**log** | **log-verbose**]
2.	In L2 ACL configuration command mode, optionally, create a L2 ACL deny rule entry.	**deny** {**any** | **host** *source-macAddr* | *source-macAddr source-wildcard*} [**any** | **host** *destination-macAddr* | *destination-macAddr destination-wildcard*] [**dei**] [**cos** *cos*] [**vlan** *vlan* [*vidhi*]] [**ethertype** *data*] [**log** | **log-verbose**]
3.	In L2 ACL configuration command mode, optionally, insert a new L2 ACL rule entry before the specified preexisting entry for this L2 ACL.	**insert before** *entry* {**remark** "*text*" | {**permit** | **deny**} {**any** | **host** *source-macAddr* | *source-macAddr source-wildcard*} [**any** | **host** *destination-macAddr* | *destination-macAddr destination-wildcard*] [**dei**] [**cos** *cos*] [**vlan** *vlan* [*vidhi*]] [**ethertype** *data*] [**log** | **log-verbose**]

Procedure 52-7 Entering and Managing L2 ACL Rules (continued)

Step	Task	Command(s)
4.	In L2 ACL configuration command mode, optionally, replace the specified L2 ACL entry with the specified new entry.	**replace** *entry* {**remark "***text***"** | {**permit** | **deny**} {**any** | **host** *source-macAddr* | *source-macAddr source-wildcard*} [**any** | **host** *destination-macAddr* | *destination-macAddr destination-wildcard*] [**dei**] [**cos** *cos*] [**vlan** *vlan* [*vidhi*]] [**ethertype** *data*] [**log** | **log-verbose**]

Procedure 52-8 describes how to manage ACL rules.

Procedure 52-8 Managing IPv4, IPv6 and L2 ACL Rules

Step	Task	Command(s)
1.	In IPv4, IPv6, or L2 ACL configuration command mode, optionally, enable logging for the specified rule, the final implicit deny rule, or all rules.	**log** [*entry*] [**implicit**] [**all**]
2.	In IPv4, IPv6, or L2 ACL configuration command mode, optionally, delete a preexisting ACL rule entry.	**delete** {*entry* | **from** *entry* **to** *entry*}
3.	In IPv4, IPv6, or L2 ACL configuration command mode, optionally, move a preexisting ACL entry before the specified entry or range of entries.	**move before** *entry* **from** *entry* **to** *entry*
4.	In IPv4, IPv6, or L2 ACL configuration command mode, optionally, enter a text comment as the next ACL entry.	**remark "***text***"**

Procedure 52-9 describes how to apply and display ACLs.

Procedure 52-9 Applying and Displaying ACLs

Step	Task	Command(s)
1.	In interface configuration command mode, apply an ACL to a routing interface specifying the whether the ACL applies to inbound or outbound frames.	**ipv4 access-group** {*access-list-number* | *name*} {**in** | **out**} **ipv6 access-group** *access-list-name* {**in** | **out**} **l2 access-group** *name* {**in** | **out**}
2.	In configuration command mode, apply an IPv4 or IPv6 ACL to the host services for this device.	**ipv4 host-access** {*access-list-number* | *name*} **ipv6 host-access** *name*
3.	In any command mode, optionally, display ACL configuration.	**show access-lists** [*access-list-number* | *name*] [**from** *start-range* **to** *end-range*]] [**brief**]
4.	In any command mode, optionally, display applied ACLs.	**show access-lists applied** [**host** | **interfaces** [**vlan** | **inbound** | **outbound** | **in-and-out**]]
5.	In any command mode, optionally, clear ACL display counters.	**clear access-lists counters** [{*access-list-number* | *name*} | **applied** [**host** | **interfaces** [**vlan** *vlan-id*] [**inbound** | **outbound** | **in-and-out**]]

Procedure 52-10 describes how to enter VRF access configuration mode and apply ACLs.

Procedure 52-10 Entering VRF Access Mode and Applying ACLs

Step	Task	Command(s)
1.	In VRF configuration mode, enter VRF access configuration mode.	**vrf-access**
2.	In VRF access configuration mode, apply an IPv4 access list to traffic from the specified VRF.	**ip access-group** *list-name* **from-vrf** *vrf-name*
3.	In VRF access configuration mode, apply an IPv4 access list to traffic inbound from any VRF.	**ip access-group** *list-name* **from-any-vrf**
4.	In VRF access configuration mode, apply an IPv4 access list to traffic outbound to the specified VRF.	**ip access-group** *list-name* **to-vrf** *vrf-name*
5.	In VRF access configuration mode, apply an IPv4 access list to traffic outbound to any VRF.	**ip access-group** *list-name* **to-any-vrf**
6.	In VRF access configuration mode, apply an IPv6 access list to traffic from the specified VRF.	**ipv6 access-group** *list-name* **from-vrf** *vrf-name*
7.	In VRF access configuration mode, apply an IPv6 access list to traffic from any VRF.	**ipv6 access-group** *list-name* **from-any-vrf**
8.	In VRF access configuration mode, apply an IPv6 access list to traffic outbound to the specified VRF.	**ipv6 access-group** *list-name* **to-vrf** *vrf-name*
9.	In VRF access configuration mode, apply an IPv6 access list to traffic outbound to any VRF.	**ipv6 access-group** *list-name* **to-any-vrf**

Terms and Definitions

Table 52-1 lists terms and definitions used in this ACL configuration discussion.

Table 52-1 ACL Configuration Terms and Definitions

Term	Definition
Access Control List (ACL)	A container of permit, deny, and comment entries for the purpose of • Forwarding or not forwarding L3 packets based upon one or more packet fields, such as source and destination IP address, and protocol • Allowing or dropping L2 packets based upon one or more packet fields such as source and destination MAC address, DEI, or VLAN
entry	A member of an ACL that either permits or denies the packet based upon one or more specified packet fields, or provides an ACL comment.
DEI	The drop eligibility indicator in the VLAN tag
rule	An ACL entry that allows or drops packets using a permit or deny entry.
standard ACL	An ACL for which forwarding decisions are made based only upon a source IP address.
extended ACL	An ACL for which forwarding decisions are made based upon the packet protocol, source and destination ip address, or host address, port matching in the case of the TCP or UDP protocols, as well as, optionally, a specified DSCP, ToS, or IP precedence value.

Table 52-1 ACL Configuration Terms and Definitions (continued)

Term	Definition
Layer 2 (L2) ACL	An ACL for which permit or deny decisions are made based upon some combination of packet source and destination MAC address, DEI, Class of Service, VLAN, and Ethernet II type.
VRF access	A VRF command mode in which access lists can be applied to groups to and from specified or any VRF for this VRF context.

53

Quality of Service (QoS) Configuration

This chapter describes the QoS feature as it is implemented on the Extreme Networks S-Series devices.

Using Quality of Service in Your Network

Quality of Service (QoS) is:

- A mechanism for the management of bandwidth
- The ability to give preferential treatment to some packets over others
- Based upon packet classification and forwarding treatment

You configure packet preference and forwarding treatment based upon a flow's sensitivity to delay, delay variation (jitter), bandwidth, availability and packet drop. QoS uses packet priority, in conjunction with queue treatment configuration, to determine the interface's inbound and forwarding behavior for this packet. Packet preference and forwarding treatment for a given flow can be applied to roles configured in Extreme Networks policy.

Without QoS, all packets are treated as though the delivery requirements and characteristics of any given packet are equal to any other packet. In other words, non-QoS packet delivery is not able to take into account application sensitivity to packet delay, jitter, amount of bandwidth required, packet loss, or availability requirements of the flow. QoS provides management mechanisms for these flow characteristics.

QoS achieves its bandwidth management capabilities by:

- Setting priorities that define traffic handling
- Dedicating bandwidth and prioritizing queuing for specific applications, and reducing packet transmission delay and jitter
- Managing congestion by shifting packet loss to applications that can tolerate it

The S-Series Flex-Edge feature, supported on all S-Series switches, provides the unique capability to classify traffic as it enters the switch. Traffic critical to ensuring the operational state of the network and to maintain application continuity is identified and prioritized at ingress, prior to being passed on for packet processing. See "Flex-Edge" on page 53-2 for more details.

Implementing Quality of Service

QoS determines how a flow will be treated as it transits the link. To determine how a flow should be treated, you must first understand the characteristics of the flows on your network, and secondly, you must identify these flows in a way that QoS can recognize. In this sense, QoS is the third step in a three step process. The three steps Extreme Networks recommends for configuring QoS are:

- Understand your network flows using NetFlow. See Chapter 33, **NetFlow Configuration** for NetFlow configuration details.
- Associate the flows on your network with a well defined role using Extreme Networks policy. See Chapter 26, **Policy Configuration** for policy configuration details.
- Configure the appropriate link behavior for that role by associating the role with a QoS configuration.

Quality of Service Overview

QoS is all about managing the bandwidth in a manner that aligns the delivery characteristics of a given flow with the available port resources. In a QoS context, a flow is a stream of IP packets that are classified with the same class of service as it transits the interface. QoS manages bandwidth for each flow by taking advantage of its ability to:

- Assign different priority levels to different packet flows
- Mark or re-mark the packet priority at port ingress with a Type of Service
- Sort flows by transit queue such that a higher priority queue gets preferential access to bandwidth during packet forwarding
- Limit the amount of bandwidth available to a given flow by either dropping (rate limiting) or buffering (rate shaping) packets in excess of configured limits

These QoS abilities collectively make up a Class of Service (CoS). The remainder of this section will briefly describe CoS and its components.

Flex-Edge

All S-Series switches support the Flex-Edge feature, which provides a unique mechanism for the classification of traffic as it enters the switch. With the Extreme Networks Flex-Edge feature, the switch is significantly less vulnerable to network congestion issues at peak traffic times. Traffic critical to ensuring the operational state of the network and maintaining application continuity is identified and prioritized at ingress, prior to being passed on for packet processing. Network high availability is assured, and important users and applications are guaranteed bandwidth and priority.

The Flex-Edge feature assigns one of five traffic categories to each packet as it enters the switch. Flex-Edge, using the advanced Media Access Control (MAC) capability on the switch, queues each of five traffic categories into its own prioritized queue. Each queue will not pass any traffic on to the packet processor until all higher priority queues are empty (see "Strict Priority Queuing" on page 53-5 for more information on this type of queuing).

If flow control is enabled on the port, either manually or using auto-negotiation, Flex-Edge applies backpressure to front and aggregator ports to avoid discard. The MAC capability monitors traffic on all ports, by category and priority, and makes intelligent decisions concerning which front panel ports to initiate flow control on, by sending a MAC PAUSE frame to the sending device out the port causing the congestion.

Note: The Flex-Edge feature and the port priority (IEEE 802.1D) configuration are functionally separate and have no affect on each other.

Priority queueing, from high priority to low priority, is given to the following five traffic categories:

1. Network control – Protocol packets necessary for maintaining network topology such as:
   - L2 (STP, GVRP, LACP)
   - L3 (VRRP, OSPF, RIP, BGP, DVMRP, PIM)
   - ARP
2. Network discovery - Protocol packets used for dissemination of network characteristics such as: LLDP, CtronDP, and CiscoDP
3. Authentication
4. Configured drop-precedence - Packets associated with a policy rule that specifies a Class of Service with a configured drop-precedence of favored (**0**), best-effort (**1**), or unfavored (**2**)
5. Best effort - All traffic that doesn't fall into any other category listed here

Network control, network discovery, and authentication priorities are hard coded and cannot be modified. Drop-precedence is assigned to a Class of Service using the **set cos settings** command and applied to a policy rule using the **set policy rule** command. Best-effort is traffic that is undefined within the Flex-Edge context, and therefore by definition cannot be configured for purposes of backpressure or packet drop. Best-effort categorized traffic is given the lowest priority by the Flex-Edge mechanism, with the exception of unfavored drop-precedence which is the lowest priority possible within the Flex-Edge mechanism.

The only user configurable aspect of the Flex-Edge feature is drop-precedence. Drop-precedence is a CoS settings option. CoS settings are assigned to a policy rule. In a Flex-Edge context, drop precedence is limited to rules that apply to a single port and specify a traffic classification of either port or macsource. For any packets matching the policy rule, you can assign one of three drop-precedence priority levels:

- Favored - A drop-precedence value of **0** provides a better chance of being passed on for packet processing than traffic categorized as best-effort.
- Best-Effort - A drop-precedence value of **1** provides a best-effort level of priority within the Flex-Edge priority scheme.
- Unfavored - A drop-precedence value of **2** provides a somewhat worse chance of being passed on for packet processing than traffic categorized as best-effort. This is the lowest possible priority setting within the Flex-Edge mechanism.

Class of Service (CoS)

You implement QoS features in a Class of Service (CoS). How the firmware treats a packet as it transits the link depends upon the priority and forwarding treatments configured in the CoS. Up to 256 unique CoS entries can be configured. CoS entries 0–7 are configured by default with an 802.1p priority assigned and default forwarding treatment. For purposes of backward

compatibility, CoS entries 0–7 cannot be removed. CoS entries 8-255 can be configured for the following services:

- 802.1p priority
- IP ToS rewrite value
- Priority Transmit Queue (TxQ) with configurable forwarding behavior
- In-bound (IRL) and outbound (ORL) rate limiter per transmit queue
- Outbound rate shaper per transmit queue

The CoS configuration for each service can be easily viewed using the CoS setting tables. Ports are bundled into port groups with the group assigned to a CoS, significantly cutting down on operational overhead and complexity.

CoS Priority and ToS Rewrite

The two parameters configurable for CoS priority are 802.1p and Type of Service (ToS). Each CoS can be mapped to an 802.1p priority and a ToS rewrite value. The 802.1p parameter is:

- A subset of ToS with values 0–7 (upper 3 bits of the 8 bit ToS field)
- Supported in both layer 2 and layer 3

The ToS parameter is:

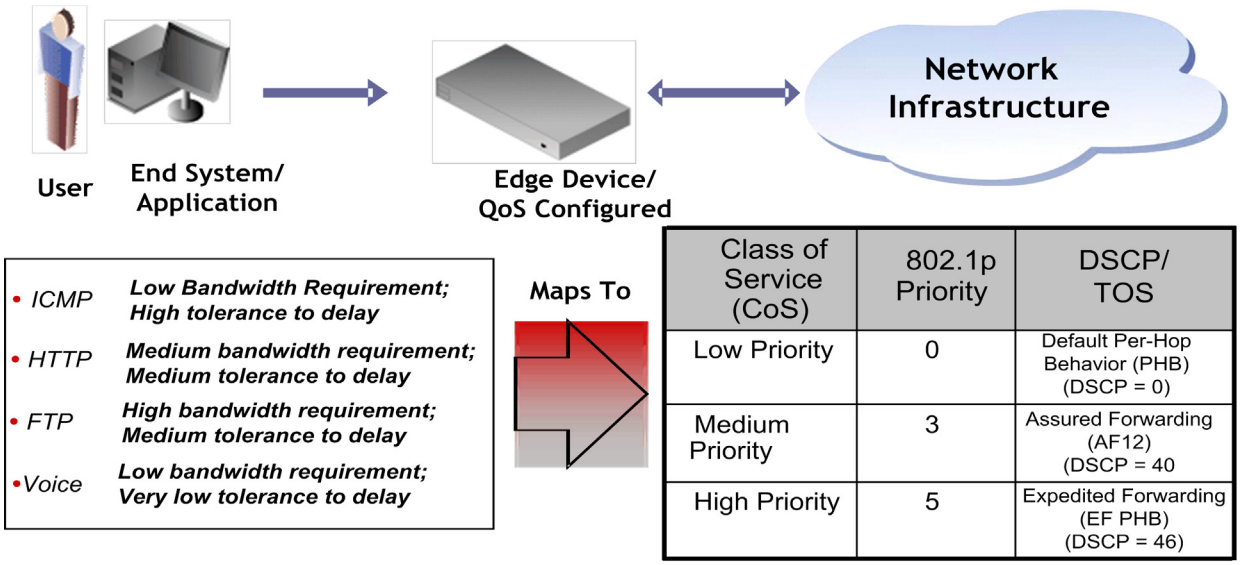
- An 8-bit field with values 0–255
- Supported in layer 3 only
- Also referred to as the Differentiated Services Code Point (DSCP) when limited to the lower 5 bits of the field

Figure 53-1 displays the relationship between your application, priority level, 802.1p, and ToS assignments (shown here using DSCP terminology).

QoS priority/ToS configuration:

- Derives its characteristic requirements from the end-system application
- Is configured on the edge device the application is connected to
- Is propagated through the network in the protocol packet header

Figure 53-1 Assigning and Marking Traffic with a Priority



The ICMP protocol, used for error messaging, has a low bandwidth requirement, with a high tolerance for delay and jitter, and is appropriate for a low priority setting. HTTP and FTP protocols, used respectively for browser-generated and file transfer traffic, have a medium to high bandwidth requirement, with a medium to high tolerance for delay and jitter, and are appropriate for a medium priority level. Voice (VoIP), used for voice calls, has a low bandwidth requirement, but is very sensitive to delay and jitter and is appropriate for a high priority level.

See RFC 1349 for further details on ToS. See RFCs 2474 and 2475 for further details on DSCP.

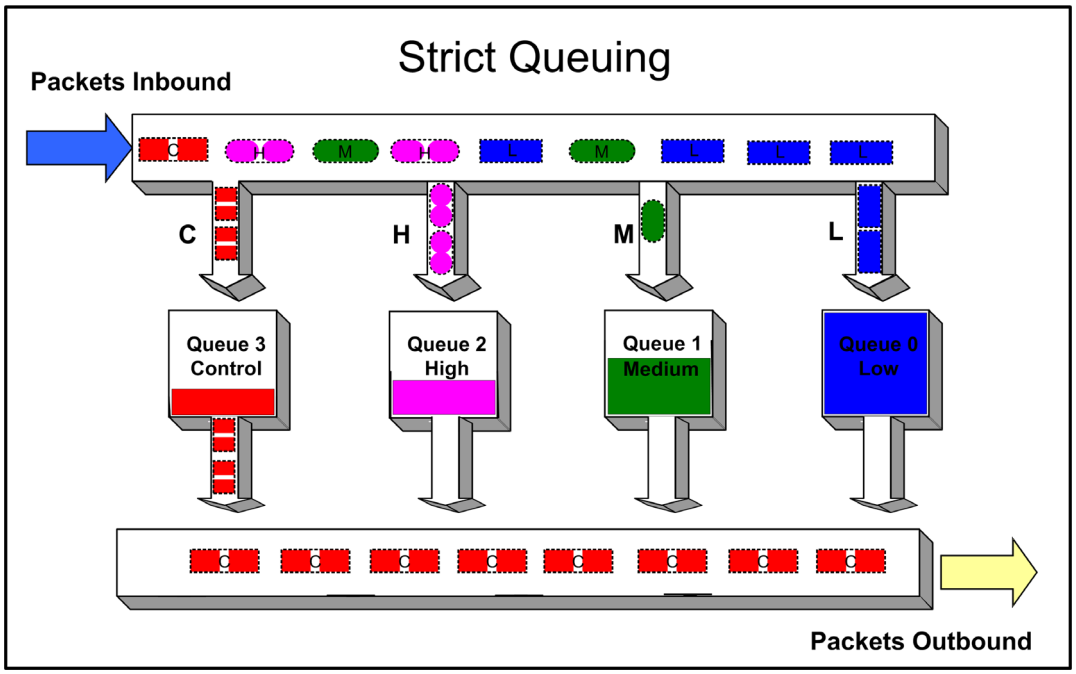
Preferential Queue Treatment for Packet Forwarding

There are three types of preferential queue treatments for packet forwarding: strict priority, weighted fair, and hybrid.

Strict Priority Queuing

With strict priority queuing, a higher priority queue must be empty before a lower priority queue can transmit any packets. Strict priority queuing is depicted in Figure 53-2. Inbound packets enter on the upper left and proceed to the appropriate queue, based upon the TxQ configuration in the CoS. Outbound packets exit the queues on the lower right. At this time only queue 3 packets are forwarded. This will be true until queue 3 is completely empty. Queue 2 packets will then be forwarded. Queue 1 packets will only forward if both queue 2 and queue 3 are empty. Queue 0 packets will only forward if all other queues are empty. Strict priority queuing assures that the highest priority queue with any packets in it will get 100 percent of the bandwidth available. This is particularly useful for one or more priority levels with low bandwidth and low tolerance for delay. The problem with strict priority queuing is that should the higher level queues never fully empty, lower level queues can be starved of bandwidth.

Figure 53-2 Strict Priority Queuing Packet Behavior



Low Latency Queuing

A Low Latency Queue (LLQ) is a non-configurable strict priority queue. LLQs are designed to guard against:

- Packet loss
- Delay
- Jitter

LLQ hardware resources can not be configured, but a policy can be configured for a CoS that is mapped to an LLQ. In this way, traffic associated with high value real-time voice or video packets can be mapped to an LLQ. The LLQ priority will determine when mapped traffic will be serviced relative to other traffic. For example, S-Series queues 0, 9, and 10 are LLQs. If a voice policy is mapped to a CoS with a TxQ reference that is in turn mapped to queue 9, this voice traffic will be serviced as soon as queue 10 is empty and will continue to be serviced ahead of any lower priority queue until there is no traffic left in queue 9.

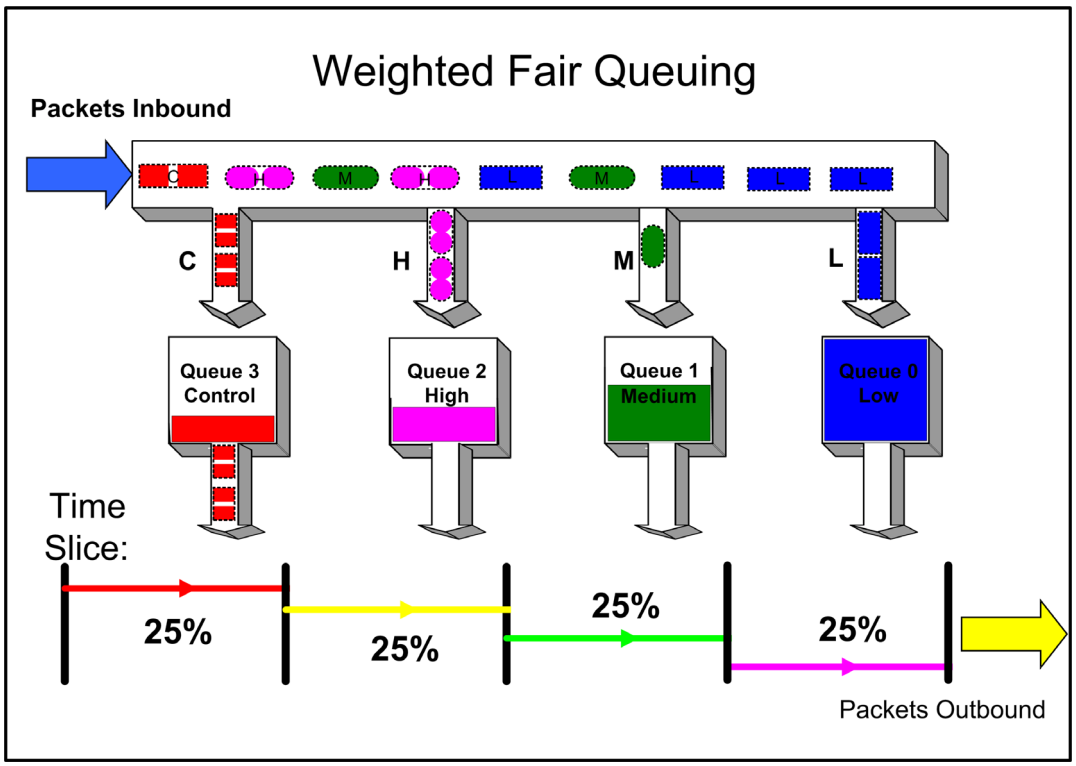
LLQs are hardware dependent. Use the **show cos port-config txq** command to display LLQs for a given module.

Weighted Fair Queuing

With weighted fair queuing, queue access to bandwidth is divided up by percentages of the time slices available. For example, if 100 percent is divided into 64 time slices, and each queue is configured for 25 percent, each queue will get 16 time slices, after which the next lowest priority queue will get the next 16, and so on. Should a queue empty before using its current share of time slices, the next priority queue inherits the time slices that remain. Figure 53-3 on page 53-7 depicts how weighted fair queuing works. Inbound packets enter on the upper left of the box and proceed to the appropriate priority queue. Outbound packets exit the queues on the lower right. Queue 3 has access to its percentage of time slices so long as there are packets in the queue. Then queue 2 has access to its percentage of time slices, and so on round robin. Weighted fair queuing assures

that each queue will get at least the configured percentage of bandwidth time slices. The value of weighted fair queuing is in its assurance that no queue is starved for bandwidth. The downside of weighted fair queuing is that packets in a high priority queue, with low tolerance for delay, will wait until all other queues have used the time slices available to them before forwarding. So weighted fair queuing would not be appropriate for applications with high sensitivity to delay or jitter, such as VoIP.

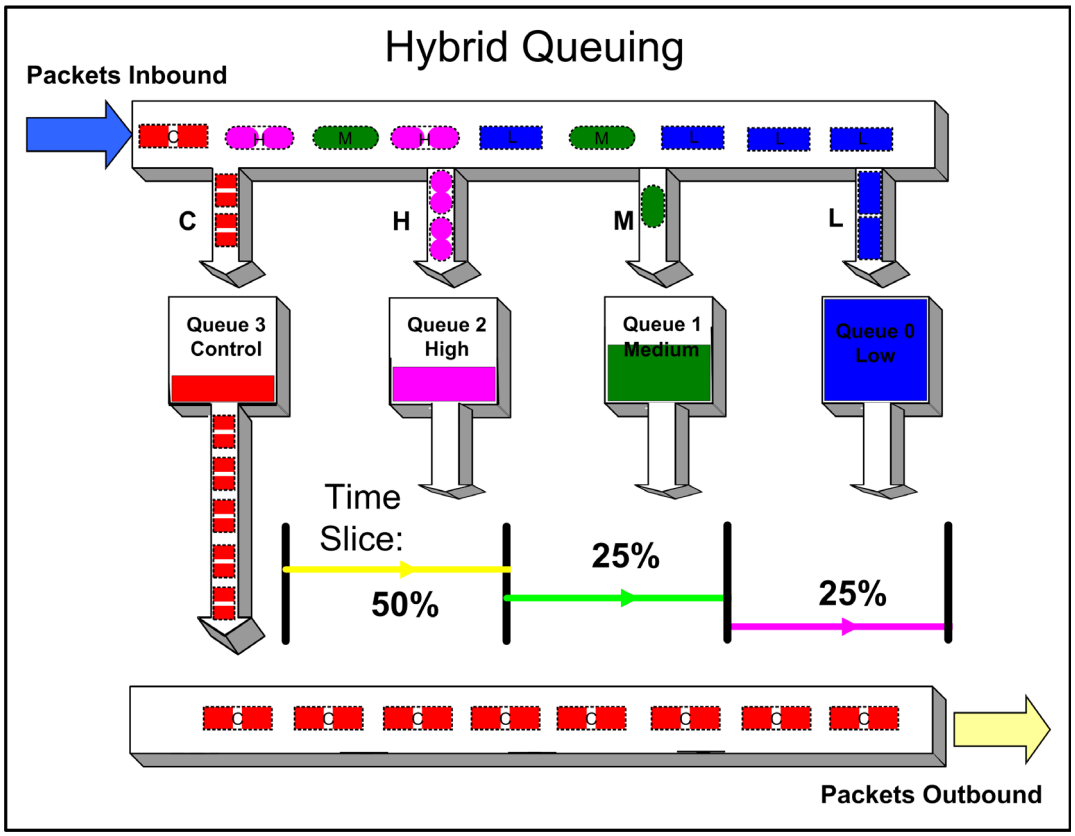
Figure 53-3 Weighted Fair Queuing Packet Behavior



Hybrid Queuing

Hybrid queuing combines the properties of both strict priority and weighted fair queuing. Figure 53-4 on page 53-8, depicts hybrid queuing. The configuration is for strict priority queuing on queue 3 and weighted fair queuing for the remaining queues, with queue 2 receiving 50 percent of the remaining time slices, and the other queues receiving 25 percent each. The benefit of hybrid queuing is that queues configured as strict priority will receive all the bandwidth that is available in the order of their priority until empty. Remaining bandwidth will be used by the weighted fair queues based upon the time slice percentages configured. The down side remains that anytime strict priority queuing is used, should the strict priority queues never fully empty, remaining queues will be starved of bandwidth.

Figure 53-4 Hybrid Queuing Packet Behavior



Enhanced Transmission Selection

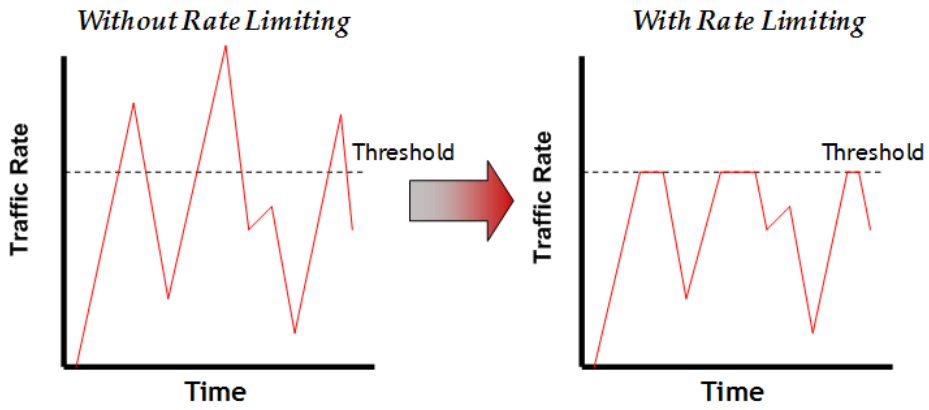
Enhanced Transmission Selection (ETS) queuing provides for configuring two or more traffic class queues (transmit queue (TxQ)) to be allocated for bandwidth that will not be serviced until all non-ETS queues are empty. See “Enhanced Transmission Selection Configuration” on page 19-2 for ETS feature details.

Rate Limiting

Rate limiting is used to control the rate of traffic entering (inbound) and/or leaving (outbound) a switch per CoS. Rate limiting allows for the throttling of traffic flows that consume available bandwidth, in the process providing room for other flows. Rate limiting guarantees the availability of bandwidth for other traffic by preventing the rate limited traffic from consuming more than the assigned amount of a network’s resources. Rate limiting accomplishes this by setting a cap on the bandwidth utilization of specific types of both inbound and outbound traffic. When a rate limit has been exceeded, the CoS can be configured to perform one or all of the following: record a Syslog message, send an SNMP trap to inform the administrator, and automatically disable the port.

Figure 53-5 on page 53-9 illustrates how bursty traffic is clipped above the assigned threshold with rate limiting applied.

Figure 53-5 Rate Limiting Clipping Behavior

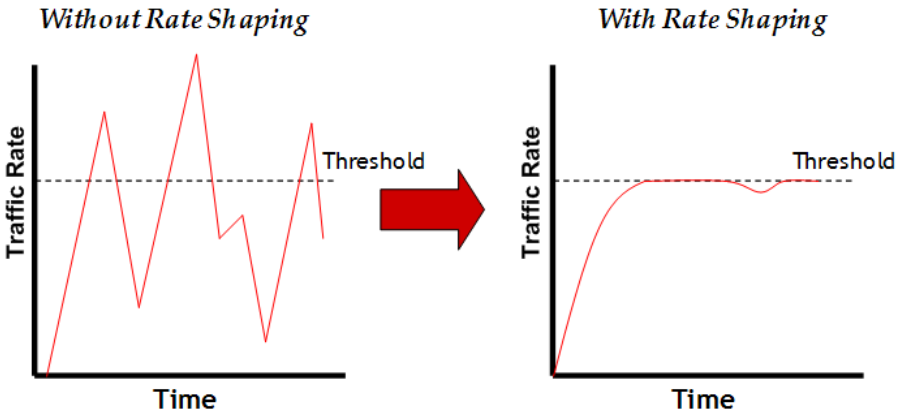


Rate Shaping

Rate Shaping throttles the rate at which a port transmits (outbound) queued packets. Rate Shaping buffers packets received above the configured rate on a per CoS basis, rather than dropping them. Only when buffer capacity is exceeded are packets dropped. Rate shaping may be configured for a CoS on a port, for an 802.1p priority on a port, or for all Classes of Service on a port.

Figure 53-6 illustrates how bursty traffic is smoothed out when it bursts above the assigned threshold with rate shaping applied.

Figure 53-6 Rate Shaping Smoothing Behavior



Rate shaping retains excess packets in a queue and then schedules these packets for later transmission over time. Therefore, the packet output rate is smoothed and bursts in transmission are not propagated as seen with rate limiting.

Rate shaping can be implemented for multiple reasons, such as controlling bandwidth, to offer differing levels of service, or to avoid traffic congestion on other links in the network by removing the burstiness property of traffic that can lead to discarded packets. Rate shaping is important for real-time traffic, where packet loss is extremely detrimental to these applications. Instead of discarding traffic imposed by rate limiting, delays are induced into its transmission by retaining the data for future transmission. However, the delays must also be bounded to the degree that the traffic is sensitive to delays.

Understanding QoS Configuration on the S-Series

This section discusses the six components for configuring QoS and displaying QoS status on an S-Series switch router:

CoS Port-Type: Based upon the transmit queue (TxQ), Inbound Rate Limiting (IRL), Outbound Rate Limiting (ORL), and flood control resource capabilities of the ports in your system. Knowledge of these capabilities is important when configuring queue behaviors. Port group membership and the port resources available are determined by port type.

CoS Port Groups: Provide for the grouping of ports by the same class of service features and port type.

CoS Port Resource Table: Enables the association of rate limiter and rate shaper values to a port.

CoS Reference Mapping Table: Maps your defined TxQ, IRL, and ORL index references, used by the CoS settings table, to the physical queue and rate limiter settings created in the port-resource table.

CoS Settings Table: Used for CoS parameter assignment and contains the current settings for each class of service feature. Each class of service entry consists of an entry index, an 802.1p priority, an optional ToS value, a transmit queue reference, an IRL reference, an ORL reference and a flood control reference.

CoS State: A global setting that must be enabled for a configured CoS to affect port behavior. When enabled, CoS state associated with a port supersedes current default or modified port-level controls for priority queue mapping, port rate limiting, and transmit queue. When disabled the port settings apply.

Note: It is recommended that you use Extreme Networks NetSight Policy Manager as an alternative to CLI for configuring policy-based CoS on Extreme Networks Series devices.

A policy discussion is outside the scope of this document and will be limited to the relevant configuration example commands. See Chapter 26, **Policy Configuration** for a detailed policy discussion.

Numerous QoS values are associated with each other through reference. With the exception of 802.1p priority and ToS, CoS values are first mapped to a port group, which associates a CoS configuration with a port type. A port group has the following CoS parameters associated with it:

- Physical port(s)
- Strict priority or weighted fair queuing behavior
- Rate-limit setting(s)
- Rate-shaping setting(s)
- A port queue
- A port reference

Understanding how these parameters are first mapped to the port group and then to a TxQ or IRL reference is the key to understanding QoS configuration. Where appropriate, the task column in Procedure 53-1 on page 53-23 identifies these mapping relationships.

See "Determining CoS Port-Type" on page 53-10 and "Configuring CoS Port Groups" on page 53-13 for a port group discussion.

Determining CoS Port-Type

Based on physical capability, all physical ports belong to one of two port-types. The importance of this port-type distinction lies in the resources available for transmit queue, inbound rate limiting,

outbound rate limiting, and flood control CoS features. The nomenclature distinguishes the types as port type 0 (11 queues and port type 1 (4 queues).

TxQ

Port type 0 supports eleven transmit queues, while type 1 supports four. Use the **show cos port-type txq** to display all the system's ports currently associated to each type.

The following example displays default values for the **show cos port-type txq** command output:

```
S Chassis(rw)->show cos port-type txq

Number of resources:              Supported rate types:
 txq = transmit queue(s)           perc  = percentage
 irl = inbound rate limiter(s)     pps   = packets per second
 orl = outbound rate limiter(s)    Kbps  = kilobits per second
 fld = flood rate limiter(s)       Mbps  = megabits per second
                                   Gbps  = gigabits per second
                                   Tbps  = terabits per second

                    Number of
                    slices /
       Port type    Number of  Supported  Eligible          Unselected
Index  description  queues     rate type  ports             ports
-----  -----------  ---------  ---------  ----------------  ----------------
0      S-Series      100/11    perc       ge.1.1-48;        ge.1.1-48;
       11Q                     Kbps       tg.1.101-104;     tg.1.101-104;
                               Mbps       tg.1.201-204;     tg.1.201-204;
                               Gbps       tg.3.1-8;         tg.3.1-8;
                                          tg.3.101-104;     tg.3.101-104;
                                          tg.3.201-204      tg.3.201-204
```

IRL

Type 0 supports 24 Inbound Rate Limiters. Type 1 supports 32 Inbound Rate Limiters. Use the **show cos port-type irl** command to display the port types and their associated ports.

The following example displays default values for the **show cos port-type irl** command output:

```
S Chassis(rw)->show cos port-type irl

Number of resources:              Supported rate types:
 txq = transmit queue(s)           perc  = percentage
 irl = inbound rate limiter(s)     pps   = packets per second
 orl = outbound rate limiter(s)    Kbps  = kilobits per second
 fld = flood rate limiter(s)       Mbps  = megabits per second
                                   Gbps  = gigabits per second
                                   Tbps  = terabits per second

       Port type    Number of  Supported  Eligible          Unselected
Index  description  limiters   rate type  ports             ports
-----  -----------  ---------  ---------  ----------------  ----------------
0      S-Series       24 irl   perc       None              None
       24 IRL                  pps
                               Kbps
                               Mbps
                               Gbps

1      SK-Series      32 irl    perc       ge.4.1-48;        None
       32 IRL                  pps        ge.6.1-48;
                               Kbps       ge.8.1-48;
```

```
                                    Mbps       ge.8.101-112
                                    Gbps
```

ORL

Type 0 supports 4 Outbound Rate Limiters, while type 1 supports sixteen Outbound Rate Limiters. Use the **show cos port-type orl** command to display the port types and their associated ports.

The following example displays default values for the **show cos port-type orl** command output:

```
S Chassis(rw)->show cos port-type orl

Number of resources:              Supported rate types:
 txq = transmit queue(s)           perc  = percentage
 irl = inbound rate limiter(s)     pps   = packets per second
 orl = outbound rate limiter(s)    Kbps  = kilobits per second
 fld = flood rate limiter(s)       Mbps  = megabits per second
                                   Gbps  = gigabits per second
                                   Tbps  = terabits per second

       Port type     Number of  Supported  Eligible           Unselected
Index  description   limiters   rate type  ports              ports
-----  ------------  ---------  ---------  -----------------  -----------------
0      S-Series         4 orl   perc       None               None
       4 ORL                    pps
                                Kbps
                                Mbps
                                Gbps

1      S-Series        16 orl   perc       ge.4.1-48;         ge.4.1-48;
       16 ORL                   pps        ge.6.1-48;         ge.6.1-48;
                                Kbps       ge.8.1-48;         ge.8.1-48;
                                Mbps       ge.8.101-112       ge.8.101-112
                                Gbps
```

Flood Control

Flood Control is only supported on port-type 0. Three reference limiters are supported. Use the **show cos port-type flood-ctrl** command to display the port types and their associated ports.

The following example displays default values for the **show cos port-type flood-crtl** command output:

```
S Chassis(rw)->show cos port-type flood-ctrl

Number of resources:              Supported rate types:
 txq = transmit queue(s)           perc  = percentage
 irl = inbound rate limiter(s)     pps   = packets per second
 orl = outbound rate limiter(s)    Kbps  = kilobits per second
 fld = flood rate limiter(s)       Mbps  = megabits per second
                                   Gbps  = gigabits per second
                                   Tbps  = terabits per second

       Port type     Number of  Supported  Eligible           Unselected
Index  description   limiters   rate type  ports              ports
-----  ------------  ---------  ---------  -----------------  -----------------
0      S-Series         3 fld   perc       ge.4.1-48;         ge.4.1-48;
       Flood Ctrl               pps        ge.6.1-48;         ge.6.1-48;
                                Kbps       ge.8.1-48;         ge.8.1-48;
                                Mbps       ge.8.101-112       ge.8.101-112
                                Gbps
```

Configuring CoS Port Groups

CoS port groups provide for grouping ports by CoS feature configuration and port type. Ports are required to be configured by groups: this feature provides a meaningful way of identifying ports by similar functionality and port type.

Groups consist of a group number and port type and are numbered as such, *port-group.port-type*. For example: port group 0, port type 0 would be numbered port group **0.0**. Three default port groups exist for TxQ, IRL, ORL, and flood control CoS features and are identified as port group 0 and port type 0 or 1 and are indexed as **0.0** or **0.1** respectively for each feature. These default port groups cannot be removed and all physical ports in the system are assigned to one of the three port groups for each feature (remember group assignment is determined by port type).

Additional port groups, up to eleven total, may be created. Ports assigned to a new port group cannot belong to another non-default port group entry and must be comprised of the same port type as defined by the port group you are associating it with. The creation of additional port groups could be used to combine similar ports by their function for flexibility. For instance, ports associated to users can be added to a port group called Users and ports associated to uplink ports can be added to a port group called Uplink. Using these port groups, a class of service unique to each group can assign different rate limits to each port group. User ports can be assigned a rate limit configured in one CoS, while Uplink ports can be assigned a different rate limit configured in another CoS. A maximum of 8 port groups per CoS transmit queue and/or rate-limiter function are supported.

Port-Groups: TxQ Configuration

TxQ Port-Groups contain user settings for specific types of ports and their matching transmit queue settings. Port groups 0.0 through 0.1 exist by default. New port groups can be configured with a name and ports can be added according to device port-type. Transmit queue behavior can also be configured per port group; default port-groups are configured in strict priority queuing mode. Additional port groups also default to strict priority queuing mode, though each TxQ port group can be configured for weighted-fair queuing if desired.

The **show cos port-config txq** command displays all configured TxQ port-groups. Group name and type are displayed as well as ports associated with the port group. For **show cos port-config txq** output, arbiter mode (TxQ mode) is displayed along with a picture of the supported queues and the number of slices allotted to the group. Queuing is also displayed by percentage.

The following example displays default values for the **show cos port-config txq** command output:

```
S Chassis(rw)->show cos port-config txq

 * Percentage/queue (if any) are approximations based on
   [(slices/queue) / total number of slices]

Transmit Queue Port Configuration Entries
-------------------------------------------------------------------------
Port Group Name  :S-Series 11Q
Port Group       :0
Port Type        :0
Assigned Ports   :ge.2.1-48,101-112;ge.4.1-48;tg.4.201-204
Arbiter Mode     :Low Latency Queue
Slices/queue     :Q [0]: LLQ  Q [1]:   0  Q [2]:   0  Q [3]:   0
                 :Q [4]:   0  Q [5]:   0  Q [6]:   0  Q [7]:   0
                 :Q [8]: 100  Q [9]: LLQ  Q [10]: LLQ
Percentage/queue :Q [0]: LLQ  Q [1]:   0% Q [2]:   0% Q [3]:   0%
                 :Q [4]:   0% Q [5]:   0% Q [6]:   0% Q [7]:   0%
                 :Q [8]: 100% Q [9]: LLQ  Q [10]: LLQ
-------------------------------------------------------------------------
```

Additional port groups can be created using the **set cos port-config txq** command. Name and associated ports can be configured, as well as TxQ settings. You need to:

- Identify the port-group for configuration
- Optionally, specify port-group Name, associated ports, and arb-percentage or arb-slices

Port-Groups: IRL Configuration

IRL port-groups contain user settings for specific types of ports and their matching inbound rate limiting configurations. Port groups 0.0 through 0.1 exist by default. Each new group can be configured with a name and ports added to each group according to device port-type. Use the **show cos port-config irl** command to display each IRL port-group configured by group and type, with group name and associated ports.

The following example displays default values for the **show cos port-config irl** command output:

```
S Chassis(rw)->show cos port-config irl

Inbound Rate Limiting Port Configuration Entries
------------------------------------------------------------------------
Port Group Name  :S-Series 32 IRL
Port Group       :0
Port Type        :0
Assigned Ports   :ge.1.1-60
------------------------------------------------------------------------
Port Group Name  :S-Series 8 IRL
Port Group       :0
Port Type        :1
Assigned Ports   :none
------------------------------------------------------------------------
Port Group Name  :S-Series 24 IRL
Port Group       :0
Port Type        :2
Assigned Ports   :none
------------------------------------------------------------------------
```

Additional port groups can be created using the **set cos port-config irl** command. Port group name and associated ports can be configured. You need to:

- Identify the port-group for configuration
- Optionally, specify port-group Name and associated ports

Port-Groups: ORL Configuration

ORL port-groups contain user settings for specific types of ports and their matching outbound rate limiting configurations. Port groups 0.0 through 0.2 exist by default. Each new group can be configured with a name and ports added to each group according to device port-type. Use the **show cos port-config orl** command to display each ORL port-group configured by group and type, with group name and associated ports.

The following example displays default values for the **show cos port-config orl** command output:

```
S Chassis(rw)->show cos port-config orl
Outbound Rate Limiting Port Configuration Entries
------------------------------------------------------------------------
Port Group Name  :S-Series 48 ORL
Port Group       :0
```

```
Port Type         :0
Assigned Ports    :none
--------------------------------------------------------------------------
Port Group Name   :N/A 16 ORL
Port Group        :0
Port Type         :1
Assigned Ports    :none
--------------------------------------------------------------------------
Port Group Name   :N/A 4 ORL
Port Group        :0
Port Type         :2
Assigned Ports    :none
--------------------------------------------------------------------------
```

Additional port groups can be created using the **set cos port-config orl** command. Port group name and associated ports can be configured. You need to:

- Identify the port-group for configuration
- Optionally, specify port-group Name and associated ports

Port-Groups: Flood Control Configuration

CoS-based flood control prevents configured ports from being disrupted by a traffic storm by rate limiting specific types of packets through those ports. When flood control is enabled on a port, incoming traffic is monitored over one second intervals. During an interval, the incoming traffic rate for each configured traffic type (unicast, broadcast, or multicast) is compared with the configured traffic flood control rate, specified in packets per second. If, during a one second interval, the incoming traffic of a configured type reaches the traffic flood control rate configured on the port, CoS-based flood control drops the traffic until the interval ends. Packets are then allowed to flow again until the limit is again reached.

Flood control port-groups contain user settings for specific types of ports and their matching flood limiting configurations. Port groups 0.0 through 0.1 exist by default. Each new group can be configured with a name and ports added to each group according to device port-type. Use the **show cos port-config flood-ctrl** command to display each flood control port-group configured by group and type, with group name and associated ports.

The following example displays default values for the **show cos port-config flood-ctrl** command output:

```
S Chassis(rw)->show cos port-config flood-ctrl

Flood Rate Limiting Port Configuration Entries
--------------------------------------------------------------------------
Port Group Name   :S-Series Flood Ctrl
Port Group        :0
Port Type         :0
Assigned Ports    :ge.1.1-48;tg.1.1-4
--------------------------------------------------------------------------
```

Additional port groups can be created using the **set cos port-config flood-ctrl** command. Port group name and associated ports can be configured. You need to:

- Identify the port-group for configuration
- Optionally, specify port-group Name and associated ports

Configuring CoS Port-Resource

Physical rate limiters and rate shapers are configured in CoS port resources. Resources map directly to the number of queues and rate limiters supported by each port-type. Remember, group 0.0 supports 16 TxQ resources and 32 IRL resources while group 0.1 supports 4 TxQ resources and 8 IRL resources. Resources exist for each port group and are indexed as *port-group.port-type resource-index*. Port-resources initially default to none, as rate limiting and shaping is not required.

CoS TxQ Port-Resource (Outbound Rate Shapers)

Rate shaping throttles the rate at which queues transmit packets. See "Rate Shaping" on page 53-9 for a general discussion of rate shaping. Rate shaping is TCP friendly; it buffers packets that are above the rate rather than drop them. CoS rate shaping allows you to configure rate shapers based on a unit rate (kilobits/second, megabits/second, gigabits/second), or a percentage of the port's line speed.

The **show cos port-resource txq** command displays resources for each port group created along with the resource index (physical queue). By default, no resources are configured for TxQ port-resources. Rates displayed as none indicate no resources exist. The default Rate Shaping algorithm is tail-drop and is not configurable.

The following example displays default values for the **show cos port-resource txq** command output:

```
S Chassis(rw)->show cos port-resource txq

Group Index Resource Type Unit    Rate      Algorithm
----------- -------- ---- ---- ----------   ---------
0.0         0        txq  perc none         tail-drop
0.0         1        txq  perc none         tail-drop
0.0         2        txq  perc none         tail-drop
0.0         3        txq  perc none         tail-drop
0.0         4        txq  perc none         tail-drop
0.0         5        txq  perc none         tail-drop
0.0         6        txq  perc none         tail-drop
0.0         7        txq  perc none         tail-drop
0.0         8        txq  perc none         tail-drop
0.0         9        txq  perc none         tail-drop
0.0         10       txq  perc none         tail-drop
0.0         11       txq  perc none         tail-drop
0.0         12       txq  perc none         tail-drop
0.0         13       txq  perc none         tail-drop
0.0         14       txq  perc none         tail-drop
0.0         15       txq  perc none         tail-drop
0.1         0        txq  perc none         tail-drop
0.1         1        txq  perc none         tail-drop
0.1         2        txq  perc none         tail-drop
```

The **set cos port-resource txq** command is used for creating outbound rate shapers. You need to:

- Identify the port group for configuration
- Identify the queue resource ID, along with unit and rate desired for that queue

CoS IRL Port-Resource (Inbound Rate Limiter)

Unlike rate shaping, inbound rate limiting or rate policing simply drops or clips traffic inbound if a configured rate is exceeded. See "Rate Limiting" on page 53-8 for a general discussion of rate limiting. CoS inbound rate limiting allows you to configure rate limits based on a unit rate (kilobits/second, megabits/second, gigabits/second), or percentage of the port's line speed. The IRL

port-resource configuration allows you to enable sending syslog messages or traps once a rate limit is exceeded, as well as to disable the port.

The **show cos port-resource irl** command displays resources for each port group created along with the index, as described above. By default, no resources are configured for IRL port-resources. Rates displayed as none indicate no resources exist. The default Rate Limiting algorithm is tail-drop. The Action field in the display indicates user-desired action for each syslog, trap, and port disable behavior when configured.

The following example displays default values for the **show cos port-resource irl** command output:

```
S Chassis(rw)->show cos port-resource irl

'?' after the rate value indicates an invalid rate value

Group Index Resource Type Unit    Rate      Rate Limit Type Action
----------- -------- ---- ---- ----------   --------------- ------
0.0         0        irl  perc none         drop            none
0.0         1        irl  perc none         drop            none
0.0         2        irl  perc none         drop            none
0.0         3        irl  perc none         drop            none
0.0         4        irl  perc none         drop            none
0.0         5        irl  perc none         drop            none
0.0         6        irl  perc none         drop            none
0.0         7        irl  perc none         drop            none
0.0         8        irl  perc none         drop            none
.
.
.
0.2         20       irl  perc none         drop            none
0.2         21       irl  perc none         drop            none
0.2         22       irl  perc none         drop            none
0.2         23       irl  perc none         drop            none

No violators exist for this/these irl(s)
```

The **set cos port-resource irl** command is used for creating inbound rate limiters. You need to:

- Identify the port group for configuration
- Identify the limiter resource ID, along with desired unit, rate, and actions

CoS ORL Port-Resource (Outbound Rate Limiter)

Outbound rate limiting or rate policing simply drops or clips outbound traffic if a configured rate is exceeded. See “Rate Limiting” on page 53-8 for a general discussion of rate limiting. CoS outbound rate limiting allows you to configure rate limits based on a unit rate (kilobits/second, megabits/second, gigabits/second), or percentage of the port’s line speed. The ORL port-resource configuration allows you to enable sending syslog messages or traps once a rate limit is exceeded, as well as to disable the port.

The **show cos port-resource orl** command displays resources for each port group created along with the index, as described above. By default, no resources are configured for ORL port-resources. Rates displayed as none indicate no resources exist. The default rate limiting algorithm is tail-drop. The action field in the display indicates user-desired action for each syslog, trap, and port disable behavior when configured.

The following example displays default values for the **show cos port-resource orl** command output:

```
S Chassis(rw)->show cos port-resource orl
```

```
'?' after the rate value indicates an invalid rate value

Group Index Resource Type Unit    Rate      Rate Limit Type Action
----------- -------- ---- ---- ----------   --------------- ------
0.0         0        orl  perc none         drop            none
0.0         1        orl  perc none         drop            none
0.0         2        orl  perc none         drop            none
0.0         3        orl  perc none         drop            none
0.0         4        orl  perc none         drop            none
0.0         5        orl  perc none         drop            none
0.0         6        orl  perc none         drop            none
0.0         7        orl  perc none         drop            none
.
.
.
0.2         20       orl  perc none         drop            none
0.2         21       orl  perc none         drop            none
0.2         22       orl  perc none         drop            none
0.2         23       orl  perc none         drop            none

No violators exist for this/these orl(s)
```

The **set cos port-resource orl** command is used for creating outbound rate limiters. You need to:

- Identify the port group for configuration
- Identify the limiter resource ID, along with desired unit, rate, and actions

CoS Flood Control Port-Resource (Flood Limiter)

Flood control limiting prevents configured ports from being disrupted by a traffic storm by rate limiting configured traffic types such as multicast or broadcast through those ports. CoS flood limiting allows you to configure traffic type limiting based on a unit rate (kilobits/second, megabits/second, gigabits/second), or percentage of the port's line speed. The flood control port-resource configuration allows you to enable sending syslog messages or traps once a rate limit is exceeded, as well as to disable the port.

The **show cos port-resource flood-ctrl** command displays resources for each port group created along with the index, as described above. By default, no traffic type is configured for flood control port-resources. Rates displayed as none indicate no resources exist. The default rate limiting algorithm is tail-drop. The action field in the display indicates user-desired action for each syslog, trap, and port disable behavior when configured.

The following example displays default values for the **show cos port-resource flood-ctrl** command output:

```
S Chassis(rw)->show cos port-resource flood-ctrl

'?' after the rate value indicates an invalid rate value

Group Index Resource Type Unit    Rate      Rate Limit Type Action
----------- -------- ---- ---- ----------   --------------- ------
0.0         0        fld  perc none                         none
0.0         1        fld  perc none                         none
0.0         2        fld  perc none                         none
0.0         3        fld  perc none                         none
```

Configure a CoS flood control resource entry, by mapping a port group with a traffic type such as multicast or broadcast, along with the ability to optionally set syslog, trap, and/or disable port behaviors should the limit be exceeded. This index is used by the rate-limit option when setting a flood control cos reference

The **set cos port-resource flood-ctrl** command is used for configuring a CoS flood control resource entry.

Configuring CoS Reference Mapping

The CoS Reference Table maps the TxQ, ORL, and IRL references, defined by you and configured in the CoS Settings Table, to physical queues and rate limiters created in the port-resource table. A CoS reference table exists for each port group. The CoS reference table indexes can be thought of as virtual queues or rate limiters. The table accounts for the maximum number of queues and rate limiters supported by the device. The virtual queues and limiters map to the physical queues and rate limiters. The TxQ reference table is populated by default, because queues are required for all forwarding. The TxQ reference maps each reference value to a physical queue. The IRL Reference Table is not configured by default, because inbound rate limiting is optional.

CoS TxQ Reference Mapping

The CoS TxQ reference table uses 16 indexes or virtual queues, and maps each to a physical queue or resource. A TxQ reference table exists for each port group configured and is indexed similarly to port-resources, as *port-group.port-type reference*. For port-types with 16 queues, the 16-txq reference indexes map directly to the 16 physical queues. For port-types with 4 queues, the 16-txq reference indexes map:

- virtual queues 12-15 to physical queue **3**
- virtual queues 8-11 map to physical queue **2**
- virtual queues 4-7 map to physical queue **1**
- virtual queues 0-3 map to physical queue **0**

The TxQ reference table can be displayed using the **show cos reference txq** command and displays port-group, reference index, and physical queue.

The following example displays default values for the **show cos reference txq** command output:

```
S Chassis(rw)->show cos reference txq

Group Index Reference Type    Queue
----------- --------- ---- ------------
0.0         0         txq  0
0.0         1         txq  1
0.0         2         txq  2
0.0         3         txq  3
0.0         4         txq  4
0.0         5         txq  5
0.0         6         txq  6
0.0         7         txq  7
.
.
.
0.2         10        txq  5
0.2         11        txq  5
0.2         12        txq  6
0.2         13        txq  6
0.2         14        txq  7
0.2         15        txq  7
```

Although the TxQ reference table is populated by default, the Queue-to-Reference mapping can be configured using the **set cos reference txq** command. You need to:

- Identify the port group for configuration

- Identify the transmit queue reference, along with the associated queue

CoS IRL Reference Mapping Table

The CoS IRL reference table uses 32 indexes or virtual rate limiters, and maps each virtual limiter to a physical limiter or resource. An IRL reference table exists for each port group configured, and is indexed similarly to port-resources, as *port-group.port-type reference*. Because it is an optional configuration, IRL references are not populated with limiters (resources), but can be configured by you. The IRL reference table can be displayed using the **show cos reference irl** command.

The following example displays default values for the **show cos reference irl** command output:

```
S Chassis(rw)->show cos reference irl

Group Index Reference Type Rate Limiter
----------- --------- ---- ------------
0.0         0         irl  none
0.0         1         irl  none
0.0         2         irl  none
0.0         3         irl  none
0.0         4         irl  none
0.0         5         irl  none
0.0         6         irl  none
0.0         7         irl  none
0.0         8         irl  none
.
.
.
0.2         28        irl  none
0.2         29        irl  none
0.2         30        irl  none
0.2         31        irl  none
```

Physical-Limiter to reference mapping can be configured using the **set cos reference irl** command. The other references not configured are indicated by rate limiter “none”. To configure a physical limiter to reference mapping, you need to:

- Identify the port group for configuration
- Identify the rate-limit reference

CoS ORL Reference Mapping Table

The CoS ORL reference table uses 48 indexes or virtual rate limiters, and maps each virtual limiter to a physical limiter or resource. An ORL reference table exists for each port group configured, and is indexed similarly to port-resources, as *port-group.port-type reference*. Because it is an optional configuration, ORL references are not populated with limiters (resources), but can be configured by you. The ORL reference table can be displayed using the **show cos reference orl** command.

The following example displays default values for the **show cos reference orl** for port group 0.0 command output:

```
S Chassis(rw)->show cos reference orl 0.0

Group Index Reference Type Rate Limiter
----------- --------- ---- ------------
0.0         0         orl  none
0.0         1         orl  none
0.0         2         orl  none
0.0         3         orl  none
0.0         4         orl  none
0.0         5         orl  none
```

```
0.0          6          orl  none
0.0          7          orl  none
.
.
.
0.0          44         orl  none
0.0          45         orl  none
0.0          46         orl  none
0.0          47         orl  none
```

Physical-Limiter to reference mapping can be configured using the **set cos reference orl** command. The other references not configured are indicated by rate limiter "none". To configure a physical limiter to reference mapping, you need to:

- Identify the port group for configuration
- Identify the rate-limit reference

Configuring the CoS Index

The CoS settings table assigns a priority, a ToS value, TxQ reference table and an IRL reference to a CoS entry as follows:

CoS Index - Indexes are unique IDs for each CoS settings table entry. CoS indexes 0–7 are created by default and mapped directly to an 802.1p priority values 0–7 for backwards compatibility. These entries cannot be removed and the 802.1p value cannot be changed. When CoS is enabled using the **set cos state enable** command, indexes are assigned. Entries 0–255 are configurable for a total of 256 CoS entries.

Priority: For each new CoS index created, you have the option to assign an 802.1p priority value 0-7 for the class of service. CoS indexes 0-7 map directly to 802.1p priorities and cannot be changed as they exist for backward compatibility. All other CoS index entries can have a priority value set between 0 and 7.

ToS: The IP header Type of Service field is an 8-bit field also referred to as the DiffServ Code Point (DSCP) field. This optional value can be set per class of service to a value between 0–255. When a frame is assigned to a class of service for which this value is configured, the ToS field of the incoming IP packet will be overwritten to values defined by you. This ToS rewrite option also allows masking. The ToS can selectively mask (change) certain bits of the field, without changing others. For instance, masking the ToS could be used to modify the ToS precedence without modifying the DTR/ECN bits. The mask specified contains the bits to be changed. CLI input can be in decimal or hex value, and a mask is not required. If the mask is not specified in the ToS input, all bits will be overwritten. ToS can be set for CoS indexes 0-7.

TxQ Reference: Because all traffic requires association to a transmit queue, the CoS TxQ reference field will always be populated when a new CoS index is created. If a TxQ reference value is not chosen, TxQ reference 0 will be assigned. The reference does not indicate the actual transmit queue to be assigned by CoS; it points to the CoS TxQ reference mapping table index entry. It may be thought of as the virtual queue that is associated to a physical queue defined by the TxQ reference mapping table. TxQ reference mapping table defines 16 TxQ references, therefore CLI input for TxQ reference in the CoS Settings Table is 0-15. See "CoS TxQ Reference Mapping" on page 53-19 for a TxQ reference configuration discussion.

IRL Reference: The CoS IRL reference field is optional, as rate limits are not required. Like the TxQ reference field, the IRL reference does not assign an inbound rate limit but points to the CoS IRL Reference Mapping Table. This reference may also be thought of as the virtual rate limiter that will assign the physical rate limiter defined by the IRL Reference Mapping Table. The IRL Reference Mapping Table defines 32 IRL references, therefore input for IRL reference in the CoS

Settings Table is 0-31. See “CoS IRL Reference Mapping Table” on page 53-20 for an IRL reference configuration discussion.

ORL Reference: The CoS ORL reference field is optional, as rate limits are not required. Like the TxQ and IRL reference fields, the ORL reference does not assign an outbound rate limit but points to the CoS ORL Reference Mapping Table. This reference may also be thought of as the virtual rate limiter that will assign the physical rate limiter defined by the ORL Reference Mapping Table. The IRL Reference Mapping Table defines 48 ORL references, therefore input for ORL reference in the CoS Settings Table is 0 - 47. See “CoS ORL Reference Mapping Table” on page 53-20 for an ORL reference configuration discussion.

Drop-Precedence Reference: Drop-Precedence indicates a preference for dropping packets, often used in association with Weighted Random Early Detection (WRED) queues. The S-Series implementation uses the values to prioritize packets. Drop precedence has a special meaning within a Flex-Edge context. Packets assigned a drop-precedence value are assigned a 4th level of priority in the Flex-Edge mechanism, and are limited to rules applied to a single port. See “Flex-Edge” on page 53-2 for a detailed Flex-Edge drop-precedence discussion.

Flood Control Reference: The CoS flood control reference field is optional. Flood control limiting is not required. Enable or disable flood control for the specified CoS index.

New CoS Indexes can be created using the **set cos settings** command. ToS, 802.1p priority, TxQ reference, and IRL Reference can be configured for each CoS Index. You need to:

- Enter a CoS Index value from 0–255
- Specify 802.1p priority (Index entries 8–255 only), tos-value, txq-reference and irl-reference

Use the **set cos settings** command to create or modify an already existing CoS index.

Use the **show cos settings** command to display current CoS indexes.

The following example displays default values for the **show cos settings** command output:

```
S Chassis(rw)->show cos settings
 * Means attribute has not been configured

CoS Index  Priority    ToS      TxQ    IRL    ORL    Drop Prec Flood-Ctrl
---------  ----------  -------  -----  -----  -----  --------- ----------
0          0           *        0      *      *      *         Disabled
1          1            32.0    4      *      *      *         Disabled
2          2            64.0    8      *      *      *         Disabled
3          3            96.0    12     *      *      *         Disabled
4          4           128.0    16     *      *      *         Disabled
5          5           *        10     11     *      *         Disabled
6          6           *        12     *      *      *         Disabled
7          7           *        14     *      *      *         Disabled
```

Enabling CoS State

CoS state is a global setting that must be enabled for CoS configurations to be applied to a port. When CoS state is enabled, controls configured for CoS supersede port level controls for priority queue mapping, IRL, and TxQ. These port level settings can be configured independent of CoS state, but will have no affect while CoS is enabled. Disabling CoS results in the restoration of current port level settings.

Use the **set cos state enable** command to enable CoS state globally for this system.

Use the **set cos state disable** command to disable CoS state globally for this system.

Use the **show cos state** command to display the current status of CoS state.

Displaying CoS Violations

CoS violations can be displayed per physical rate limit for IRL, ORL, and flood control to show you when an rate limit has been violated. Use the **show cos violation** command to display ports that have a limiter violated as well as any ports that may be disabled by the limiter.

The following example displays default values for the **show cos violation irl** command output:

```
S Chassis(rw)->show cos violation irl ge.1.1:*

              Rate-Limiter        Rate-Limiter  Rate-Limiter
Port          Index         Type  Status        Counter
------------  ------------  ----  ------------  --------------------
ge.1.1        0             irl   not-violated  0
ge.1.1        1             irl   not-violated  0
ge.1.1        2             irl   not-violated  0
ge.1.1        3             irl   not-violated  0
ge.1.1        4             irl   not-violated  0
ge.1.1        5             irl   not-violated  0
ge.1.1        6             irl   not-violated  0
ge.1.1        7             irl   not-violated  0
ge.1.1        8             irl   not-violated  0
ge.1.1        9             irl   not-violated  0
ge.1.1        10            irl   not-violated  0
...
ge.1.1        29            irl   not-violated  0
ge.1.1        30            irl   not-violated  0
ge.1.1        31            irl   not-violated  0
```

Violations are also displayed by resource and port using the **show cos port-resource** command. Violating ports are displayed at the end of the resource table.

The QoS CLI Command Flow

Procedure 53-1 provides a CLI flow summary of each step in the configuration flow along with the show commands to verify the configuration.

Procedure 53-1 Class of Service CLI Configuration Command Summary

Step	Task	Command(s)
1.	Inspect the TxQs, IRL, ORL, and flow control support for the installed ports. This information is used to determine the module port type for port group.	**show cos port-type txq** **show cos port-type irl** **show cos port-type orl** **show cos port-type flood-ctrl**
2.	Set the CoS transmit queue port group configuration by mapping a physical port list to a port group for purposes of TxQ configuration. Optionally associate a name and the configuration of a TxQ weighted fair queue behavior configuration. Verify the new configuration.	**set cos port-config txq** *group-type-index* [**name** *name*] [**ports** *port-list*] [**append**] | [**clear**] [**arb-slice** *slice-list*] [**arb-percentage** *percentage-list*] [**enhanced-groups** *group-id*] [**enhanced-percentage** *bandwidth*] **show cos port-config txq** *port_group.port_type*

Procedure 53-1 Class of Service CLI Configuration Command Summary (continued)

Step	Task	Command(s)
3.	Set the CoS inbound rate-limit port group configuration by mapping a physical port list to a port group for purposes of IRL configuration, optionally allowing the association of a name for this configuration. Verify the new configuration.	**set cos port-config irl** *port_group.port_type* **name** *name* **ports** *ports_list* **show cos port-config irl**
4.	Set the CoS outbound rate-limit port group configuration by mapping a physical port list to a port group for purposes of ORL configuration, optionally allowing the association of a name for this configuration. Verify the new configuration.	**set cos port-config orl** *port_group.port_type* **name** *name* **ports** *ports_list* **show cos port-config orl**
5.	Set the CoS flood control limit port group configuration by mapping a physical port list to a port group for purposes of flood control configuration, optionally allowing the association of a name for this configuration. Verify the new configuration.	**set cos port-config flood-ctrl** *port_group.port_type* **name** *name* **ports** *ports_list* **show cos port-config flood-ctrl**
6.	Configure a Class of Service transmit queue port resource entry, by mapping a port group with a transmit queue and applying a TxQ rate shaping value to the mapping. Verify configuration changes.	**set cos port-resource txq** *port_group.port_type tx_queue* **unit** *unit* **rate** *rate* **show cos port-resource txq** *port_group.port_type*
7.	Configure a CoS inbound rate limiting index entry, by mapping a port group with a rate-limit value, along with the ability to optionally set syslog, trap, and/or disable port behaviors should the limit be exceeded. This index is used by the rate-limit option when setting an IRL cos reference.	**set cos port-resource irl** *port_group.port_type index* **unit** *unit* **rate** *rate* **syslog** *setting* **trap** *setting* **disable-port** *setting* **show cos port-resource irl** *port_group.port_type*
8.	Configure a CoS outbound rate limiting index entry, by mapping a port group with a rate-limit value, along with the ability to optionally set syslog, trap, and/or disable port behaviors should the limit be exceeded. This index is used by the rate-limit option when setting an ORL cos reference.	**set cos port-resource orl** *port_group.port_type index* **unit** *unit* **rate** *rate* **syslog** *setting* **trap** *setting* **disable-port** *setting* **show cos port-resource orl** *port_group.port_type*
9.	Configure a CoS flood control index entry, by mapping a port group with a traffic type such as multicast or broadcast, along with the ability to optionally set syslog, trap, and/or disable port behaviors should the limit be exceeded. This index is used by the rate-limit option when setting a flood control cos reference.	**set cos port-resource flood-ctrl** *port_group.port_type traffic-type* **unit** *unit* **rate** *rate* **syslog** *setting* **trap** *setting* **disable-port** *setting* **show cos port-resource flood-ctrl** *port_group.port_type*
10.	Set a CoS transmit queue reference configuration, by mapping a port group to a queue resource ID and associating the mapping with a transmit reference. Verify the new CoS reference configuration.	**set cos reference txq** *port_group.port_type reference* **queue** *queue* **show cos reference txq** *port_group.port_type*

Procedure 53-1 Class of Service CLI Configuration Command Summary (continued)

Step	Task	Command(s)
11.	Set a CoS inbound rate limiting reference configuration, by mapping a port group with a rate limiter resource ID and associating the mapping with an IRL reference. Verify the new CoS reference configuration.	**set cos reference irl** *port_group.port_type reference* **rate-limit** *IRLreference* **show cos reference irl** *port_group.port_type*
12.	Set a CoS outbound rate limiting reference configuration, by mapping a port group with a rate limiter resource ID and associating the mapping with an ORL reference. Verify the new CoS reference configuration.	**set cos reference orl** *port_group.port_type reference* **rate-limit** *IRLreference* **show cos reference orl** *port_group.port_type*
13.	Modify a currently configured CoS or create a new CoS. Verify the new CoS configuration. All TxQ to port group mappings are associated with the transmit queue reference. All IRL to port group mappings are associated with the inbound rate limiter reference.	**set cos settings** *cos-list* [**priority** *priority*] [**tos-value** *tos-value*] [**txq-reference** *txq-reference*] [**irl-reference** *irl-reference*] [**orl-reference** *orl-reference*] [**drop-precedence** *drop-precedence*] [**flood-ctrl** *flood-ctrl*] **show cos settings**
14.	Enable CoS state for the system. Verify the new CoS state.	**set cos state enable** **show cos state**

QoS Configuration Example

In our example, an organization's network administrator needs to assure that VoIP traffic, both originating in and transiting the network of S-Series edge switches and a S-Series core router, is configured for QoS with appropriate priority, ToS, and queue treatment. We will also rate limit the VoIP traffic at the edge to 1024 Kbps to guard against DOS attacks, VoIP traffic into the core at 25 Mbps, and H.323 call setup at 5 pps. Data traffic retains the default configuration.

This example places QoS configuration within a policy context. Policy is not required to configure QoS.

This example assumes CEP authentication using H.323 for VoIP. If you are not authenticating your VoIP end point with CEP H.323 authentication, you will need to adjust the VoIP policy accordingly. For instance, SIP uses UDP port 5060, not the TCP port 1720.

Notes: Extreme Networks highly recommends that you use the NetSight Policy Manager to configure QoS on your network, whether you are applying policy or not. This example discusses the QoS configuration using Policy Manager followed by CLI input summaries.

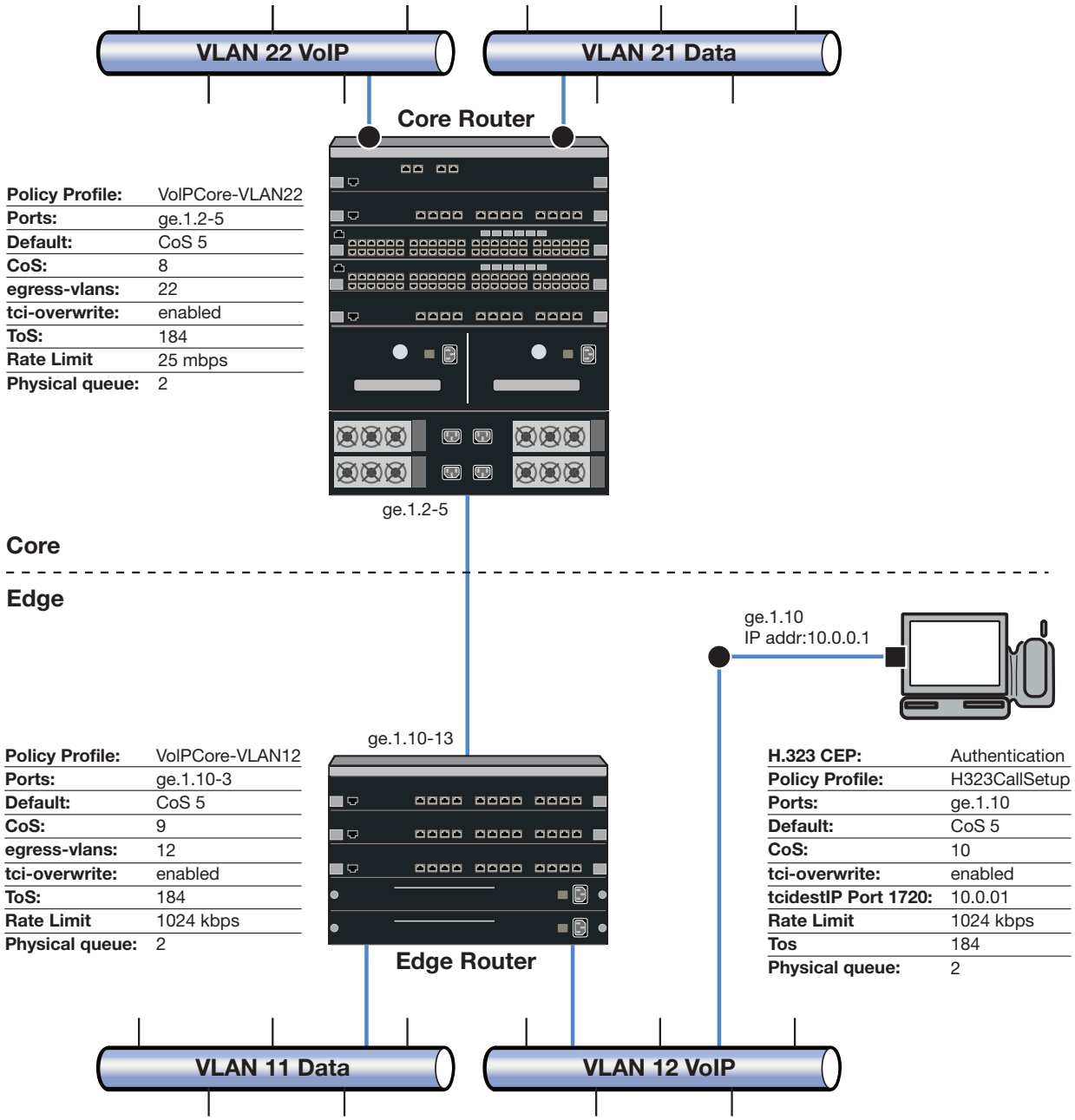
To simplify this discussion of the configuration process, this example is limited to the VoIP configuration context. Table 53-1 provides a set of sample values for priority, IRL, and transmit queue across a number of real world traffic types. This table can be used as an aid in thinking about how you might want to apply CoS across your network. Note that scavenger class is traffic that should be treated as less than best effort: external web traffic, for instance.

Table 53-1 CoS Sample Values By Traffic Type

Name	Priority	IRL		Transmit Queue					
				Queue #		Shaping		WFQ	
		Edge	Core	Edge	Core	Edge	Core	Edge	Core
Loop Detect	0	10 PPS	10 PPS	0	0	10%		5%	5%
Scavenger	0	15 Mbps							
Best Effort	1			1	1	80%		45%	45%
Bulk Data	2								
Critical Data	3								
Network Control	4	40 PPS	1 Mbps	2	2	1Mbps		25%	25%
Network Management	5	2 Mbps							
RTP	6	1 Mbps	25 Mbps	3	3			25%	25%
Voice/Video	7								

Figure 53-7 displays the network setup for this example configuration, with the desired Profile/QoS summary for each network node. Each node is configured with VoIP and Data VLANs. Each VoIP VLAN contains four 1-gigabit interfaces for each node.

Figure 53-7 QoS Configuration Example



A core profile for the router and an edge profile for the switch provide for the difference in rate limiting needs between the enduser and aggregation devices. A call setup profile provides rate limiting for the setup aspect of the VoIP call. Each edge and core VLAN profile will be configured for default CoS 5 (best default priority for voice and video), the addition of its associated VLAN to its egress VLAN list, and ToS overwrite. We will create a separate CoS for both the edge and core to handle ToS, rate-limit and queue configuration for these devices.

The H.323 call setup profile will be configured so that TCP call setup traffic on the TCP destination port 1720:10.0.0.1 of its gigabit link will be configured for the proper rate limit on that port.

Using NetSight Policy Manager, configure the policy roles and related services as follows:

Setting the VoIP Core Policy Profile (Router 1)

For S-Series router 1, we configure a separate policy for VoIP Core. VoIP Core policy deals with packets transiting the core network using VoIP VLAN 22. For role VoIPCore we will:

- Configure VoIPEdge-VLAN22 as the name of the role.
- Set default CoS to 5.
- Set the default access control to VLAN 22.
- Enable TCI overwrite so that ToS will be rewritten for this policy.

Create a Policy Service

- Name the service VoIPCore Service.
- Apply the service to the VoIPCore Policy Role.

Create a Rate-limiter

Create a rate-limit as follows:

- Inbound rate-limit of 25 mbps
- Apply it to port group types 32/8/100 for index 0

Create Class of Service for VoIPEdge Policy

Create CoS 8 as follows:

- 802.1p priority: 5
- ToS: B8
- Specify IRL index 0 to associate this CoS to the rate limit

Create a Rule

- Create a Layer 2 traffic classification rule for VLAN ID 22 within the VoIPCore service.
- Associate CoS 8 as the action for the rule.

Setting the VoIP Edge Policy Profile (Switch 1)

For S-Series Switch 1, we configure a separate policy for VoIP edge. VoIP edge policy deals with packets transiting the edge network using VoIP VLAN 12 with edge access. For role VoIPEdge we will:

- Configure VoIPEdge-VLAN12 as the name of the role.
- Set default CoS to 5.
- Set the default access control to VLAN 22.
- Enable TCI overwrite so that ToS will be rewritten for this policy.

Create a Policy Service

- Name the service VoIPEdge Service.
- Apply the service to the VoIPEdge Policy Role.

Create a Rate-limiter

Create a rate-limit as follows:

- Inbound rate-limit of 1 mbps
- Apply it to port group types 32/8/100 for index 0

Create Class of Service for VoIPEdge Policy

Create CoS 9 as follows:

- 802.1p priority: **5**
- ToS: **B8**
- Specify IRL index 0 to associate this CoS to the rate limit

Create a Rule

- Create a Layer 2 traffic classification rule for VLAN ID 22 within the VoIPEdge service.
- Associate CoS 9 as the action for the rule.

Setting the H.323 Call Setup Policy Profile

H.323 Call Setup policy deals with the call setup traffic for VoIP H.323 authenticated users directly attached to Switch 1 using link ge.1.10. For role H.323 Call Setup we will:

- Configure H323CallSetup as the name of the role.
- Set default CoS to 5.
- Enable TCI overwrite so that ToS will be rewritten for this policy.

Create a Policy Service

- Name the service H323CallSetup Service.
- Apply the service to the H323CallSetup Policy Role.

Create a Rate-limiter

Create a rate-limit as follows:

- Inbound rate-limit of 5 pps
- Apply it to port group types 32/8/100 for index 1

Create Class of Service for H323CallSetup Policy

Create CoS 10 as follows:

- 802.1p priority: 5
- ToS: B8
- Specify IRL index 1 to associate this CoS to the rate limit

Create a Traffic Classification Layer Rule

Create a transport layer 3 rule as follows:

- Traffic Classification Type: IP TCP Port Destination
- Enter in Single Value field: 1720 (TCP Port ID)
- For IP TCP Port Destination value: 10.0.0.1 with a mask of 255.255.255.255
- Associate CoS 10 as the action for the rule

Applying Role and Associated Services to Network Nodes

Once you have created your roles and associated the appropriate services to them, you must apply the appropriate role(s) to the network nodes as follows:

Router 1

The policy role creation discussed above is appropriate for Router 1 as follows:

- Apply role VoIPCore-VLAN22 to ports ge.1.2-5.

Switch 1

VoIPEdge and H323CallSetup roles are applied to Switch 1 as follows:

- Apply role VoIPEdge-VLAN12 to ports ge.1.10-13.
- Apply role H323CallSetup to port ge.1.10

CLI Summaries for This QoS Configuration

This QoS configuration can be input from the CLI using the following entries:

Summary of Command Line Input for S-Series Router 1

```
S Chassis(rw)->set policy profile 1 name VoIPCore-VLAN22 cos 5 egress-vlans 22
tci-overwrite enable

S Chassis(rw)->set policy rule admin-profile vlantag 22 mask 12 port-string
ge.1.2-5 admin-pid 1

S Chassis(rw)->set policy rule 1 vlantag 22 mask 12 vlan 22 cos 8

S Chassis(rw)->set cos port-resource irl 1.1 0 unit mbps rate 25

S Chassis(rw)->set cos reference irl 1.1 8 rate-limit 0

S Chassis(rw)->set cos 8 priority 5 tos-value 184.0 txq-reference 8 irl-reference
0

S Chassis(rw)->set cos state enable
```

Summary of Command Line Input for S-Series Switch 1

```
S Chassis(rw)->set policy profile 1 name VoIPEdge-VLAN12 cos 5 egress-vlans 12
tci-overwrite enable

S Chassis(rw)->set policy rule admin-profile vlantag 12 mask 12 port-string
ge.1.10-13 admin-pid 1

S Chassis(rw)->set policy rule 1 vlantag 12 mask 12 vlan 12 cos 9

S Chassis(rw)->set cos port-resource irl 2.1 0 unit mbps rate 1

S Chassis(rw)->set cos reference irl 2.1 9 rate-limit 0
```

```
S Chassis(rw)->set cos 9 priority 5 tos-value 184.0 txq-reference 8 irl-reference
1
S Chassis(rw)->set policy profile 2 name H323CallSetup cos 5 tci-overwrite enable
S Chassis(rw)->set policy rule admin-profile port ge.1.10 mask 16 port-string
ge.1.10 admin-pid 2
S Chassis(rw)->set policy rule 1 tcpdestportIP 1720:10.0.0.1 cos 10 port-string
ge.1.10
S Chassis(rw)->set cos port-resource irl 3.1 2 unit pps rate 5
S Chassis(rw)->set cos reference irl 3.1 10 rate-limit 1
S Chassis(rw)->set cos 10 priority 5 tos-value 184.0 txq-reference 8 irl-reference
2
S Chassis(rw)->set cos state enable
```

Terms and Definitions

Table 53-2 lists terms and definitions used in this Quality of Service configuration discussion.

Table 53-2 Quality of Service Configuration Terms and Definitions

Term	Definition
Class of Service (CoS)	The grouping of priority and forwarding behaviors that collectively determine packet bandwidth behavior as it transits the link, including: 802.1p, IP ToS rewrite, priority Transmit Queue (TxQ), Inbound and/or outbound Rate Limiter (IRL) and outbound rate shaper.
DSCP	Differentiated Services Code Point. The lower 6 bits of the ToS field defined by RFC 2474.
Flows	In a QoS context, a sequence of IP packets that share a common class of service and forwarding treatment as they transit the interface.
Forwarding Treatment	Queue behavior during the packet egress stage (strict priority, weighted fair, hybrid).
Jitter	The change in a flow's packet spacing on the link due to the bursty and congestive nature of the IP network. This irregular spacing - jitter - can severely degrade the quality of voice calls or multimedia presentations.
Port Group	The grouping of ports based upon the same CoS features and port type.
Port Type	The differentiation of ports based upon TxQ, IRL, ORL, and flood control resource capabilities.
Priority	The preference of one packet (classification) or queue (packet forwarding) over another.
Quality of Service (QoS)	A bandwidth management mechanism able to preferentially treat packets based upon packet classification and forwarding treatment.
Rate Limiting	The bounding of bandwidth used by a QoS packet flow such that excess packets are dropped/clipped.
Rate Shaping	The rescheduling of bursty traffic while in the queue based upon packet buffering such that traffic beyond the configured bandwidth threshold is delayed until bandwidth usage falls below the configured threshold.
Type of Service (ToS)	An 8-bit field defined by RFC 1349 used for the prioritization of packets within a QoS context.

54

Anti-Spoofing Configuration

This chapter describes the anti-spoofing features and how to configure them.

Anti-Spoofing Feature Overview

Attacks on IP networks can easily be performed using readily available tools found on the internet today. Malicious users can spoof DHCP server response packets, allowing them to give false information to a user for such fields as the default gateway or domain name resolution servers. Man in the middle attacks can take advantage of ARP, allowing a hacker to redirect user traffic through his own device to and from the default gateway. The hacker can then spy on the private information being sent from the user, without either the user or gateway knowing. A malicious user can spoof an innocent user's IP address, allowing the malicious user to bypass other possible security features of a network that are based on a user's subnet.

The Extreme Networks anti-spoofing solution provides a flexible and secure approach to IP spoofing detection and prevention. To mitigate the effects of these types of attacks on a network, a source MAC to source IP address binding table is created. The three basic tools used to detect source IP to source MAC address associations, based on the entries in the binding table, and take action on violations are:

- DHCP snooping,
- Dynamic ARP inspection (DAI), and
- IP source guard.

All three methods can create IP-to-MAC bindings in the binding table, although both DAI and IP source guard can be configured to run in inspection-only mode, limiting the association of IP addresses to MAC addresses to DHCP-snooping. Bindings created as a result of DHCP exchanges with trusted servers (DHCP-snooping) take precedence over bindings created through DAI or IP source guard. Use of all three tools allows bindings to be created for users in a network where DHCP is not in use or where a DHCP exchange has not occurred since the anti-spoofing feature has been enabled.

The actions that may be taken against a violating user include:

- Logging a message
- Sending a notification
- Putting the user in quarantine, as defined by a policy profile

DHCP Snooping

DHCP snooping provides the foundation for IP spoofing detection and prevention. DHCP ACK packets received on a trusted port from a DHCP server create a MAC-to-IP binding for the user along with the lease time and expiration. DHCP ACK packets received on any ports that are configured as untrusted should be dropped as configured by policy.

On edge devices, an optional configuration is to verify the SA (source address) MAC address of the client with the client hardware address found in the DHCP payload. Provided that policy is appropriately configured to determine trusted ports for DHCP servers versus DHCP clients in an exclusively DHCP environment, and configured on an edge switch, DHCP snooping is deterministic in binding an IP address to a MAC address. Dynamic ARP inspection and IP source guard can be used to supplement the bindings database and create a secure network.

DHCP Snooping Port Mode

In a DHCP snooping context, there are three configurable port modes that determine anti-spoofing behavior:

- Trusted – When port mode is set to trusted, DHCP server traffic is accepted and used to create bindings in the source MAC address to IP address binding table for the user. Binding verification does not take place on trusted ports.
- Bypass – When port mode is set to bypass, snooping of DHCP server traffic does not take place on the port.
- Untrusted – When port mode is set to untrusted, the untrusted server counter is incremented when DHCP server traffic is detected on the port. Client traffic on these ports is processed when MAC verification is enabled on these ports.

Bindings created as a result of DHCP exchanges on trusted ports using DHCP snooping take precedence over bindings created through dynamic ARP inspection or IP source guard.

DHCP Snooping MAC Verification

The DHCP client packet contains an L2 source MAC address and an L3 client hardware address. When DHCP snooping MAC verification is enabled, DHCP snooping verifies that the source MAC address and the client hardware address match in DHCP client packets that transit untrusted ports. If the addresses do not match, the packet is dropped.

DHCP MAC verification is a network edge feature that should be enabled on ports transited by client packets from the intended client. For DHCP snooping MAC verification to be operational:

- DHCP snooping must be enabled, globally and on the port
- The port mode must be set to untrusted

Dynamic ARP Inspection (DAI)

Dynamic ARP inspection uses the MAC-to-IP binding database to ensure that ARP packets have the proper MAC-to-IP binding. When an ARP packet enters the switch, the source MAC and IP addresses are compared to the entry in the table. If the packet data conflicts with the binding in the table, the IP change is counted and logged, and any configured actions are taken against the user.

DAI can also be configured to populate the MAC-to-IP binding table. Successfully limiting ARPs to the bound addresses in the table prevents a malicious user from inserting himself in between the end user and a gateway and poisoning network devices' ARP caches or succeeding in MITM (man in the middle) attacks.

IP Source Guard

IP source guard is another means to restrict IP traffic and take configured actions against violating users. IP traffic on a port is inspected to ensure that a user's MAC and IP addresses are found in the binding table created by DHCP snooping. Changes to a user's IP address are counted and action is taken, as configured.

Like DAI, the anti-spoofing feature can be configured so that IP source guard is also able to add entries to the MAC-to-IP binding database dynamically, based upon IP traffic traversing the switch. This is particularly beneficial in an environment not limited to edge devices or one in which DHCP is not the sole proprietor of network IP addresses.

Duplicate IP Address Detection

In addition to the anti-spoofing tools described above, the anti-spoofing feature can also be configured to log, through SYSLOG and SNMP traps, duplicate IP addresses when they are bound to different MAC addresses. This situation is usually due to a misconfiguration in the network and is generally not indicative of an attack, but can be a worthwhile event to record, as administrative action may be needed to reconcile the condition. These duplicate IP addresses are only detected upon a user's binding change, and do not apply to duplicate IP addresses over ports for the same MAC address (for example, if a single user moves from one port to another).

Populating the MAC-to-IP Binding Table

The anti-spoofing MAC-to-IP binding table can be populated through DHCP snooping, dynamic ARP inspection, and IP source guard. Regardless of which of these three methods are adding entries to that table, an entry cannot be added if there is not already an entry for the user's MAC address in the multiauth session table. (Refer to the chapter entitled “Authentication Configuration” in this book for more information about the session table.)

Bindings Created by DHCP Snooping

DHCP snooping watches DHCP exchanges to create a MAC to IP binding for a client. A basic DHCP client/server exchange is as follows:

1. client -> server: DISCOVER
2. server -> client: OFFER
3. client -> server: REQUEST
4. server ->client: ACKNOWLEDGE

It is the acknowledgement from the server that creates the binding, and the server message is considered authoritative. (No other security measures other than those described here are used to ensure that the server is legitimately responding to a client request.) The ACK message includes the client hardware address and the client's confirmed IP address. It is the client hardware address (not the MAC destination address) that is used in determining if there is already an entry in the multiauth session table for the user, to which the IP address will be bound. If there is no entry in the session table for the client, a message will be logged.

Only DHCP server ACK messages received on trusted ports will populate MAC-IP address bindings. On untrusted ports, any DHCP server packets are recorded (that is, the counter is incremented), but they are allowed to be further processed. If policy is properly configured, the packets will be dropped or the port will be shut down, as configured. These server messages are not used to populate MAC to IP bindings. Bypass ports ignore all DHCP server packets for purposes of populating the binding database.

DHCP server messages are are limited to trusted ports, so the bindings that are created by them are not intended to be recorded as violations. In the case that a server sends a client a new binding (with a different IP) before the current binding's lease has expired, the event will trigger a SYSLOG message, but will not increment the violation counter.

If neither DAI nor IP source guard are configured to populate the bindings table (disabled or inspection-only), DHCP server ACK message are required to create the IP binding for a user. In this configuration, the switch will drop any DHCP server messages that cannot be processed by the soft path. The expected result of this would be for the DHCP client to re-initiate a request to the server and thus give the switch another opportunity to add the entry to the binding table. When either DAI or IP source guard is configured to populate the bindings table this is not necessary (the switch can forward the ACK), as the user's binding is able to be populated by other traffic sourced from the user, so the packet would not be discarded.

Bindings Created by DAI or IP Source Guard

When DAI or IP source guard are enabled, the other traffic being inspected (ARP or IP) can also populate the IP address bindings table. With ARP inspection, the sender MAC and IP and target MAC and IP from the ARP payload are used to populate the bindings, as provided by the ARP request or reply. With IP inspection, the source MAC address and IP address are used in creating these bindings.

If a binding already exists for a user due to DHCP, and the lease time has not expired, the DHCP binding takes precedence and a violation is recorded, but the binding does not change. If there is an entry for the user in the multiauth session table and DHCP snooping has not provided a MAC to IP address binding table entry, the ARP or IP traffic can create the MAC to IP address binding table entry. This form of entry creation allows for the anti-spoofing feature to adapt to environments that are not on the edge or are not able to monitor and process all DHCP exchanges on the network for attached users.

Expiration of Bindings

IP address bindings will timeout when a lease expires, a DHCP release frame is received, or upon manual clearing of an entry, whichever occurs first. For DHCP-snooping created bindings, after the lease expires, the binding also expires. However, for DAI and IP-inspection, the counter resets after the timeout period, but the binding remains active (restarts the timer).

When you manually set a timeout period, be aware that the lease time defined in the DHCP server scope takes precedence over manually set timeouts.

Implementing Anti-Spoofing in Your Network

Using DHCP Snooping Only

On an edge device in an environment where DHCP is exclusively the provider of IP addresses, the switch with DHCP snooping enabled will record all user's DHCP interactions and should have an IP address binding for each connected user.

Untrusted ports do not create bindings from DHCP server packets. Optionally, the client hardware address in the DHCP packet is verified to match the source MAC address of the packet. If it does not, it is dropped. This is a more robust security feature that can be used on the edge of the network where it is expected that the client requests are coming from the client, not a different switch, router, or AP.

No port class actions are taken against users whose IP address assignment changes due to DHCP (where the server responses are on a trusted port), and user counters don't increment. Without DAI or IP source guard configured, anti-spoofing ensures that server packets are only handled

where appropriate, that malicious users do not release or decline DHCP IP address assignments for other users, that DHCP client request packets are coming from the actual client (optional), and that the MAC-IP address binding database is populated. In addition, policy should be configured to drop any unwanted server traffic on untrusted ports.

If IP source guard and DAI are disabled or configured for inspection-only away from the edge of a network, DHCP exchange packets could be missed — for example, link loss at the distribution or core layer would not necessarily cause DHCP renewals from the end users at the edge, thus the binding table would not be repopulated — and users could suffer the consequence of unintended violations (for example, denial of service).

However, there are still benefits for using DHCP snooping without IP source guard or DAI away from the edge of the network. This type of network configuration allows for user accounting (user IP address change counters) and allows for the population of the user IP address binding table from known DHCP servers. The binding table will then allow user leases to run for the configured lease time used on the network before turning on other anti-spoofing features. In this scenario, an administrator should recognize that configuring any actions that limit a user's traffic after a violation could potentially disrupt network traffic for an otherwise legitimate user. Generally, this configuration would not be used away from the edge to quarantine or otherwise limit the users' traffic, as these limitations could be manipulated to cause denial of service attacks against a user.

Using DAI, IP Source Guard, and Duplicate IP Detection

Once DAI is enabled or set to inspection-only, ARP packet inspection occurs. On those ports, all ARP traffic is intercepted and the MAC and IP address of the ARP is verified against the entry in the MAC to IP address binding table. Actions may be taken against the user if the violation threshold has been crossed for the port, as configured by the port class.

Similarly, if IP source guard is enabled or configured for inspection-only, IP traffic is intercepted and verified against the binding table. Once a connection is created, that traffic won't be inspected again unless the source IP address associated with the MAC address changes. As IP address changes are detected and configured thresholds for that value are crossed, the anti-spoofing feature will take action, depending on the configuration of the class of port with which the user is associated. These actions will be to SYSLOG the event, send an SNMP notification, or perform the quarantine action. The quarantine action is configurable through the policy and multiauth quarantine controls. Extreme Networks **highly recommends that you use quarantine policies to classify the user traffic upon violation hits.**

If the duplicate IP detection feature is enabled, when new MAC to IP bindings are created or current bindings are changed, an IP address lookup is run on the bindings database to verify that the IP is not currently in use. If it is in use, a SYSLOG message and trap are sent.

Anti-Spoofing Configuration

Overview

You can enable and disable anti-spoofing on a global and per-port basis. When the feature is globally disabled, no anti-spoofing features are active. Anti-spoofing must be globally enabled before port control values are considered when inspecting traffic. The default value for all anti-spoofing features, global and per port, is disabled.

DHCP snooping is controlled through port enable/disable commands, as well as per port MAC verification enable/disable commands. DAI and IP source guard have individual controls to enable, disable, and enable inspection-only (no binding association) on a per port level. Duplicate IP address detection can be enabled or disabled globally.

Port mode, or type, determines the role traffic traversing the port will take in DHCP snooping. DHCP server messages are only processed (for DHCP snooping purposes) on trusted ports. On untrusted ports, DHCP server messages are counted in the untrusted packet counter (per port). If configured by policy, these message can also be dropped.

On bypass ports, DHCP server messages are ignored (that is, they do not affect the source MAC/source IP binding database, but they are not dropped). Ports are untrusted by default.

Port Classes

Enabling anti-spoofing on both the global and port level results in snooping frames, but it does not necessarily result in any actions being taken on IP address binding violations. For this, port classes must be defined and ports added to the appropriate port class. Port classes are configured with thresholds and actions, and potentially an action value. Currently, up to 3 port classes can be configured on the switch.

Up to 6 thresholds can be configured per port class, and each threshold can be assigned one of the following actions: sending SYSLOG messages, sending SNMP notifications (traps), and applying the quarantine policy profile. Only the quarantine action can have an action value applied, which is the quarantine profile index. The quarantine profile must be configured independently, and no error checking occurs to ensure the policy profile is present.

Each port can be configured with a single class. If you only have a single anti-spoofing detection type enabled on the port, DHCP snooping for example, the class thresholds and actions can be set for that anti-spoofing detection type. If multiple anti-spoofing types are enabled on a port, DHCP snooping and dynamic ARP inspection for example, the class thresholds and actions must take into account any combination of anti-spoofing events for the configured anti-spoofing types.

If the quarantine action is specified, Extreme Networks highly recommends that you associate a valid quarantine profile with the quarantine action. Refer to the chapter entitled “Policy Configuration” in this book for information about configuring policy profiles and the chapter “Authentication Configuration” for information about using quarantine policies with the quarantine agent.

Managing the Binding Database

An entry in the source MAC address to source IP address binding table can be deleted by port, source MAC address, or source IP address. Clearing the binding also clears the IP address change count associated with the user. Alternatively, a user’s violation count can be cleared without clearing the current binding.

Configuration Examples

Procedure 54-1 describes the tasks and commands used to configure anti-spoofing features on the switch. Table 54-1 on page 54-8 describes the tasks and commands used to manage anti-spoofing features. Table 54-2 on page 54-8 describes the commands used to display anti-spoofing information.

Refer to the “Anti-Spoofing Commands” chapter in the S-Series *CLI Reference* for details about using these commands.

Procedure 54-1 Configuring Anti-Spoofing Features

Step	Task	Command(s)
1.	Create a port class and optionally, configure a name and timeout value. Up to 3 classes can be configured.	**set antispoof class** *class-index* {**name** *name* | **timeout** *timeout*}
2.	Configure thresholds and actions for the class. Up to 6 threshold indexes can be specified per class.	**set antispoof class** *class-index* **threshold-index** *thresh-index* [**threshold-value** *thresh-value*] [**quarantine-profile** *quar-profile*] [**action** {[**syslog**] [**trap**] [**quarantine**]}]
3.	Enable DHCP snooping on the desired port or ports.	**set antispoof dhcp-snooping enable** *port-string*
4.	Configure the ports on which trusted DHCP server traffic will be accepted. DHCP ACK packets received on these ports will be used to populate the MAC-to-IP address binding table. All other ports will default to untrusted mode. DHCP packets received on untrusted ports will increment the untrusted server counter.	**set antispoof dhcp-snooping port-mode trusted** *port-string*
5.	Optionally, enable DHCP snooping MAC verification on the desired untrusted port or ports.	**set antispoof dhcp-snooping mac-verification enable** *port-string*
6.	Optionally, enable dynamic ARP inspection or specify ARP packet inspection only, on the desired port or ports.	**set antispoof arp-inspection enable** *port-string* **set antispoof arp-inspection inspection-only** *port-string*
7.	Optionally, enable IP source guard or specify IP packet inspection only, on the desired port or ports.	**set antispoof ip-inspection enable** *port-string* **set antispoof ip-inspection inspection-only** *port-string*
8.	Optionally, configure bypass ports. DHCP server packets received on these ports will be ignored.	**set antispoof dhcp-snooping port-mode bypass** *port-string*
9.	Assign port classes to ports.	**set antispoof port-class** *class-index port-string*
10.	Globally enable anti-spoofing features on the switch.	**set antispoof enable**
11.	Optionally, change the notifications interval. The default value is 60 seconds. Note that sending notifications is enabled by default.	**set antispoof notifications interval** ***interval***
12.	Optionally, enable duplicate IP address detection.	**set antispoof duplicateIP enable**

Table 54-1 lists the commands used to disable or reset anti-spoofing features and to manage the binding table entries.

Table 54-1 Managing Anti-Spoofing Features

Task	Command(s)
Disable anti-spoofing globally	**set antispoof disable** **clear antispoof**
or Reset all anti-spoofing configuration to default values.	**clear antispoof all**
Disable sending anti-spoofing notifications.	**set antispoof notifications disable**
Reset the notification interval to the default of 60 seconds.	**clear antispoof notifications interval**
Disable duplicate IP address detection.	**set antispoof duplicateIP disable** **clear antispoof duplicateIP**
Delete an anti-spoofing port class or clear specific configuration values to their defaults.	**clear antispoof class** *class-index* [**name**] [**timeout**] [**threshold-index** *thresh-index*]
Disable DHCP snooping on the specified port or ports.	**clear antispoof dhcp-snooping** *port-string*
Disable DHCP snooping MAC verification on the specified port or ports.	**clear antispoof dhcp-snooping mac-verification** *port-string*
Reset the DHCP snooping port mode to untrusted for the specified port or ports.	**clear antispoof dhcp-snooping port-mode** *port-string*
Disable dynamic ARP inspection on the specified port or ports.	**set antispoof arp-inspection disable** *port-string* **clear antispoof arp-inspection** *port-string*
Disable IP source guard on the specified port or ports.	**set antispoof ip-inspection disable** *port-string* **clear antispoof ip-inspection** *port-string*
Remove an anti-spoofing port class assignment from the specified port or ports.	**clear antispoof port-class** *port-string*
Delete an anti-spoofing user source MAC address to source IP address binding from the binding table.	**clear antispoof binding** {**port** *port-string* | **mac** *mac-addr* | **ip** *ip-addr*}
Reset the anti-spoofing threshold counters to 0 by port, MAC address, or IP address.	**clear antispoof counters** {**port** *port-string* | **mac** *mac-addr* | **ip** *ip-addr*}

Table 54-2 lists the commands used to display anti-spoofing information.

Table 54-2 Displaying Anti-Spoofing Information

Task	Command
Display global anti-spoofing values	**show antispoof**
Display anti-spoofing class information	**show antispoof class** [*class-index*]
Display anti-spoofing port configuration	**show antispoof port** [*port-string*] [**-interesting**]
Display anti-spoofing source MAC address to source IP address bindings	**show antispoof binding** [**port** *port-string*] [**mac** *mac-addr*] [**ip** *ip-addr*] [**all**] [**-verbose**]

Table 54-2 Displaying Anti-Spoofing Information (continued)

Task	Command
Display anti-spoofing statistics	**show antispoof counters** [**port** *port-string*] [**mac** *mac-addr*] [**ip** *ip-addr*] [**all**] [**-verbose**]

Code Example

The following example configures anti-spoofing features on a switch at the edge of the network, with two ports connected to a DHCP server and the rest of the ports connected to users. DHCP snooping is configured on the ports connected to the DHCP server so the binding table will be populated by DHCP snooping.

Two sets of user ports are configured for ARP inspection or IP source guard inspection, but are enabled for inspection only, since the binding table entries are added by DHCP snooping on the DHCP server trusted ports. Also, DHCP snooping MAC verification is enabled on the untrusted user ports.

As part of the configuration:

- Two port classes and timeout, threshold, and action values for those classes are configured.
- DHCP-snooping is enabled on the ports connected to the DHCP server, and they are configured as trusted ports.
- A simple policy profile is created that will drop DHCP server traffic and it is applied to untrusted ports.
- DAI and IP source guard are configured for inspection only on user ports 10 through 40.
- MAC verification is enabled on all user ports. (DHCP snooping must also be enabled on these ports for MAC verification to work.)
- The appropriate port class is assigned to the user ports.
- The notifications interval is changed to 30 seconds.
- Anti-spoofing is enabled globally and duplicate IP address detection is enabled.

This example assumes that quarantine policy profile 3 has previously been configured. Refer to the “Authentication Configuration” chapter in this book for more information about using quarantine profiles and the quarantine agent.

```
S Chassis(su)->set antispoof class 1 name DHCP
S Chassis(su)->set antispoof class 1 timeout 7200
S Chassis(su)->set antispoof class 1 threshold-index 1 threshold-value 1
action syslog trap

S Chassis(su)->set antispoof class 2 name “IPSG and DAI”
S Chassis(su)->set antispoof class 2 timeout 3600
S Chassis(su)->set antispoof class 2 threshold-index 1 threshold-value 1 action
syslog
S Chassis(su)->set antispoof class 2 threshold-index 2 threshold-value 2 action
trap
S Chassis(su)->set antispoof class 2 threshold-index 3 threshold-value 3
quarantine-profile 3 action quarantine

S Chassis(su)->set policy profile 1 name DHCP
```

```
S Chassis(su)->set policy rule 1 udpsourceportIP 67 mask 16 drop
S Chassis(su)->set policy port ge.2.10-40 1

S Chassis(su)->set antispoof dhcp-snooping enable ge.2.2,4
S Chassis(su)->set antispoof dhcp-snooping port-mode trusted ge.2.2,4

S Chassis(su)->set antispoof arp-inspection inspection-only ge.2.10-40
S Chassis(su)->set antispoof ip-inspection inspection-only ge.2.10-40
S Chassis(su)->set antispoof dhcp-snooping enable ge.2.10-40
S Chassis(su)->set antispoof dhcp-snooping mac-verification enable ge.2.10-40

S Chassis(su)->set antispoof port-class 1 ge.2.2,4
S Chassis(su)->set antispoof port-class 2 ge.2.10-40

S Chassis(su)->set antispoof enable
S Chassis(su)->set antispoof notifications interval 30
S Chassis(su)->set antispoof duplicateIP enable
```

55

RADIUS-Snooping Configuration

This document provides the following information about configuring RADIUS-Snooping on the Extreme Networks S-Series platforms.

Using RADIUS-Snooping in Your Network

RADIUS-Snooping (RS) is one of the Extreme Networks MultiAuth suite of authentication methods. See Chapter 56, **Authentication Configuration** for a detailed discussion of the other authentication methods supported by the S-Series platform. RS resides on the distribution-tier switch, allowing for management of any directly connected edge switch that uses the RADIUS protocol to authenticate a network end-station, but does not support the full complement of the Extreme Networks Secure Networks™ capabilities.

The RADIUS client edge-switch initiates an authentication request, by sending a RADIUS request to the RADIUS server that resides upstream of the distribution-tier switch. By investigating the RADIUS request frames, RS can determine the MAC address of the end-user device being authenticated. The network administrator creates a user account on the RADIUS server for the end-user that includes any policy, dynamic VLAN assignment, and other RADIUS and RS attributes for this end-station. By investigating the RADIUS response from the RADIUS server, RS can build a MutiAuth session as though the end-user were directly connected to the distribution-tier device.

Sessions detected by RS function identically to local authenticated sessions from the perspective of the Extreme Networks MultiAuth framework, with the exception that RS can not force a reauthentication event; it can only timeout the session.

RADIUS-Snooping allows the Extreme Networks S-Series distribution-tier switch to identify RADIUS exchanges between devices connected to edge switches and apply policy to those devices even when the edge switch is from another vendor and does not support policy. RADIUS-Snooping provides, but is not limited to, the following functionalities:

- RFC 3580 Dynamic VLAN assignment
- Authentication modes support

- Idle and session timeouts support
- Multi-user authentication on a port
- Multi-authentication method support

With RS-enabled on the distribution-tier switch, these Secure Networks capabilities can be configured by the network administrator on an end-user basis.

RADIUS-Snooping accounting is supported.

Implementing RADIUS-Snooping

RS requires that unencrypted RADIUS request frames, from the edge switch, transit the distribution-tier switch, before proceeding to the up-stream RADIUS server for validation.

Note: A router cannot reside between the RADIUS client and the distribution-tier switch enabled for RS. The presence of a router would modify the calling-station ID of the RADIUS request frame that RS depends upon to learn the MAC address of the end-station for this session.

To configure RS on a distribution-tier switch:

- Set the global MultiAuth mode to **multi**
- Set the MultiAuth port mode to **auth-opt** for all ports that are part of the RS configuration
- Globally enable RS on the distribution-tier switch
- Enable RS on all ports over which RADIUS request and response frames will transit
- Optionally change the period RS will wait for a RADIUS response frame from the server
- Populate the RADIUS-Snooping flow table with RS client and RADIUS server combinations
- Optionally enable RADIUS-Snooping accounting

RADIUS-Snooping Overview

This section provides an overview of RADIUS-Snooping configuration and management.

RADIUS-Snooping Configuration

MultiAuth Configuration

MultiAuth must be enabled if the RADIUS-Snooping configuration involves the authentication of more than a single user on a port. There are two aspects to multiauthentication in a RADIUS-Snooping configuration:

- The global MultiAuth mode must be changed from the default mode of **strict** to **multi**, in order to authenticate multiple downstream users.
- The MultiAuth port mode must be set to **auth-opt** for both upstream (to the RADIUS server) and downstream (to the authenticating switch) ports. Setting global MultiAuth to **multi** sets the default port value from **auth-opt** to **force-auth**. Reset the mode for the affected ports to **auth-opt**.

See the "MultiAuth Authentication" on page 56-7 for a complete discussion on MultiAuth configuration.

Enabling RADIUS-Snooping

RS is enabled globally on the distribution-tier switch. It is also enabled on the distribution-tier switch ports directly attached to the edge switch that the RADIUS request frames transit, from the edge switch to the RADIUS server, as well as the ports the response frames transit, from the RADIUS server back to the edge switch.

Configuring Enabled Port Settings

The number of seconds the firmware waits for a RADIUS response after it successfully snoops a RADIUS request can be set per-port. If you do not set this timeout at the port level, the system level setting is used.

In some cases it may be necessary to drop RADIUS traffic between the distribution tier device and the edge switches. You can enable or disable packet drop on a per port basis. Packets are always dropped for a resource issue situation. RS is not capable of forcing a reauthentication event should it be unable to investigate a RADIUS request exchange. Dropping a RADIUS request packet due to resource exhaustion, in most cases, will cause the edge device to retry a RADIUS request, providing another opportunity to snoop the RADIUS exchange. Frames with an invalid format for the calling station ID are only dropped when drop is enabled. In the case of dropping frames with an invalid format, authentication will not take place for this end-user.

The **authallocated** value specifies the maximum number of RS users per port. You can configure this number of allowed RS users on a per port basis. The default value depends upon the system license for this device. You should set this **authallocated** value equal to or less than the configured value for the **set multiauth port numusers** command. This value is the maximum number of users per port for all authentication clients. Typically, **authallocated** and **multiauth port numusers** are set to the same value.

Populating the RADIUS-Snooping Flow Table

The RADIUS-Snooping flow table is a filter that determines which RADIUS server and client combinations will be snooped. If the secret is configured, the response frames are checked for valid MD5 checksum, in order to validate the sender.

The RS flow table contains RADIUS server and client entries for each RADIUS server and client combination for which RS will be used on this system. The RADIUS client IP address and authenticating RADIUS server IP address are manually entered into the RADIUS-Snooping flow table. By default, the RADIUS-Snooping flow table is empty. Entries are added to the flow table based upon an index entry. The first matching entry in the table is used for the continuation of the authentication process.

When an investigated RADIUS frame transits the RS-enabled port with a match in the flow table, RS will track that RADIUS request and response exchange and will build a MultiAuth session for the end-user, based upon what it finds in the RADIUS response frames.

Setting the RADIUS-Snooping Timeout

A timeout is configured to set the number of seconds that the firmware waits for a RADIUS response frame to be returned from the RADIUS server, after successfully snooping a RADIUS request frame from the client. If no response is seen before the timeout expires, the session is terminated.

RADIUS-Snooping Management

RADIUS-Snooping management options are available to:

- Terminate all RS sessions or on a per port or MAC address basis

- Reset all RS configuration to its default settings
- Clear all RADIUS-Snooping flow table entries or per index entry
- Display RS statistics

RADIUS Session Attributes

The RADIUS attributes defining the session are returned in the RADIUS response frame. RADIUS attributes are used to configure the user on the system. Attributes explicitly supported by RS that may be included in the RADIUS response frame are:

- Idle-Timeout – If no frames are seen from this MAC address, for the number of seconds configured, the session will be terminated.
- Session-Timeout – The session is terminated after the number of seconds configured.
- Filter-ID - Defines the policy profile (role) and CLI management privilege level, just as it would for any other local authentication agent.
- Tunnel-Group-Id – Specifies the VLAN ID for this session.

Note: Numerous attributes may be supported by the RADIUS client for general RADIUS protocol support. Such attributes are beyond the scope of this document. This RS implementation does not interfere with normal RADIUS client attribute support. The list above indicates attributes actually used by this RADIUS-Snooping application once authentication is successfully completed.

Figure 55-1 RADIUS-Snooping Overview

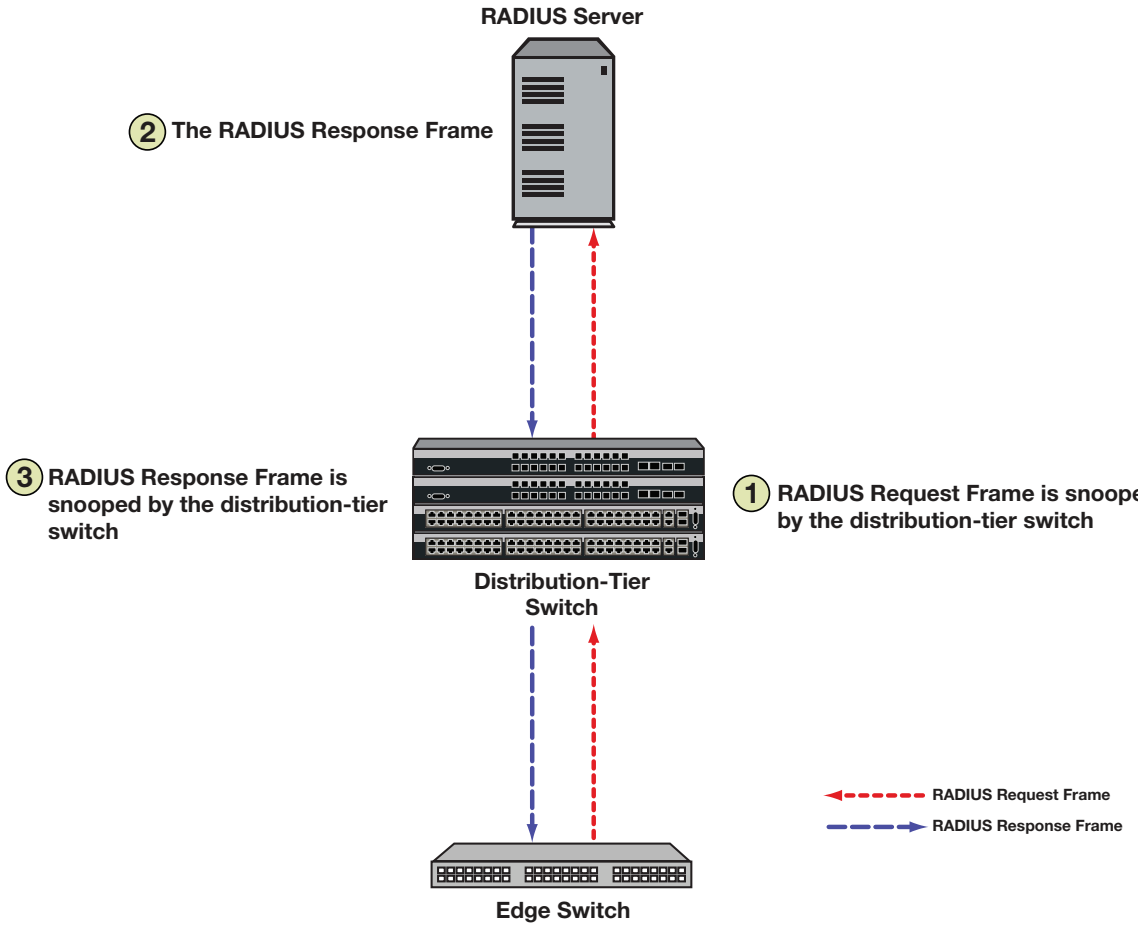


Figure 55-1 illustrates the RADIUS request frame and RADIUS response frame paths. As the RADIUS request frame from the RADIUS client edge device transits the distribution-tier switch, it is snooped. An RS session is created on the distribution-tier switch, if:

- RADIUS snooping is enabled on the switch
- RADIUS-Snooping is enabled on the port
- The RADIUS client edge device and RADIUS server combination are defined in the RADIUS snooping flow table

When the RADIUS server receives the request, the authenticating device is first validated. After validating the authenticating device, the server authenticates the user session itself based on passed username and password attributes. If that succeeds an access accept message containing RADIUS attributes is sent back to the client, otherwise an access reject message is sent back. As the RADIUS response frame transits the distribution-tier switch, the RADIUS attributes contained in the response frame are applied to this session, if an RS session was created for this client server combination and the session has not timed out.

RADIUS-Snooping agent accounting is supported and defaults to disabled. To use RADIUS-Snooping accounting, RADIUS accounting must be enabled using the **set radius accounting** command. RADIUS-Snooping agent accounting can be enabled using the **set radius-snooping accounting** command.

Configuring RADIUS-Snooping

This section provides details for the configuration of RADIUS-Snooping on the S-Series products.

Table 55-1 lists RS parameters and their default values.

Table 55-1 Default Authentication Parameters

Parameter	Description	Default Value
RADIUS-Snooping timeout	Specifies the number of seconds that the firmware waits, from the time it successfully snoops a RADIUS request frame, for a RADIUS response frame from the RADIUS server, before terminating the session.	20 seconds
RADIUS-Snooping accounting	Specifies whether RADIUS-Snooping accounting is enabled or disabled on the device.	Disabled
RS system and port state	Enables or disables RS on the distribution-tier switch in a system context or on this port in a port context. Enables or disables packet drop in a port context.	Disabled
authallocated	Specifies the maximum number of allowed RS sessions from all RADIUS clients, on a per port basis.	8, 128, or 256 depending upon the system license for this device
drop	Specifies traffic drop behavior for this port.	Disabled
index	The numeric ID of a RADIUS-Snooping flow table entry.	None
UDP port	Specifies the RADIUS UDP port.	1812
secret	Specifies the RADIUS secret for this RADIUS-Snooping flow table entry.	No secret

Configuring RADIUS-Snooping on the Distribution-Tier Switch

Procedure 55-1 describes how to configure RADIUS-Snooping on the distribution-tier switch.

Procedure 55-1 RADIUS-Snooping Configuration

Step	Task	Command(s)
1.	Globally enable MultiAuth for **multi** mode.	**set multiauth mode multi**
2.	Configure each upstream and downstream port for the **auth-opt** mode.	**set multiauth port mode auth-opt** *port-string*
3.	Globally enable RADIUS-Snooping on the distribution-tier switch.	**set radius-snooping enable**
4.	Optionally enable RADIUS-Snooping accounting on the device.	**set radius-snooping accounting enable**

Procedure 55-1 RADIUS-Snooping Configuration

Step	Task	Command(s)
5.	Enable RADIUS-Snooping on each distribution-tier switch port over which RADIUS request and response frames transit.	**set radius-snooping port** [**enable**] [**timeout** *seconds*] [**drop** {**enabled** | **disabled**}] [**authallocated** *number*] [*port-string*]
6.	Configure RADIUS-Snooping flow table index entries.	**set radius-snooping flow** *index* {*client-IP-Address server-IP-Address* {*port* } [*secret*]
7.	Optionally modify the RADIUS-Snooping timeout setting.	**set radius-snooping timeout** *seconds*

Managing RADIUS-Snooping

Table 55-2 describes how to manage RADIUS-Snooping on the distribution-tier switch.

Table 55-2 Managing RADIUS-Snooping

Task	Command(s)
To terminate active sessions on the system for the specified port or MAC address.	**set radius-snooping initialize** {**port** *port-string* | *mac-address*}
To reset all RS configuration to its default value on this system.	**clear radius-snooping all**
To clear all entries or the specified index entry from the RS flow table.	**clear radius-snooping flow** {**all** | *index*}

Displaying RADIUS-Snooping Statistics

Table 55-3 describes how to display RADIUS-Snooping statistics.

Table 55-3 Displaying RADIUS-Snooping Statistics

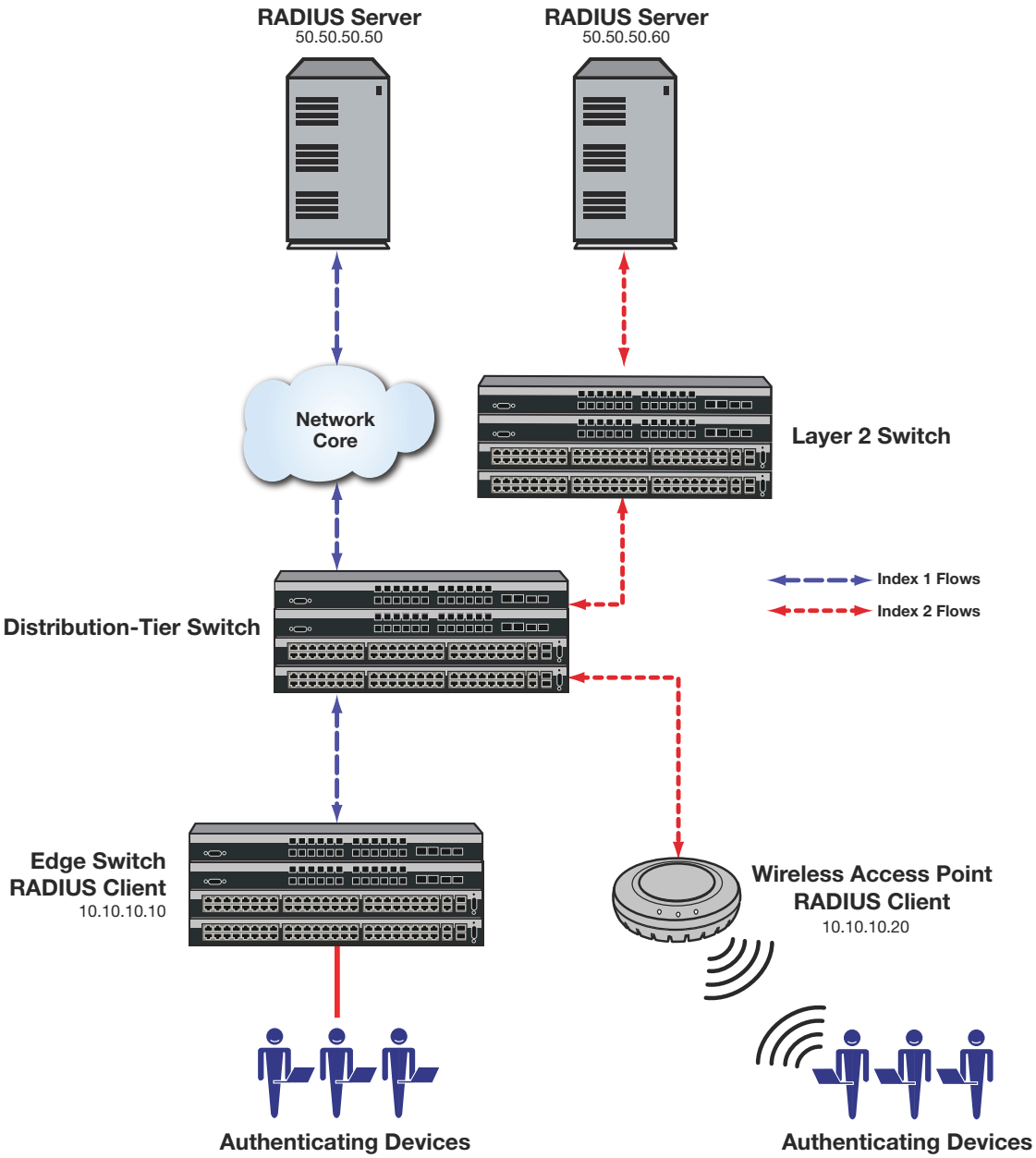
To display a general overview of the global RS status.	**show radius-snooping**
To display the RS status for the specified port.	**show radius-snooping port** *port-string*
To display information for all or the specified flow index entry.	**show radius-snooping flow** {*index* | **all**}
To display a summary of sessions for the specified port or MAC address.	**show radius-snooping session** {**port** *port-string* | **mac** *mac-address*}

RADIUS-Snooping Configuration Example

Our RADIUS-Snooping configuration example will configure a distribution-tier switch for two RADIUS request and response flows (index 1 and index 2). Index 1 is from RADIUS client 10.10.10.10 through the network core to the RADIUS server 50.50.50.50. Index 2 is from RADIUS client 10.10.10.20 through a layer 2 switch to the local RADIUS server 50.50.50.60. Each flow is transiting the single distribution-tier switch configured in this example.

See Figure 55-2 for an illustration of the example setup.

Figure 55-2 RADIUS-Snooping Configuration Example Overview



We first enable RADIUS-Snooping at the system level for the distribution-tier switch. We then enable two sets of ports (ge.1.5-10 and ge.1.15-24) over which all RADIUS-Snooping request and response frames will transit. In the same command line we:

- Enable drop on all ports
- Set the maximum number of RS sessions per port to 256

We then configure the two flows as specified above for UDP port 1812 and a secret of **mysecret**.

We complete the configuration by changing the timeout value at the system level to **15** seconds from a default of **20** seconds.

Configure the Distribution-tier Switch

Set the MultiAuth mode for the system

```
S Chassis(su)->set multiauth mode multi
```

Set the MultiAuth authentication mode for each port

```
S Chassis(su)->set multiauth port mode auth-opt ge.1.5-10,15-24
```

Enable RS on this system:

```
S Chassis(su)->set radius-snooping enable
```

Enable RS and set configuration for ports on this system

```
S Chassis(su)->set radius-snooping port enable drop enabled authallocated 256
ge.1.5-10
S Chassis(su)->set radius-snooping port enable drop enabled authallocated 256
ge.1.15-24
```

Configure RS flow table entries

```
S Chassis(su)->set radius-snooping flow 1 10.10.10.10 50.50.50.50 1812 mysecret
S Chassis(su)->set radius-snooping flow 2 10.10.10.20 50.50.50.60 1812 mysecret
```

Configure RS timeout for this system

```
S Chassis(su)->set radius-snooping timeout 15
```

Managing RADIUS-Snooping on the Distribution-tier Switch

Terminate an active session on port ge.1.15:

```
S Chassis(su)->set radius-snooping initialize port ge.1.15
```

Reset all RS configuration to its default value:

```
S Chassis(su)->clear radius-snooping all
```

Clear entry index 2 from the RS flow table:

```
S Chassis(su)->clear radius-snooping flow 2
```

This completes the RADIUS-Snooping configuration example.

Terms and Definitions

Table 55-4 lists terms and definitions used in this RADIUS-Snooping configuration discussion.

Table 55-4 RADIUS-Snooping Configuration Terms and Definitions

Term	Definition
Calling-Station ID	An attribute field in the RADIUS request and response frames containing the RADIUS client MAC address.
Distribution-Tier Switch	The switch that aggregates edge switch traffic heading into the core network or other distribution devices.
Edge Switch	The switch directly connected to the end-user device.
Filter-ID	A vendor defined RADIUS attribute that the Extreme Networks S-Series authentication implementation makes use of, allowing the authenticating device to assign policy, CLI privilege level, and dynamic VLAN assignment to the end-user.
Multi-Authentication Methods	The ability to authenticate a user for multiple authentication methods such as 802.1x, MAC, PWA, or CEP, while only applying the authentication method with the highest authentication precedence.
Multi-User Authentication	The ability to authenticate multiple users on a port, assigning unique policy to each user based upon the user account RADIUS server configuration and policy configuration on the distribution-tier switch.
MutiAuth Framework	The aspect of Secure Networks functionality that provides authentication capabilities including, but not limited to, multi-user and multi-method authentication, application of policy and Dynamic VLAN assignment.
RADIUS Client	In a RADIUS-Snooping context the RADIUS client is the non-Secure Networks capable edge switch that is responsible for authenticating its attached end-user device or port.
RADIUS-Snooping flow table	A table containing the RADIUS client and server ID defining valid RS sessions.
RADIUS Request Frames	Frames sent by the RADIUS client to the RADIUS server requesting end-user authentication validation.
RADIUS Response Frames	Frames sent by the RADIUS server to the RADIUS client either validating or rejecting an authentication validation request. These frames can also contain the Filter-ID attribute allowing the assignment of policy, CLI privilege, and dynamic VLAN assignment.
RADIUS-Snooping	Provides non-Secure Networks capable edge switches with the full range of Secure Networks authentication capabilities when the RADIUS server is upstream of the distribution-tier switch.

56

Authentication Configuration

This document provides the following information about configuring user authentication on the Extreme Networks S-Series platforms.

Using Authentication in Your Network

Authentication is the ability of a network access server, with a database of valid users and devices, to acquire and verify the appropriate credentials of a user or device (supplicant) attempting to gain access to the network. Extreme Networks authentication uses the RADIUS protocol to control access to switch ports from an authentication server and to manage the message exchange between the authenticating device and the server. Both MultiAuth and Multi-User authentication are supported. MultiAuth is the ability to configure multiple authentication modes for a user and apply the authentication mode with the highest precedence. Multi-User is the ability to appropriately authenticate multiple supplicants on a single link and provision network resources, based upon an appropriate policy for each supplicant. The Extreme Networks switch products support the following seven authentication methods:

- Quarantine agent
- IEEE 802.1x
- Port Web Authentication (PWA)
- MAC-based Authentication (MAC)
- Convergence End Point (CEP)
- RADIUS Snooping (Refer to Chapter 55, **RADIUS-Snooping Configuration** for RADIUS Snooping configuration details)
- Auto tracking

Extreme Networks switch products support the configuration of up to seven simultaneous authentication methods per user, with a single authentication method applied based upon MultiAuth authentication precedence.

Network resources represent a major capital investment for your organization and can be vulnerable to both undesired resource usage and malicious intent from outside users. Authentication provides you with a user validation function which assures that the supplicant requesting access has the right to do so and is a known entity. To the degree a supplicant is not a known entity, access can be denied, granted on a limited basis, or granted without restriction. The ability of authentication to both validate a user's identity and define the resources available to the user assures that valuable network resources are being used for the purposes intended by the network administrator.

Accounting is supported for all authentication agent types.

Implementing User Authentication

Take the following steps to implement user authentication:

- Determine the types of devices to be authenticated.
- Determine the correct authentication type for each device.
- Determine an appropriate policy best suited for the use of that device on your network.
- Configure RADIUS user accounts on the authentication server for each device.
- Configure user authentication.

Authentication Overview

Quarantine

The quarantine agent works in conjunction with a quarantine policy rule to perform the action specified in the associated policy role if the policy rule is hit. The quarantine agent also acts in conjunction with anti-spoofing and will perform the configured class action if an anti-spoofing class threshold is met (see Chapter 54, **Anti-Spoofing Configuration** for anti-spoofing configuration details).

The quarantine agent must be enabled globally on the switch and locally on the port to be operational on the port. The quarantine agent is a form of authentication that depends upon the existence of one or more configured quarantine policy rules, with each rule associated with a policy profile. To configure a policy rule as a quarantine profile, configure the policy rule with the desired traffic filtering specifications and specify the quarantine-profile rule option, indicating the associated policy profile.

Once one or more quarantine policy rules are configured and associated with a policy profile, the quarantine authentication agent behaves as any other MultiAuth authentication agent. By default, the quarantine agent has the highest configurable MultiAuth precedence. Static rules have the highest multiauth precedence. Static rule multiauth precedence is not configurable.

There are two circumstance for which actions specified in a quarantine policy are used:

- A quarantine policy rule is hit. In this case, the quarantine agent becomes one of the authentication agents from which the authentication provisioning result will be chosen based upon MultiAuth precedence. So long as the default precedence is not changed, if a quarantine policy rule hit occurs, quarantine authentication is selected and any actions configured in the policy profile taken.
- An anti-spoofing class threshold has been met for which a quarantine action has been configured.

Should you configure quarantine authentication for a lower MultiAuth precedence, if a non-quarantine authentication agent both returns a result and has the highest MultiAuth precedence, quarantine authentication will not be used in that context. If you change the quarantine agent MultiAuth precedence level to a lower precedence, make sure this is the behavior you want.

Quarantine agent accounting is supported and defaults to disabled. To use quarantine agent accounting, RADIUS accounting must be enabled using the **set radius accounting** command. Quarantine agent accounting can be enabled using the **set quarantine-agent accounting** command.

IEEE 802.1x Using EAP

The IEEE 802.1x port-based access control standard allows you to authenticate and authorize user access to the network at the port level. Access to the switch ports is centrally controlled from an authentication server using RADIUS. The Extensible Authentication Protocol (EAP), defined in RFC 3748, provides the means for communicating the authentication information.

There are three supported types of EAP:

- **MD5** – EAP-MD5 is a challenge-handshake protocol over EAP that authenticates the user with a normal username and password.
- **TLS** – EAP-TLS provides a transport layer security based upon the presentation and acceptance of digital certificates between the supplicant and the authentication server.
- **Protected** – Protected Extensible Authentication Protocol (PEAP) optionally authenticates the authentication server to the client using an X-509 certificate using a TLS tunnel, after which the client authentication credentials are exchanged.

All Extreme Networks platforms support IEEE 802.1x, which protects against unauthorized access to a network, DoS attacks, theft of services and defacement of corporate web pages.

802.1x configuration consists of setting port, global 802.1x parameters, and RADIUS parameters on the switches to point the switch to the authentication server. The Filter-ID RADIUS attribute can be configured on the authentication server to direct dynamic policy assignment on the switch to the 802.1x authenticating end system.

802.1x agent accounting is supported and defaults to enabled. RADIUS accounting must be enabled using the **set radius accounting** command. 802.1x agent accounting can be enabled using the **set dot1x accounting** command.

MAC-Based Authentication (MAC)

MAC-based authentication (MAC) authenticates a device using the source MAC address of received packets. Two modes are supported for MAC authentication: password and RADIUS username. The MAC authentication mode is set using the **set macauthentication auth-mode** command.

By default the MAC authentication server uses an administratively configured password to authenticate a user. The default value for the password is "NOPASSWORD". The administratively configured password is set using the **set macauthentication password** command.

MAC authentication can be configured to use the RADIUS server configured username credential where the password is the same as the username. The following is an example RADIUS server configuration for MAC address 00-00-22-22-02-01, first with a mask of 48, followed by the address with a mask of 40.

The full user name with a mask of 48:

```
00-00-22-22-02-01 Auth-Type := Local, User-Password == "00-00-22-22-02-01"
                  Service-Type = Framed-User
```

The user name with a mask of 40:

```
00-00-22-22-03-00 Auth-Type := Local, User-Password == "00-00-22-22-03-00"
                  Service-Type = Framed-User
```

In either case, if the authentication server receives valid credentials from the switch, RADIUS returns an Accept message to the switch.

MAC authentication enables switches to authenticate end systems, such as printers and camcorder devices that do not support 802.1x or web authentication. Since MAC-based authentication authenticates the device, not the user, and is subject to MAC address spoofing attacks, it should not be considered a secure authentication method. However, it does provide a level of authentication for a device where otherwise none would be possible.

MAC-based authentication agent accounting is supported and defaults to enabled. RADIUS accounting must be enabled using the **set radius accounting** command. MAC-based authentication agent accounting can be disabled using the **set macauthentication accounting** command.

Port Web Authentication (PWA)

Port Web Authentication (PWA) authenticates a user by utilizing a web browser for the login process to authenticate to the network. To log in using PWA, a user opens the web browser requesting a URL that either directly accesses the PWA login page or is automatically redirected to the login page. At the PWA login page, the user enters a login username and password. On the switch, either the Challenge Handshake Authentication Protocol (CHAP) or the Password Authentication Protocol (PAP) verifies the username and password credentials provided to the authentication server. If the credentials are validated, the authentication server returns a RADIUS Access-Accept message, optionally containing Filter-ID or tunnel attributes, to the switch.

PAP uses an unencrypted password. CHAP uses the password to generate a digest that is transmitted to the authentication server. If RADIUS determines that the digest matches the digest generated on the authentication server, access is granted. The acceptance message back to the switch can contain any Filter-ID attribute configured on the authentication server, allowing policy to be applied for the authenticating user.

PWA enhanced mode is supported. PWA enhanced mode allows a user on an un-authenticated PWA port to enter any URL into the browser and be presented the PWA login page on their initial web access. When enhanced mode is disabled, a user must enter the correct URL to access login.

PWA agent accounting is supported and defaults to enabled. RADIUS accounting must be enabled using the **set radius accounting** command. PWA agent accounting can be disabled using the **set pwa accounting** command.

Convergence End Point (CEP)

CEP detects an IP telephony or video device on a port and dynamically applies a specific policy to the port. The switch detects a convergence end point by inspecting received packets for specific traffic attributes. CEP does not require a RADIUS configuration.

The CEP implementation supports the following detection methods:

- **Cisco Phone Detection** - the firmware parses a Cisco Discovery Protocol (CDP) packet to identify the phone type. If it was sent by an IP phone, the firmware uses the phone type. A response is sent back to the phone, verifying authentication.
- **Siemens HiPath Phone Detection** - TCP/UPD port number snooping is used. Port 4060 is the default port for communication.
- **H.323 Phone Detection** - TCP/UDP port number snooping and reserved IP address snooping are used. Ports 1718 - 1720 and IP address 224.0.1.41 are the default values.
- **LLDP-MED** – Detects LLDP-MED on the specified port.
- **Session Initiation Protocol (SIP) Phone Detection** - TCP/UDP port number snooping and reserved IP address snooping are used. Port 5060 and IP address 224.0.1.75 are the default values.

CEP agent accounting is supported and defaults to disabled. To use CEP agent accounting RADIUS accounting must be enabled using the **set radius accounting** command. PWA agent accounting can be enabled using the **set pwa accounting** command.

Auto-Tracking

The auto-tracking agent is a form of authentication that authenticates those sessions that are not captured by the other supported MultiAuth authentication agents (quarantine, 802.1x, PWA, MAC, CEP, and RADIUS snooping). If auto-tracking is disabled, these sessions are never entered into the session table. Many policy driven switch features depend on the session being in the session table for the feature to interact with the session. It is important that a network administrator have the ability to determine which station addresses on which ports are not being authenticated through traditional MultiAuth methods. Auto-tracking provides the administrator with the ability to assign these sessions a provisioning result based upon the contents of the admin-policy. Because these sessions can now be tracked, an administrator can determine whether and how to provision them in the future, allowing for increased security and control.

The auto-tracking authentication agent must be enabled globally on the switch and locally on the port to be operational on the port.

The auto-tracking authentication agent behaves the same as any other authentication agent, with the exception that it always returns an authentication result. By default, the auto-tracking agent has the lowest MultiAuth precedence. The auto-tracking agent is one of the authentication agents from which the authentication provisioning result will be chosen based upon MultiAuth precedence. Each authentication agent attempts to authenticate the user. All authentication agents that return a result are grouped. The authentication agent with the highest MultiAuth precedence is selected to authorize the user. For the default MultiAuth precedence ordering, all other authentication agents must fail to return an authentication result for auto-tracking to be selected. If auto-tracking is the selected authentication method, an auto-tracking session is created and if an admin-policy exists, the admin-policy provisions the user session.

It is recommended that you do not configure auto-tracking authentication for a higher MultiAuth precedence than its default setting of lowest. If a non-auto-tracking authentication agent both returns a result and has a lower MultiAuth precedence, that authentication method will never be used, because auto-tracking always returns a result and has been configured with a higher MultiAuth precedence.

Auto-tracking agent accounting is supported and defaults to disabled. To use auto-tracking accounting, RADIUS accounting must be enabled using the **set radius accounting** command. Auto-tracking agent accounting can be enabled using the **set pwa accounting** command.

Auto-tracking can be configured with a RADIUS timeout profile. The RADIUS timeout profile allows you to provision a session that encounters a RADIUS timeout condition, on a per port basis, with a policy profile other than the default policy. The RADIUS timeout profile allows a MAC address that attempted to authenticate during a RADIUS outage to be dealt with in a non-default manner based upon the contents of the specified policy profile. The RADIUS timeout profile is configured using the **set auto-tracking port radius-timeout-profile** command.

Auto-tracking can be configured with a RADIUS access reject profile. The RADIUS access reject profile allows you to provision a session that encounters a RADIUS access reject response from the RADIUS server, on a per port basis, with a policy profile other than the default policy. The RADIUS access reject profile allows a MAC address that was rejected by the RADIUS server to be dealt with in a non-default manner based upon the contents of the specified policy profile.

The RADIUS access reject profile takes precedence over the RADIUS timeout profile configured using the set auto-tracking port radius-timeout-profile command, should a RADIUS timeout take place and a RADIUS access reject has already occurred for this session.

The RADIUS access reject profile is configured using the **set auto-tracking port radius-reject-profile** command.

Multi-User And MultiAuth Authentication

This section will discuss multi-user and MultiAuth authentication. Multi-user and MultiAuth are separate concepts. The primary difference between the two is as follows:

- Multi-user authentication refers to the ability to authenticate multiple users and devices on the same port, with each user or device being provided the appropriate level of network resources based upon policy.
- MultiAuth authentication refers to the ability of a single or multiple user(s), device(s), or port(s) to successfully authenticate using multiple authentication methods at the same time, such as 802.1x, PWA, and MAC, with precedence determining which authentication method is actually applied to that user, device, or port.

Multi-User Authentication

Multi-user authentication provides for the per-user or per-device provisioning of network resources when authenticating. It supports the ability to receive from the authentication server:

- A policy traffic profile, based on the user account's RADIUS Filter-ID configuration
- A base VLAN-ID, based on the RFC 3580 tunnel attributes configuration, also known as dynamic VLAN assignment

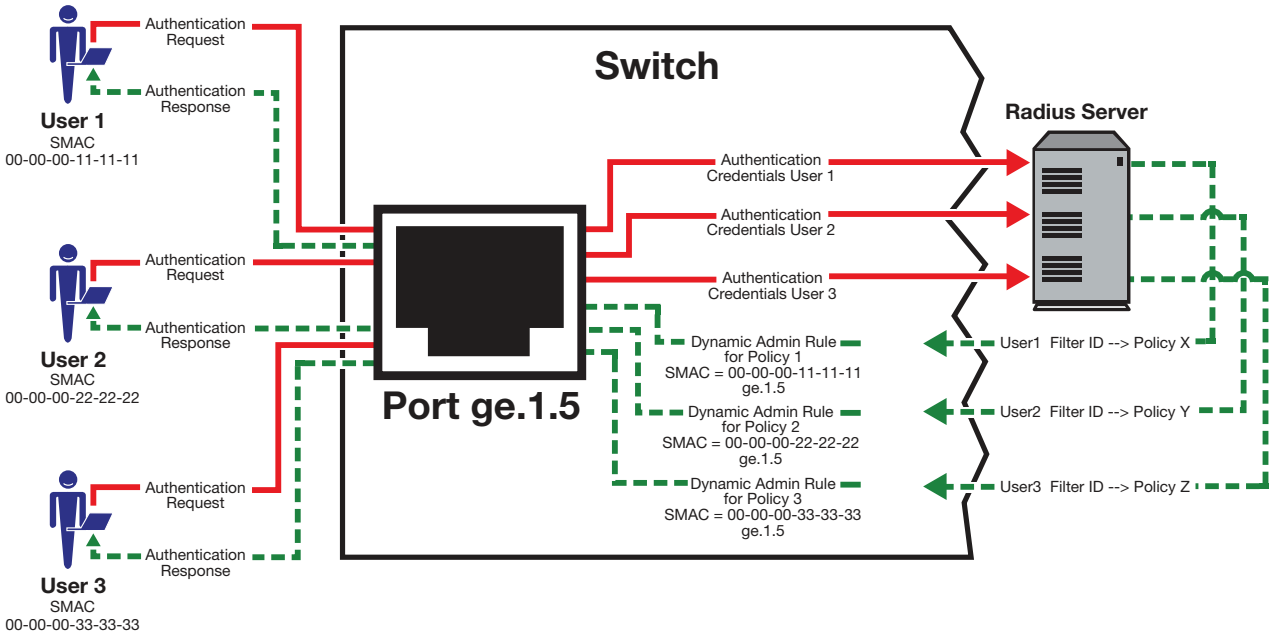
When a single supplicant connected to an access layer port authenticates, a policy profile can be dynamically applied to all traffic on the port. When multi-user authentication is not implemented, and more than one supplicant is connected to a port, firmware does not provision network resources on a per-user or per-device basis. Different users or devices may require a different set of network resources. The firmware tracks the source MAC address for each authenticating user regardless of the authenticating protocol being used. Provisioning network resources on a

per-user basis is accomplished by applying the policy configured in the RADIUS Filter-ID, or the base VLAN-ID configured in the RFC 3580 tunnel attributes, for a given user's MAC address. The RADIUS Filter-ID and tunnel attributes are part of the RADIUS user account and are included in the RADIUS Access-Accept message response from the authentication server.

The number of allowed users per port can be configured using the **set multiauth port numusers** command. See the **set multiauth port** command in the *Extreme Networks S-Series CLI Reference* for the number of supported users per module. The **show multiauth port** command displays both the allowed number of users configured and the maximum number of users supported per port for the device. The allowed number of users defaults to the maximum number of supported users for the port.

In Figure 56-1 each user on port ge.1.5 sends an authentication request to the RADIUS server. Based upon the Source MAC address (SMAC), RADIUS looks up the account for that user and includes the Filter-ID associated with that account in the authentication response back to the switch (see section "The RADIUS Filter-ID" on page 56-10 for Filter-ID information). The policy specified in the Filter-ID is then applied to the user. See section "RFC 3580" on page 56-11 for information on dynamic VLAN assignment and tunnel attribute configuration.

Figure 56-1 Applying Policy to Multiple Users on a Single Port



MultiAuth Authentication

Authentication mode support provides for the global setting of a single authentication mode 802.1X (strict-mode) or multiple modes (MultiAuth) per user or port when authenticating.

Strict mode is the appropriate mode when authenticating a single 802.1X user. All traffic on the port receives the same policy in strict mode. When authenticating PWA, CEP, or MAC, you must use MultiAuth authentication, whether authenticating a single or multiple supplicants.

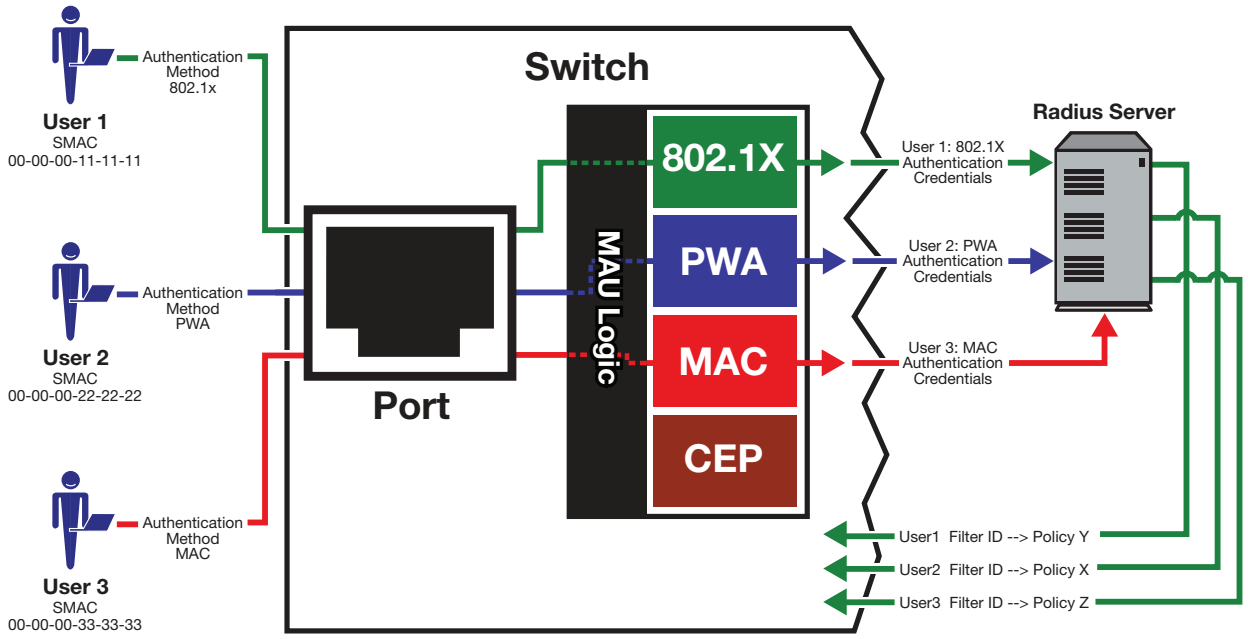
MultiAuth authentication supports the simultaneous configuration of up to seven authentication methods per user on the same port, but only one method per user is actually applied. When MultiAuth authentication ports have a combination of authentication methods enabled, and a user is successfully authenticated for more than one method at the same time, the configured authentication method precedence will determine:

- Which RADIUS-returned filter ID will be processed and result in an applied traffic policy profile, in the case of 802.1X, MAC, PWA, and CEP. See "Setting MultiAuth Authentication Precedence" on page 56-24 for authentication method precedence details.
- Whether a quarantine policy is applied, in the case of the quarantine agent
- Whether an auto-tracking session is created and an admin-policy (if it exists) is applied

The number of users or devices MultiAuth authentication supports depends upon the type of device, whether the ports are fixed access or uplink, and whether increased port capacity or extra chassis user capacity MUA licenses have been applied. See the firmware customer release note that comes with your device for details on the number of users or devices supported per port.

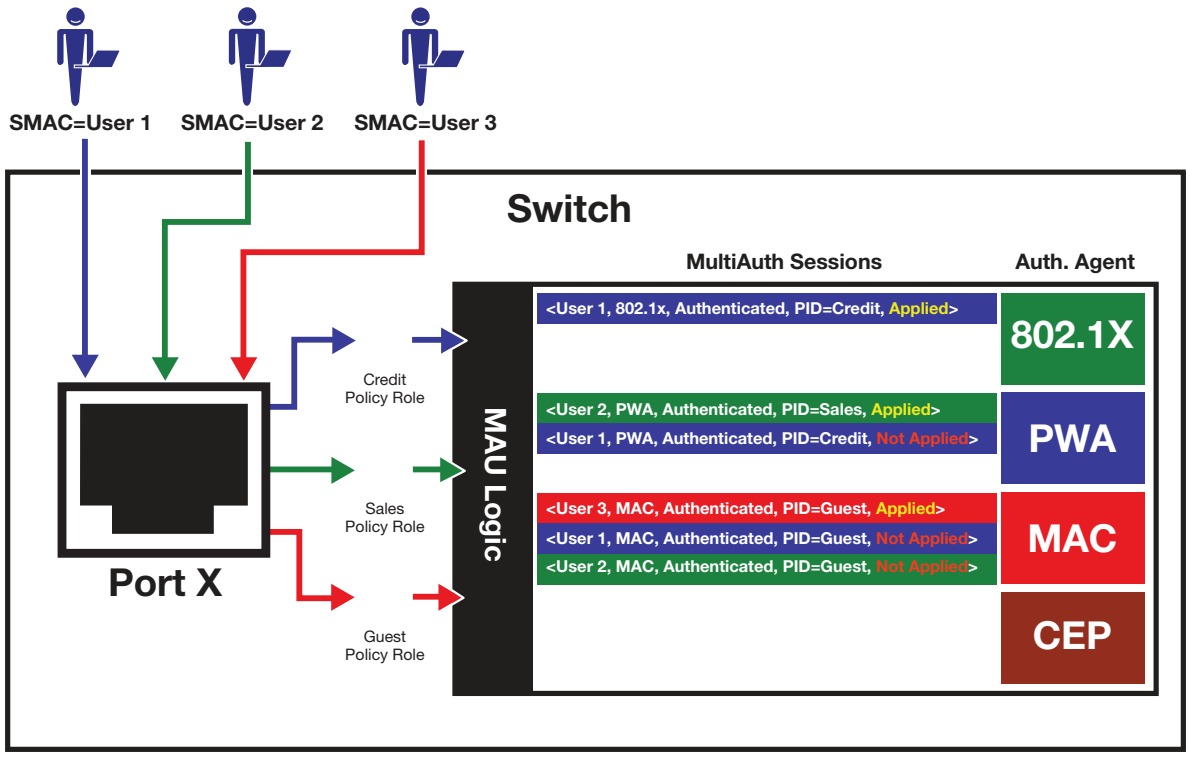
In Figure 56-2, multiple users are authenticated on a single port each with a different authentication method (in this example only 802.1X, PWA, MAC, and CEP are enabled on the device). In this case, each user on a single port successfully authenticates with a different authentication type. The authentication method is included in the authentication credentials sent to the RADIUS server. RADIUS looks up the user account for that user based upon the SMAC. The filter ID for that user is returned to the switch in the authentication response, and the authentication is validated for that user.

Figure 56-2 Authenticating Multiple Users With Different Methods on a Single Port



In Figure 56-3, full MultiAuth authentication takes place in that multiple users on a single port are validated for more than one authentication method. The applied authentication and policy are based upon the authentication method precedence level. On the far right column of the figure, the enabled authentication methods are listed from top to bottom in order of precedence. User 1 is authenticating with both the 802.1x and PWA methods, with the Credit policy. Both the 802.1x and PWA authentication methods are validated, but only the 802.1x MultiAuth session is applied, because that has the highest precedence. User 2 is authenticating with both PWA and MAC methods, with the Sales policy. PWA, having a higher precedence than MAC, is the MultiAuth session applied for User 2. User 3 is a guest and is authenticating with the MAC method only. The MAC MultiAuth session, with the Guest policy is applied for User 3.

Figure 56-3 Selecting Authentication Method When Multiple Methods are Validated



When a re-authentication attempt times out, the timeout action can either be set to terminate the session or for none, in which case the session remains authenticated and provisioned according to the prior successful RADIUS authentication response. It is recommended that you not set the re-authentication timeout action to none when using 802.1x authentication.

Remote Authentication Dial-In Service (RADIUS)

This section provides details for the configuration of RADIUS and RFC 3580 attributes.

The Remote Authentication Dial-In User Service (RADIUS) is an extensible protocol used to carry authentication and authorization information between the switch and the Authentication Server (AS). RADIUS is used by the switch for communicating supplicant supplied credentials to the authentication server and the authentication response from the authentication server back to the switch. This information exchange occurs over the link-layer protocol.

The switch acts as a client to RADIUS using UDP port 1812 by default (configurable in the **set radius** command). The authentication server contains a database of valid supplicant user accounts with their corresponding credentials. The authentication server checks that the information received from the switch is correct, using authentication schemes such as PAP, CHAP, or EAP. The

authentication server returns an Accept or Reject message to the switch based on the credential validation performed by RADIUS. The implementation provides enhanced network security by using a shared secret and MD5 password encryption.

Required authentication credentials depend upon the authentication method being used. For 802.1x and PWA authentication, the switch sends username and password credentials to the authentication server. For MAC authentication, the switch sends the device MAC address and a password configured on the switch to the authentication server. The authentication server verifies the credentials and returns an Accept or Reject message back to the switch.

How RADIUS Data Is Used

The Extreme Networks switch bases its decision to open the port and apply a policy or close the port based on the RADIUS message, the port's default policy, and unauthenticated behavior configuration.

RADIUS provides accounting functionality by way of accounting packets from the switch to the RADIUS server, for such session statistics as start and end, total packets, and session end reason events. This data can be used for both billing and network monitoring purposes.

Additionally RADIUS is widely used by VoIP service providers. It is used to pass login credentials of a SIP end point (like a broadband phone) to a SIP Registrar using digest authentication, and then to the authentication server using RADIUS. Sometimes it is also used to collect call detail records (CDRs) later used, for instance, to bill customers for international long distance.

If you configure an authentication method that requires communication with an authentication server, you can use the RADIUS Filter-ID attribute to dynamically assign either a policy profile or management level to authenticating supplicants.

The RADIUS Filter-ID

The RADIUS Filter-ID attribute consists of a string that is formatted in the RADIUS Access-Accept packet sent back from the authentication server to the switch during the authentication process.

Each user can be configured in the RADIUS server database with a RADIUS Filter-ID attribute that specifies the name of either a policy profile or management level the user should be assigned upon successful authentication. During the authentication process, when the authentication server returns a RADIUS Access-Accept packet that includes a Filter-ID matching a policy profile name configured on the switch, the switch then dynamically applies the policy profile to the physical port the supplicant is authenticating on.

The decorated Filter-ID supports a policy attribute, a management access attribute, or both in the following formats:

`Enterasys:version=1:policy=`*`policyname`*

`Enterasys:version=1:mgmt=`*`access-mgmtType`*

`Enterasys:version=1:mgmt=`*`access-mgmtType`*`:policy=`*`policyname`*

policyname is the name of the policy to apply to this authentication.

access-mgmtTypes supported are: **ro** (read-only), **rw** (read-write), and **su** (super-user).

The un-decorated Filter-ID supports the policy attribute only in the following format:

`policyname`

The undecorated format is simply a string that specifies a policy profile name. The undecorated format cannot be used for management access authentication. Decorated Filter-IDs are processed first. If no decorated Filter-IDs are found, then undecorated Filter-IDs are processed. If multiple Filter-IDs are found that contain conflicting values, a Syslog message is generated.

RADIUS Authentication Retransmission Algorithm

There are three RADIUS authentication algorithms:

Standard – RADIUS authentication always uses the primary (lowest server ID) RADIUS server if it is reachable. If a network outage occurs or server capacity is exceeded, secondary RADIUS servers are used. The standard RADIUS authentication algorithm is appropriate when multiple RADIUS servers are used for redundancy as opposed to a scaled provisioning environment.

Round Robin – RADIUS authentications are evenly spread across servers, allowing the load balancing of a large number of authentications across all available RADIUS servers. If a given server goes down, only sessions associated with that server are affected.

Sticky Round Robin – RADIUS attempts to use the same RADIUS server for any given authentication session, but uses round robin assigning a RADIUS server to each unique authentication session. The sticky round robin algorithm is appropriate for devices that support a limited number of sessions such as the Extreme Network Access Controller (NAC).

The RADIUS authentication algorithm setting defaults to standard. Use the **set radius algorithm** command to set RADIUS authentication algorithm globally on the device.

RADIUS Authentication Sticky Round Robin Maximum Sessions

Round robin sessions are only associated with a particular server when using the sticky round robin algorithm. The maximum number of sticky round robin sessions allowed can be configured on a per port basis.The maximum sessions setting can be configured between 0 (no sessions allowed) and the maximum number of users allowed on the system as displayed using the **show multiauth** command. The RADIUS authentication algorithm must be set to sticky round robin using the **set radius algorithm** command.

If **max-sessions** is not specified, the maximum number of sticky round robin sessions supported defaults to the maximum number of users supported on the device as displayed in the show multiauth command.

Use the **set radius max-sessions** command, specifying the **max-sessions** value and the server index, to set the maximum allowed sessions on when the RADIUS authentication algorithm is set to sticky round robin.

RFC 3580

Extreme Networks switches support the RFC 3580 RADIUS tunnel attribute for dynamic VLAN assignment. The VLAN-Tunnel-Attribute implements the provisioning of service in response to a successful authentication. On ports that do not support policy, the packet will be tagged with the VLAN-ID. The VLAN-Tunnel-Attribute defines the base VLAN-ID to be applied to the user.

Dynamic VLAN Assignment

The RADIUS server may optionally include RADIUS tunnel attributes in a RADIUS Access-Accept message for dynamic VLAN assignment of the authenticated end system.

RFC 3580's RADIUS tunnel attributes are often configured on a RADIUS server to dynamically assign users belonging to the same organizational group within an enterprise to the same VLAN, or to place all offending users according to the organization's security policy in a Quarantine VLAN. Tunnel attributes are deployed for enterprises that have end system authentication configured on the network. For example, all engineers can be dynamically assigned to the same VLAN upon authentication, while sales are assigned to another VLAN upon authentication.

The name of the feature on Extreme Networks platforms that implements dynamic VLAN assignment through the receipt of RADIUS tunnel attributes is VLAN authorization. VLAN authorization depends upon receipt of the RFC 3580 RADIUS tunnel attributes in RADIUS Access-Accept messages. VLAN authorization must be enabled globally and on a per-port basis

for the Tunnel attributes to be processed. When disabled per port or globally, the device will not process Tunnel attributes.

By default, all policy-capable Extreme Networks platforms will dynamically assign a policy profile to the port of an authenticating user based on the receipt of the Filter-ID RADIUS attribute. This is not the case for RADIUS tunnel attributes in that, by default, VLAN authorization is disabled.

VLAN Authorization Attributes

Three Tunnel attributes are used for dynamic VLAN Authorization:

- Tunnel-Type attribute (Type=64, Length=6, Tag=0, Value=0x0D for VLAN)
- Tunnel-Medium-Type attribute (Type=65, Length=6, Tag=0, Value=0x06 for 802 media)
- Tunnel-Private-Group-ID attribute (Type=81, Length>=3, String=VID in ASCII)

The Tunnel-Type attribute indicates the tunneling protocol to be used when this attribute is formatted in RADIUS Access-Request messages, or the tunnel protocol in use when this attribute is formatted in RADIUS Access-Accept messages. Set Tunnel-Type attribute parameters as follows:

- Type: Set to 64 for Tunnel-Type RADIUS attribute
- Length: Set to 6 for six-byte length of this RADIUS attribute
- Tag: Provides a means of grouping attributes in the same packet which refer to the same tunnel. Valid values for this field are from 0x01 through 0x1F, inclusive. Set to 0 if unused. Unless alternative tunnel types are provided, it is only necessary for tunnel attributes to specify a single tunnel. As a result, where it is only desired to specify the VLAN-ID, the tag field should be set to zero (0x00) in all tunnel attributes.
- Value: Indicates the type of tunnel A value of 0x0D (decimal 13) indicates that the tunneling protocol is a VLAN.

Tunnel-Medium-Type indicates the transport medium to use when creating a tunnel for the tunneling protocol, determined from Tunnel-Type attribute. Set Tunnel-Medium-Type attribute parameters as follows:

- Type: Set to 65 for Tunnel-Medium-Type RADIUS attribute
- Length: Set to 6 for six-byte length of this RADIUS attribute
- Tag: Provides a means of grouping attributes in the same packet which refer to the same tunnel. Valid value for this field are 0x01 through 0x1F, inclusive. Set to 0 if unused. Unless alternative tunnel types are provided, it is only necessary for tunnel attributes to specify a single tunnel. As a result, where it is only desired to specify the VLANID, the tag field should be set to zero (0x00) in all tunnel attributes.
- Value: Indicates the type of tunnel. A value of 0x06 indicates that the tunneling medium pertains to 802 media (including Ethernet)

Tunnel-Private-Group-ID attribute indicates the group ID for a particular tunneled session. Set the Tunnel-Private-Group-ID attribute parameters as follows:

- Type: Set to 81 for Tunnel-Private-Group-ID RADIUS attribute
- Length: Set to a value greater than or equal to 3.
- Tag: Provides a means of grouping attributes in the same packet which refer to the same tunnel. Valid values for this field are from 0x01 through 0x1F, inclusive. Set to 0 if unused. Unless alternative tunnel types are provided, it is only necessary for tunnel attributes to specify a single tunnel. As a result, where it is only desired to specify the VLANID, the tag field should be set to zero (0x00) in all tunnel attributes.

- String: Indicates the group. For the VLAN ID integer value, it is encoded as a string using ASCII. For example, the VLAN ID integer value 103 would be represented as 0x313033

VLAN Authorization Considerations

VLAN Authorization poses some operational and management issues on the network.

- A VLAN is not a security container. It is a broadcast container and used to segment broadcast traffic on the network. ACLs implemented at the layer 3 routed interface for a VLAN only provide access control for traffic into and out of the VLAN. No access control mechanism for intra-VLAN communications exists, therefore users within the VLAN are not protected from each other. Malicious traffic allowed onto a VLAN can potentially infect all traffic on the VLAN. Such an infection can consume valuable hardware resources on the infrastructure, such as CPU cycles and memory. Infections can be transmitted to other hosts within the VLAN and to the layer 3 routed boundary. This leads to the direct competition of malicious traffic with business critical traffic on the network.
- End-To-End QoS cannot be truly guaranteed if QoS is implemented at the layer 3 routed interface for a network where business critical applications are classified and prioritized.
- If VLANs are implemented to group together users that are members of the same organizational group, then a VLAN must be configured everywhere in the network topology where a member of that organizational unit may connect to the network. For example, if an engineer may connect to the network from any location, then the Engineering VLAN must be configured on all access layer devices in the network. These VLAN configurations lead to over-extended broadcast domains as well as added configuration complexity in the network topology.
- A problem with moving an end system to a new VLAN is that the end system must be issued an IP address on the new VLAN's subnet to which it has become a member. If the end system does not yet have an IP address, this is not usually a problem. However, if the end system has an IP address, the lease of the address must time out before it attempts to obtain a new address, which may take some time. The IP address assignment process, implemented by DHCP, and the authentication process are not conjoined on the end system. Therefore, this leads to end systems possessing an invalid IP address after dynamic VLAN Authorization and lost IP connectivity until its current IP address times out. Furthermore, when a new IP address is eventually assigned to the end system, IP connectivity is disrupted for all applications on the end system.

Policy Maptable Response

The policy maptable response, or conflict resolution, feature allows you to define how the system should handle allowing an authenticated user onto a port based on the contents of the RADIUS Accept message reply. There are three possible response settings: tunnel mode, policy mode, or both tunnel and policy, also known as hybrid authentication mode.

When the maptable response is set to **tunnel** mode, the system will use the tunnel attributes in the RADIUS reply to apply a VLAN to the authenticating user and will ignore any Filter-ID attributes in the RADIUS reply. When tunnel mode is configured, VLAN-to-policy mapping can occur.

When the maptable response is set to **policy** mode, the system will use the Filter-ID attributes in the RADIUS reply to apply a policy to the authenticating user and will ignore any tunnel attributes in the RADIUS reply. When policy mode is configured, no VLAN-to-policy mapping will occur.

When the maptable response is set to **both**, or hybrid authentication mode, both Filter-ID attributes (dynamic policy assignment) and tunnel attributes (dynamic VLAN assignment) sent in RADIUS Accept message replies are used to determine how the switch should handle authenticating users. When hybrid authentication mode is configured, VLAN-to-policy mapping can occur, as described below in When Policy Maptable Response is "Both".

Using hybrid authentication mode eliminates the dependency on having to assign VLANs through policy roles — VLANs can be assigned by means of the tunnel attributes while policy roles can be assigned by means of the Filter-ID attributes. Alternatively, VLAN-to-policy mapping can be used to map policies to users using the VLAN specified by the tunnel attributes, without having to configure Filter-ID attributes on the RADIUS server. This separation gives administrators more flexibility in segmenting their networks beyond the platform's policy role limits.

When Policy Maptable Response is "Both"

Hybrid authentication mode uses both Filter-ID attributes and tunnel attributes. To enable hybrid authentication mode, use the **set policy maptable** command and set the **response** parameter to **both**. When configured to use both sets of attributes:

- If both the Filter-ID and tunnel attributes are present in the RADIUS reply, then the policy profile specified by the Filter-ID is applied to the authenticating user, and if VLAN authorization is enabled globally and on the authenticating user's port, the VLAN specified by the tunnel attributes is applied to the authenticating user.

  If VLAN authorization is not enabled, the VLAN specified by the policy profile is applied. See "RFC 3580" on page 56-11 for information about VLAN authorization.

- If the Filter-ID attributes are present but the tunnel attributes are not present, the policy profile specified by the Filter-ID is applied, along with the VLAN specified by the policy profile.

- If the tunnel attributes are present but the Filter-ID attributes are not present, and if VLAN authorization is enabled globally and on the authenticating user's port, then the switch will check the VLAN-to-policy mapping table (configured with the **set policy maptable** command):

  – If an entry mapping the received VLAN ID to a policy profile is found, then that policy profile, along with the VLAN specified by the policy profile, will be applied to the authenticating user.

  – If no matching mapping table entry is found, the VLAN specified by the tunnel attributes will be applied to the authenticating user.

  – If the VLAN-to-policy mapping table is invalid, then the etsysPolicyRFC3580MapInvalidMapping MIB is incremented and the VLAN specified by the tunnel attributes will be applied to the authenticating user.

  If VLAN authorization is not enabled, the tunnel attributes are ignored.

When Policy Maptable Response is "Profile"

When the switch is configured to use only Filter-ID attributes, by setting the **set policy maptable** command **response** parameter to **policy**:

- If the Filter-ID attributes are present, the specified policy profile will be applied to the authenticating user. If no Filter-ID attributes are present, the default policy (if it exists) will be applied.

- If the tunnel attributes are present, they are ignored. No VLAN-to-policy mapping will occur.

When Policy Maptable Response is "Tunnel"

When the switch is configured to use only tunnel attributes, by setting the **set policy maptable** command **response** parameter to **tunnel**, and if VLAN authorization is enabled both globally and on the authenticating user's port:

- If the tunnel attributes are present, the specified VLAN will be applied to the authenticating user. VLAN-to-policy mapping can occur.

- If the tunnel attributes are not present, the default policy VLAN will be applied; if the default policy VLAN is not configured, the port VLAN will be applied.
- If the Filter-ID attributes are present, they are ignored.

If VLAN authorization is not enabled, the user will be allowed onto the port with the default policy, if it exists. If no default policy exists, the port VLAN will be applied.

Configuring Authentication

This section provides details for the configuration of authentication methods, MultiAuth and RADIUS.

Table 56-1 lists Authentication parameters and their default values.

Table 56-1 Default Authentication Parameters

Parameter	Description	Default Value
auto-tracking agent authallocated	Specifies the maximum number of users per port supported by the auto-tracking agent	Number of Multiauth users configured.
auto-tracking agent port idle timeout	Specifies the auto-tracking agent port idle timeout value in seconds	Multiauth port idle timeout.
auto-tracking agent port session timeout	Specifies the auto-tracking agent port session timeout in seconds.	Multiauth port session timeout.
auto-tracking agent state	Enables or disables the auto-tracking agent on a per port basis and globally on the device.	Disabled.
authentication agent accounting	Specifies whether accounting is enabled or disabled for the agent.	Quarantine – Disabled. 802.1x – Enabled. PWA – Enabled. MAC-based authentication – Enabled. CEP – Disabled. Auto-tracking – Disabled.
cep port	Enables or disables CEP for the specified port.	Disabled.

Table 56-1 Default Authentication Parameters (continued)

Parameter	Description	Default Value
dot1x	Enables and disables 802.1x authentication both globally and per port.	Globally: Disabled. Per Port: Enabled.
dot1x authconfig	Configures 802.1x authentication.	auto - auto authorization mode.
macauthentication	Globally enables or disables MAC authentication on a device.	Disabled.
macauthentication authallocated	Sets the number of MAC authentication sessions supported on the specified port.	Based upon the device and license. See the firmware release notes for your device.
macauthentication port	Enables or disables MAC authentication on a port	Disabled.
MultiAuth idle-timeout	Specifies the period length for which no traffic is received before a MultiAuth session is set to idle.	300 seconds.
MultiAuth mode	Globally sets MultiAuth for this device.	strict - authentication limited to 802.1x for a single user on a port.
MultiAuth port mode	Specifies the MultiAuth port mode to use for the specified port.	auth-opt - Authentication is optional based upon global and port configuration.
MultiAuth precedence	Specifies the authentication mode to use when multiple authentication types are successfully authenticated.	Precedence from high to low: Quarantine agent, 802.1x, PWA, MAC, CEP, Radius-Snooping, auto-tracking.
MultiAuth session-timeout	Specifies the maximum amount of time a session can live.	0 - no timeout in effect.
pwa	Globally enables or disables PWA authentication.	Disabled.
pwa enhancemode	Allows a user on an un-authenticated port to enter any URL in the browser to access the login page.	Disabled.
quarantine agent authallocated	Specifies the maximum number of users per port for supported by the quarantine agent	Number of Multiauth users configured.
quarantine agent port idle timeout	Specifies the quarantine agent port idle timeout value in seconds	Multiauth port idle timeout.
quarantine agent port session timeout	Specifies the quarantine agent port session timeout in seconds.	Multiauth port session timeout.
quarantine agent state	Enables or disables the quarantine agent on a per port basis and globally on the device.	Disabled.
radius	Enable or disable RADIUS on this device.	Disabled.
radius accounting	Enables or disables RADIUS accounting for this device.	Disabled.

Table 56-1 Default Authentication Parameters (continued)

Parameter	Description	Default Value
radius accounting intervalminimum	Specifies the minimum interval before sending updates for RADIUS accounting.	600 seconds.
radius accounting retries	Specifies the number of times a switch will attempt to contact an authentication server for RADIUS accounting that is not responding.	2.
radius accounting timeout	Specifies the amount of time for a switch to make contact with a RADIUS server.	5 seconds.
radius accounting updateinterval	Specifies the minimum interval between interim updates for RADIUS accounting.	1800 seconds.
radius authentication algorithm	The algorithm used for selecting the server used for a RADIUS authentication session.	standard.
radius retries	Specifies the number of times a switch will try to establish with the authentication server.	3.
RADIUS sticky round robin maximum sessions	The maximum number of RADIUS authentication sessions allowed when the RADIUS authentication algorithm is set to sticky round robin.	maximum number of users supported on the device.
radius timeout	Specifies the amount of time a switch will wait to receive a response from the authentication server before sending another request.	20 seconds.
realm	Specifies authentication server configuration scope	Both: management-access and network-access.
VLAN authorization	Enables or disables globally and per port VLAN authorization.	Globally: Disabled. Per Port: Enabled.
VLAN egress format	Determines whether dynamic VLAN tagging will be none, tagged, untagged, or dynamic for an egress frame.	Untagged.

Configuring Quarantine Agent

Configuring the Quarantine agent on an authenticator switch port consists of:

- Setting the quarantine agent state globally and per port
- Setting optional quarantine agent parameters

Procedure 56-1 describes how to configure the quarantine agent on an authenticator switch port. Unspecified parameters use their default values.

Procedure 56-1 Quarantine Agent Configuration

Step	Task	Command(s)
1.	Enable the quarantine agent on the switch.	**set quarantine-agent enable**
2.	Enable the quarantine agent on each port.	**set quarantine-agent port enable** *port-string*
3.	Optionally, modify the maximum number of quarantine agent sessions allowed on the port. Defaults to number of Multiauth users setting.	**set quarantine-agent port authallocated** *num-users port-string*
4.	Optionally, modify the idle timeout value in seconds. Defaults to the Multiauth idle timeout value.	**set quarantine-agent port idle-timeout** *idle-timeout port-string*
5.	Optionally, modify the session timeout value in seconds. defaults to the global multiauth session timeout.	**set quarantine-agent port session-timeout** *session-timeout port-string*
6.	Optionally, globally enable quarantine agent accounting.	**set quarantine-agent accounting** {**enable** | **disable**}
7.	Optionally display global quarantine agent state and the ports with quarantine agent enabled.	**show quarantine-agent** [**port** *port-string*]

Configuring IEEE 802.1x

Configuring IEEE 802.1x on an authenticator switch port consists of:

- Setting the authentication mode globally and per port
- Configuring optional authentication port parameters globally and per port
- Globally enabling 802.1x authentication for the switch

Procedure 56-2 describes how to configure IEEE 802.1x on an authenticator switch port. Unspecified parameters use their default values.

Procedure 56-2 IEEE 802.1x Configuration

Step	Task	Command(s)
1.	Set the IEEE 802.1x authentication mode both globally and per port: • **Auto** - The switch will only forward authenticated frames. • **Forced-auth** - 802.1x authentication is effectively disabled for this port. All received frames are forwarded. • **Forced-unauth** - 802.1x authentication is effectively disabled on the port. If 802.1x is the only authentication method on the port, all frames are dropped.	**set dot1x auth-config authcontrolled-portcontrol** {**auto** | **forced-auth** | **forced-unauth**}

Procedure 56-2 IEEE 802.1x Configuration (continued)

Step	Task	Command(s)
	Note: Before enabling 802.1x authentication on the switch, you must set the authentication mode of ports that will not be participating in 802.1x authentication to forced-authorized to assure that frames will be forwarded on these ports. Examples of this kind of port are connections between switches and connections between a switch and a router. See the *Extreme Networks S-Series CLI Reference* for a listing of parameter options that come with this command.	
2.	Display the access entity index values. Ports used to authenticate and authorize supplicants utilize access entities that maintain entity state, counters, and statistics for an individual supplicant. You need to know the index value associated with a single entity to enable, disable, initialize, or reauthenticate a single entity.	**show dot1x auth-session-stats**
3.	Enable IEEE 802.1x globally on the switch. Ports default to enabled.	**set dot1x** {**enable** | **disable**} [*port-string*] [**index** *index-list*]
4.	If an entity deactivates due to the supplicant logging off, inability to authenticate, or the supplicant or associated policy settings are no longer valid, you can reinitialize a deactivated access entity. If necessary, reinitialize the specified entity.	**set dot1x init** [**index** *index-list*]
5.	If the authentication for a supplicant times out or is lost for any reason, you can reauthenticate that supplicant. If necessary, reauthenticate the specified entity.	**set dot1x reauth** [**index** *index-list*]
6.	Optionally, globally disable 802.1x agent accounting.	**set dot1x accounting** {**enable** | **disable**}
7.	Display IEEE 802.1x configuration.	**show dot1x auth-config**

Configuring MAC-based Authentication

Configuring MAC-based authentication on a switch consists of:

- Setting the global MAC authentication password for the switch
- Optionally setting the number of MAC authentication sessions allowed on a port
- Enabling MAC authentication on a port
- Enabling MAC authentication globally
- Setting the authentication mode to multi
- Optionally reinitializing or reauthenticating existing sessions

Procedure 56-3 describes how to configure MAC-based authentication. Unspecified parameters use their default values.

Procedure 56-3 MAC-Based Authentication Configuration

Step	Task	Command(s)
1.	Optionally set or clear a global password on the switch.	**set macauthentication password** *password* **clear macauthentication password** *password*
2.	Set or clear the number of MAC authentication sessions supported on a port.	**set macauthentication authallocated** *number port-string*
3.	Enable or disable MAC authentication on a port. By default, MAC authentication is disabled for all ports. MAC authentication must be enabled on the ports that will use it.	**set macauthentication port** {**enable** | **disable**}
4.	Set the authentication mode for the credentials sent to the authentication server.	**set macauthentication auth-mode** {**password** | **radius-username**}
5.	Enable or disable MAC authentication globally on the device. By default, MAC authentication is globally disabled on the device.	**set macauthentication** {**enable** | **disable**}
6.	Set the MultiAuth mode.	**set multiauth mode multi**
7.	Optionally, globally disable MAC-based authentication accounting.	**set macauthentication accounting** {**enable** | **disable**}
8.	Display MAC authentication configuration or status of active sessions.	**show macauthentication** **show macauthentication session**
9.	If a session or port requires reinitialization, reinitialize a specific MAC session or port.	**set macauthentication macinitialize** *mac-address* **set macauthentication portinitialize** *port-string*
10.	If a session or port requires reauthentication, reauthenticate a specific MAC session or port.	**set macauthentication macreauthenticate** *mac-address* **set macauthentication portreauthenticate** *port-string*

Configuring Port Web Authentication (PWA)

Configuring PWA on the switch consists of:

- Setting the IP address which the user will authenticate to on the switch
- Optionally enabling PWA enhanced mode and configure guest networking privileges (not supported when auto-tracking is enabled)
- Enabling PWA on the port
- Globally enabling PWA on the switch
- Setting the authentication mode

Procedure 56-4 describes how to configure PWA authentication. Unspecified parameters use their default values.

Procedure 56-4 Port Web Authentication (PWA) Configuration

Step	Task	Command(s)
1.	Set the IP address for the end-station the supplicant accesses.	**set pwa ipaddress** *ip-address*
2.	Optionally enable or disable PWA enhanced mode.	**set pwa enhancemode enable** **set pwa enhancemode disabled**
3.	Enable or disable PWA. PWA must be enabled on the port for PWA to function.	**set pwa portcontrol enable** *port-string* **set pwa portcontrol disable** *port-string*
4.	Globally enable or disable PWA on the switch.	**set pwa enable** **set pwa disabled**
5.	Optionally, globally disable PWA authentication accounting.	**set pwa accounting** {**enable** | **disable**}
6.	Set the MultiAuth mode.	**set multiauth mode multi**
7.	Display PWA configuration.	**show pwa**

Optionally Enable Guest Network Privileges

With PWA enhanced mode enabled, you can optionally configure guest networking privileges. Guest networking allows an administrator to specify a set of credentials that will, by default, appear on the PWA login page of an end station when a user attempts to access the network. When enhanced mode is enabled, PWA will use a guest password and guest user name to grant network access with default policy privileges to users without established login names and passwords.

In order to configure guest networking privileges, you need to set the guest status, user name, and password. You can set guest status for no authentication, RADIUS authentication, or disabled. When you set guest status to no authentication, guest status is provided with its associated policy, but no authentication takes place. When you set guest status to RADIUS authentication, guest status is provided only after a successful authentication takes place. If guest networking status is disabled, all supplicants must be authenticated with a valid user name and password at the login page.

Table 56-2 describes how to optionally enable guest networking privileges.

Table 56-2 PWA Guest Networking Privileges Configuration

Task	Command(s)
Optionally enable guest status without authentication	**set pwa gueststatus authnone**
Optionally enable guest status with authentication.	**set pwa gueststatus authradius**
Optionally disable guest status	**set pwa gueststatus disable**

Configuring Convergence End Point (CEP)

Configuring CEP consists of:

- Creating a CEP detection group for Non-Cisco Detection CEP types
- Enabling the CEP group for Cisco Detection
- Setting the CEP policy per CEP type
- Enabling CEP on the port

- Setting the authentication mode

Creating a CEP Detection Group

CEP detection groups can be created, deleted, enabled, or disabled. You create a CEP detection group by associating an ID with the create command. Once a group is created, you associate a CEP type, IP address, protocol, and high or low protocol port to it. The type can be H.323, Siemens, or SIP. The IP address is the IP address of the CEP device. By default, H.323 will use 224.0.1.41 as its IP address and Siemens will have no IP address configured. The protocol can be TCP or UDP. The high or low protocol port is the maximum or minimum TCP or UDP port to be used by the group.

Procedure 56-5 describes the creation of a CEP detection group.

Procedure 56-5 CEP Detection Group Configuration

Step	Task	Command(s)
1.	Create a new CEP detection group or enable, disable, or delete an existing group.	**set cep detection-id** *id* {**create** | **enable** | **disable** | **delete**}
2.	Specify the CEP type to be associated with the this group.	**set cep detection-id** *id* **type** {**h323 | siemens | sip**}
3.	Specify the CEP device IP address and mask or set to unknown.	**set cep detection-id** *id* **address** {*ip-address* | **unknown**} **mask** {*mask* | **unknown**}
4.	Set the CEP detection group protocol.	**set cep detection-id** *id* **protocol** {**tcp** | **udp** | **both** | **none**}
5.	Set the maximum or minimum port for the TCP or UDP group protocol.	**set cep detection-id** *id* {**porthigh** | **portlow**} *port*

Procedure 56-6 describes the steps to configure CEP.

Procedure 56-6 CEP Configuration

Step	Task	Command(s)
1.	Determine the policy profile index of the profile you wish to associate with a CEP type.	**show policy profile all**
2.	Associate a policy profile with a CEP type.	**set cep policy** {**cisco** | **h323** | **lldp-med** | **siemens** | **sip**} *policy-index*
3.	Enable or disable the CEP device port for the CEP type	**set cep port** *port-string cep-type* **enable** **set cep port** *port-string cep-type* **disable**
4.	If you are using the Cisco discovery protocol, enable the Cisco discovery protocol. You can also optionally set the voice VLAN ID, whether tagged traffic is trusted or untrusted, and 802.1X priority transmitted to the Cisco IP phone to format in the 802.1Q VLAN tag of its VoIP traffic.	**set ciscodp port** { [**status** {**disable** | **enable**}] [**vvid** {*vlan-id* | **none** | **dot1p** | **untagged**}] [**trust-ext** {**trusted** | **untrusted**}] [**cos-ext** *value*] } *port-string*
5.	If the Cisco discovery protocol is enabled on any port, enable the Cisco discovery protocol globally.	**set ciscodp status**
6.	Globally enable or disable CEP on the switch.	**set cep enable** **set cep disable**
7.	Optionally, globally enable CEP agent accounting.	**set cep accounting** {**enable** | **disable**}

Procedure 56-6 CEP Configuration (continued)

Step	Task	Command(s)
8.	Set the MultiAuth mode.	**set multiauth mode multi**
9.	Display CEP connections, detection, policy and port settings.	**show cep** {**connections** | **detection** | **policy** | **port**}

Setting MultiAuth Idle and Session Timeout for CEP

There is no means of detecting if a Siemens, SIP, or H323 phone goes away other than in the case of a link down. Therefore, if these types of phones are not directly connected to the switch port and the phone goes away, the switch will still see the phone connection and any configured policy will remain on the port. Detected CEPs will be removed from the connection table if they do not send traffic for a time equal to the MultiAuth authentication idle timeout value. CEPs are also removed if the total duration of the session exceeds the time specified in the MultiAuth authentication session timeout.

Procedure 56-7 describes setting the MultiAuth idle and session timeout for CEP.

Procedure 56-7 Configuring MultiAuth Idle and Session Timeouts for CEP

Step	Task	Command(s)
1.	Optionally set the MultiAuth authentication idle timeout for this switch.	**set multiauth idle-timeout cep** *timeout*
2.	Optionally set the MultiAuth authentication session timeout for this switch.	**set multiauth session-timeout cep** *timeout*

Configuring Auto-Tracking

Configuring the auto-tracking agent on an authenticator switch port consists of:

- Setting the auto-tracking agent state globally and per port
- Setting optional auto-tracking agent parameters

Procedure 56-8 describes how to configure the auto-tracking agent on an authenticator switch port. Unspecified parameters use their default values.

Procedure 56-8 Auto-tracking Agent Configuration

Step	Task	Command(s)
1.	Enable the auto-tracking agent on the switch.	**set auto-tracking enable**
2.	Enable the auto-tracking agent on each port.	**set auto-tracking port enable** *port-string*
3.	Optionally, modify the maximum number of auto-tracking agent sessions allowed on the port. Defaults to number of Multiauth users setting.	**set auto-tracking port authallocated** *num-users port-string*
4.	Optionally, modify the idle timeout value in seconds. Defaults to the Multiauth idle timeout value.	**set auto-tracking port idle-timeout** *idle-timeout port-string*
5.	Optionally, modify the session timeout value in seconds. defaults to the global multiauth session timeout.	**set auto-tracking port session-timeout** *session-timeout port-string*

Procedure 56-8 Auto-tracking Agent Configuration (continued)

Step	Task	Command(s)
6.	Optionally, globally enable auto-tracking agent accounting.	**set auto-tracking accounting** {**enable** | **disable**}
7.	Optionally, configure a RADIUS timeout profile on a per port basis.	**set auto-tracking port radius-timeout-profile** *profile-id port-string*
8.	Optionally, configure a RADIUS access reject profile on a per port basis.	**set auto-tracking port radius-reject-profile** *profile-id port-string*
9.	Optionally display global auto-tracking agent state and the ports with the auto-tracking agent enabled.	**show auto-tracking** [**port** *port-string*]

Configuring MultiAuth Authentication

Setting MultiAuth Authentication Mode

MultiAuth authentication mode can be set to MultiAuth or strict 802.1X single user mode. Set MultiAuth authentication to MultiAuth when multiple users need to be authenticated for 802.1X or in all cases for quarantine agent, MAC, PWA, CEP, and auto-tracking agent authentication.

Procedure 56-9 describes setting the MultiAuth authentication mode.

Procedure 56-9 MultiAuth Authentication Configuration

Step	Task	Command(s)
1.	For a single user, single authentication 802.1x port configuration, set MultiAuth mode to strict.	**set multiauth mode strict**
2.	For multiple user 802.1x authentication or any non-802.1x authentication, set the system authentication mode to use multiple authenticators simultaneously.	**set multiauth mode multi**
3.	To clear the MultiAuth authentication mode.	**clear multiauth mode**

Setting MultiAuth Authentication Precedence

MultiAuth authentication administrative precedence globally determines which authentication method will be selected when a user is successfully authenticated for multiple authentication

methods on a single port. When a user successfully authenticates more than one method at the same time, the precedence of the authentication methods will determine which RADIUS-returned filter ID will be processed and result in an applied traffic policy profile.

MultiAuth authentication precedence defaults to the following order from high to low: quarantine agent, 802.1x, PWA, MAC, CEP, Radius-Snooping, auto tracking. You may change the precedence for one or more methods by setting the authentication methods in the order of precedence from high to low. Any methods not entered are given a lower precedence than the methods entered in their pre-existing order. For instance, if you start with the default order and only set quarantine agent, PWA and MAC, the new precedence order will be quarantine agent, PWA, MAC, 802.1x, CEP, and auto-tracking.

Note: It is highly recommended that if you are using the quarantine agent authentication method that it always have the highest precedence. It is also highly recommended that you keep the auto tracking authentication method at the lowest precedence.

Given the default order of precedence (quarantine, 802.1x, PWA, MAC, CEP, and auto-tracking), if a user was to successfully authenticate with PWA and MAC, the authentication method RADIUS Filter-ID applied would be PWA, because it has a higher position in the order. A MAC session would authenticate, but its associated RADIUS Filter-ID would not be applied. If no other authentication method successfully authenticated, the auto-tracking agent would authenticate and an auto-tracking session initiated. The session would authenticate based upon the contents of the admin-policy, if an admin-policy exists.

Procedure 56-10 describes setting the order for MultiAuth authentication precedence.

Procedure 56-10 MultiAuth Authentication Precedence Configuration

Step	Task	Command(s)
1.	Set a new order of precedence for the selection of the RADIUS filter ID that will be returned when multiple authentication methods are authenticated at the same time for a single user.	**set multiauth precedence** {[**quarantine-agent**] [**dot1x**] [**pwa**] [**mac**] [**cep**] [**radius-snooping**] [**auto-tracking**]}
2.	Reset the order MultiAuth authentication precedence to the default values.	**clear multiauth precedence**

Setting MultiAuth Authentication Port Properties

MultiAuth authentication supports the configuration of MultiAuth port and maximum number of users per port properties. The MultiAuth port property can be configured as follows:

- **Authentication Optional** – Authentication methods are active on the port based upon the global and port authentication method. Before authentication succeeds, the current policy role applied to the port is assigned to the ingress traffic. This is the default role if no authenticated user or device exists on the port. After authentication succeeds, the user or device is allowed to access the network according to the policy information returned from the authentication server, in the form of the RADIUS Filter-ID attribute, or the static configuration on the switch. This is the default setting.
- **Authentication Required** – Authentication methods are active on the port, based on the global and per port authentication method configured. Before authentication succeeds, no traffic is forwarded onto the network. After authentication succeeds, the user or device gains access to the network based upon the policy information returned by the authentication server in the form of the RADIUS Filter-ID attribute, or the static configuration on the switch.
- **Force Authenticated** – The port is completely accessible by all users and devices connected to the port, all authentication methods are inactive on the port, and all frames are forwarded onto the network.

- **Force Unauthenticated** – The port is completely closed for access by all users and devices connected to the port. All authentication methods are inactive and all frames are discarded.

Procedure 56-11 describes setting the MultiAuth authentication port and maximum user properties.

Procedure 56-11 MultiAuth Authentication Port and Maximum User Properties Configuration

Step	Task	Command(s)
1.	Set the specified ports to the MultiAuth authentication optional port mode.	**set multiauth port mode auth-opt** *port-string*
2.	Set the specified ports to the MultiAuth authentication required port mode.	**set multiauth port mode auth-reqd** *port-string*
3.	Set the specified ports to the MultiAuth authentication force authenticated port mode.	**set multiauth port mode force-auth** *port-string*
4.	Set the specified ports to the MultiAuth authentication force unauthenticated port mode.	**set multiauth port mode force-unauth** *port-string*
5.	Optionally set the maximum number of authenticated users for the specified port. **Notes:** This value can be set to any value up to the maximum number of MultiAuth users supported for the device. See the firmware release notes that come with your device for the maximum number of supported MultiAuth users the device supports.	**set multiauth port mode numusers** *numusers port-string*
6.	Reset the ports MultiAuth authentication port mode to the default value for the specified ports.	**clear multiauth port mode** *port-string*
7.	Reset the ports MultiAuth authentication port maximum number of users to the default value for the specified ports.	**clear multiauth port numusers** *port-string*

Setting MultiAuth Authentication Timers

The idle timeout setting determines the amount of idle time in which no traffic transits the link for a user or device before the connection is removed from the connection table. The idle timeout can be set for any authentication method.

The session timeout setting determines the maximum amount of time a session can last before being terminated.

Procedure 56-12 describes setting the MultiAuth authentication timers.

Procedure 56-12 MultiAuth Authentication Timers Configuration

Step	Task	Command(s)
1.	Optionally set the MultiAuth authentication idle timeout value for the specified authentication method.	**set multiauth idle-timeout** *auth-method timeout*
2.	Reset the MultiAuth authentication idle timeout value to its default value for the specified authentication method.	**clear multiauth idle-timeout** *auth-method*

Procedure 56-12 MultiAuth Authentication Timers Configuration (continued)

Step	Task	Command(s)
3.	Optionally set the maximum amount of time a session can last before termination for the specified authentication method.	**set multiauth session-timeout** *auth-method timeout*
4.	Reset the maximum amount of time a session can last before termination to the default value for the specified authentication method.	**clear multiauth session-timeout** *auth-method*

Setting MultiAuth Authentication Traps

Traps can be enabled at the system and module levels when the maximum number of users for the system and module, respectively, have been reached. Traps can be enabled at the port level for authentication success, failure, termination and when the maximum number of users have been reached on the port or all supported traps.

Procedure 56-13 describes setting the MultiAuth authentication traps.

Procedure 56-13 MultiAuth Authentication Traps Configuration

Step	Task	Command(s)
1.	Optionally enable MultiAuth authentication system traps.	**set multiauth trap system** {**enabled** | **disabled**}
2.	Optionally enable MultiAuth authentication module traps.	**set multiauth trap module** {**enabled** | **disabled**}
3.	Optionally enable MultiAuth authentication port traps.	**set multiauth trap port** *port-string* {**all** | **success** | **failed** | **terminated** | **max-reached**}
4.	Disable MultiAuth authentication traps for the specified trap type.	**clear multiauth trap** {**system** | **module** | **port** *portstring* {**all** | **success** | **failed** | **terminated** | **max-reached**}}

Setting the MultiAuth Re-Authentication Timeout Action

When attempting to re-authenticate and the RADIUS server is unavailable, the re-authentication attempt times out and by default the session terminates. The re-authentication timeout action can be set to none. If the re-authentication timeout action is set to none, the session remains authenticated and provisioned according to the prior successful RADIUS authentication response.

It is recommended that you not set the re-authentication timeout action to none when using 802.1x authentication.

Use the **set multiauth reauth-timeout-action** {**terminate** | **none**} command to set the re-authentication timeout action for the device.

Displaying MultiAuth Configuration Information

MultiAuth authentication supports the display of system-wide MultiAuth authentication values, MultiAuth authentication counters, port settings, end-user MAC addresses, session information, idle timeout settings, session timeout settings, and trap settings.

Table 56-3 describes displaying of MultiAuth authentication settings and statistics.

Table 56-3 MultiAuth Authentication Settings and Statistics Display

Task	Command(s)
Display system-wide MultiAuth authentication values.	**show multiauth**
Display MultiAuth authentication counters.	**show multiauth counters** [[**quarantine-agent | cep** | **dot1x** | **mac** | **pwa** | **radius-snooping | auto-tracking**] [**chassis**] | **port** *port-string*]]
Display MultiAuth authentication port settings for all or the specified ports.	**show multiauth port** [*port-string*]
Display end-user MAC addresses per port for all MAC addresses and ports or for those specified.	**show multiauth station** [*mac-address*] [*port-string*]
Display MultiAuth authentication sessions for all sessions or the specified authentication method, MAC address, or ports.	**show multiauth session** [**all**] [**agent** {**quarantine-agent | dot1x** | **mac** | **pwa** | **cep** | **radius-snooping | auto-tracking**}] [**mac** *address*] [**port** *port-string*]
Display MultiAuth authentication idle timeout values.	**show multiauth idle-timeout**
Display MultiAuth authentication session timeout values.	**show multiauth session-timeout**
Display MultiAuth authentication trap settings.	**show multiauth trap**

Configuring VLAN Authorization

VLAN authorization allows for the dynamic assignment of users to the same VLAN. You configure VLAN authorization attributes within RADIUS. On the switch you enable VLAN authorization both globally and per-port. VLAN authorization is disabled globally by default. VLAN authorization is enabled per port by default. You can also set the VLAN egress format per-port. VLAN egress format defaults to un-tagged.

VLAN egress format can be set as follows:

- **none** – No egress manipulation will be made.
- **tagged** – The authenticating port will be added to the current tagged egress for the VLAN-ID returned.
- **untagged** – The authenticating port will be added to the current untagged egress for the VLAN-ID returned.
- **dynamic** – Egress formatting will be based upon information contained in the authentication response.

The VLAN authorization table will always list any tunnel attribute's VIDs that have been received for authenticated end systems, but a VID will not actually be assigned unless VLAN authorization is enabled both globally and on the authenticating port. Dynamic VLAN authorization overrides the port PVID. Dynamic VLAN authorization is not reflected in the **show port vlan** display. The VLAN egress list may be statically configured, enabled based upon the **set vlanauthorization egress** command, or have dynamic egress enabled to allow full VLAN membership and connectivity.

Procedure 56-14 describes setting VLAN authorization configuration.

Procedure 56-14 VLAN Authorization Configuration

Step	Task	Command(s)
1.	Enable or disable VLAN authorization both globally and per port.	**set vlanauthorization** {**enable** | **disable**}

Procedure 56-14 VLAN Authorization Configuration (continued)

Step	Task	Command(s)
2.	Reset VLAN authorization configuration to default values for the specified port-list or for all.	**clear valanauthorization** {*port-list* | **all**}
3.	Display VLAN authorization configuration settings for the specified port-list or for all.	**show vlanauthorization** {*port-list* | **all**}

Setting Dynamic Policy Profile Assignment and Invalid Policy Action

Dynamic policy profile assignment is implemented using the policy mapping table. When VLAN authorization is enabled, authenticated users are dynamically assigned to the received tunnel attribute's VID, unless preempted by a policy map-table configuration entry. Dynamic policy profile assignment is supported by mapping a VID to a policy role upon receipt of a RADIUS tunnel attribute.

If the authentication server returns an invalid policy or VLAN to a switch for an authenticating supplicant, an invalid action of forward, drop, or default policy can be configured.

Procedure 56-15 describes setting dynamic policy profile assignment and invalid policy action configuration.

Procedure 56-15 Policy Profile Assignment and Invalid Action Configuration

Step	Task	Command(s)
1.	Identify the profile index to be used in the VID-to-policy mapping.	**show policy profile all**
2.	Map the VLAN ID to the profile index.	**set policy maptable** {*vlan-list profile-index* | **response** {**tunnel** | **policy** | **both**}}
3.	Display the current maptable configuration.	**show policy maptable.**
4.	Set the action to take when an invalid policy or VLAN is received by the authenticating switch.	**set policy invalid action** {**default-policy** | **drop** | **forward**}

Configuring RADIUS

You can set, clear, and display RADIUS configuration for both authentication and accounting.

Configuring the Authentication Server

There are four aspects to configuring the authentication server:

- **State** enables or disables the RADIUS client for this switch.
- **Establishment values** configure a timer setting the length of time before retries, as well as the number of retries, before the switch determines the authentication server is down and attempts to establish with the next server in its list.
- **Server identification** provides for the configuration of the server IP address and index value. The index determines the order in which the switch will attempt to establish a session with an authentication server. After setting the index and IP address you are prompted to enter a secret value for this authentication server. Any authentication requests to this authentication server must present the correct secret value to gain authentication.
- The **realm** provides for configuration scope for this server: management access, network access, or both.

The S-Series firmware supports the configuration of multiple ASs. The lowest index value associated with the server determines the primary server. If the primary server is down, the operational server with the next lowest index value is used. If the switch fails to establish contact with the authentication server before a configured timeout, the switch will retry for the configured number of times.

Servers can be restricted to management access or network access authentication by configuring the realm option.

Procedure 56-16 describes authentication server configuration.

Procedure 56-16 Authentication Server Configuration

Step	Task	Command(s)
1.	Configure the index value, IP address, and secret value for this authentication server.	**set radius server** *index ip-address* [*secret-value*]
2.	Optionally set maximum number of sticky round robin authentication sessions allowed for either the specified RADIUS server or all RADIUS servers	**set radius max-sessions max-sessions** {*index* | **all**}
3.	Optionally set the number of seconds the switch will wait before retrying authentication server establishment.	**set radius timeout** *timeout*
4.	Optionally set the number of retries that will occur before the switch declares an authentication server down.	**set radius retries** *retries*
5.	Optionally set the authentication server configuration scope to management access, network access, or both for all or the specified authentication server.	**set radius realm** {**management-access** | **network-access** | **any**} {*as-index* | **all**}
6.	Optionally set the RADIUS authentication algorithm method for RADIUS server selection.	**set radius algorithm** {**standard** | **round-robin** | **sticky-round-robin**}
7.	Globally enable or disable RADIUS on the switch.	**set radius** {**enable | disable**}
8.	Reset the specified RADIUS setting to its default value.	**clear radius** {[**state**] [**retries**] [**timeout**] [**server** [*index* | **all**] [**realm** {*index* | **all**}]
9.	Display the current RADIUS authentication server settings.	**show radius** [**state | retries | authtype | timeout | server** [*index* **| all**]]

Configuring RADIUS Accounting

There are four aspects to configuring RADIUS accounting:

- **State** enables or disables RADIUS accounting
- **Update values** allow the specification of the length of the period before accounting updates start and the interval between updates
- **Establishment values** configure a timer setting the length of time before retries, as well as the number of retries, before the switch determines the RADIUS accounting server is down and attempts to establish with the next server in its list.
- **Server identification** provides for the configuration of the RADIUS accounting server IP address and index value. The index determines the order in which the switch will attempt to establish with an accounting server. After setting the index and IP address you are prompted to enter a secret value for this accounting server.

Firmware supports the configuration of multiple RADIUS accounting servers. The lowest index value associated with the server determines the primary server. If the primary server is down, the operational server with the next lowest index value is used. If the switch fails to establish contact with the primary server before a configured timeout, the switch will retry for the configured number of times.

Procedure 56-17 describes RADIUS accounting configuration.

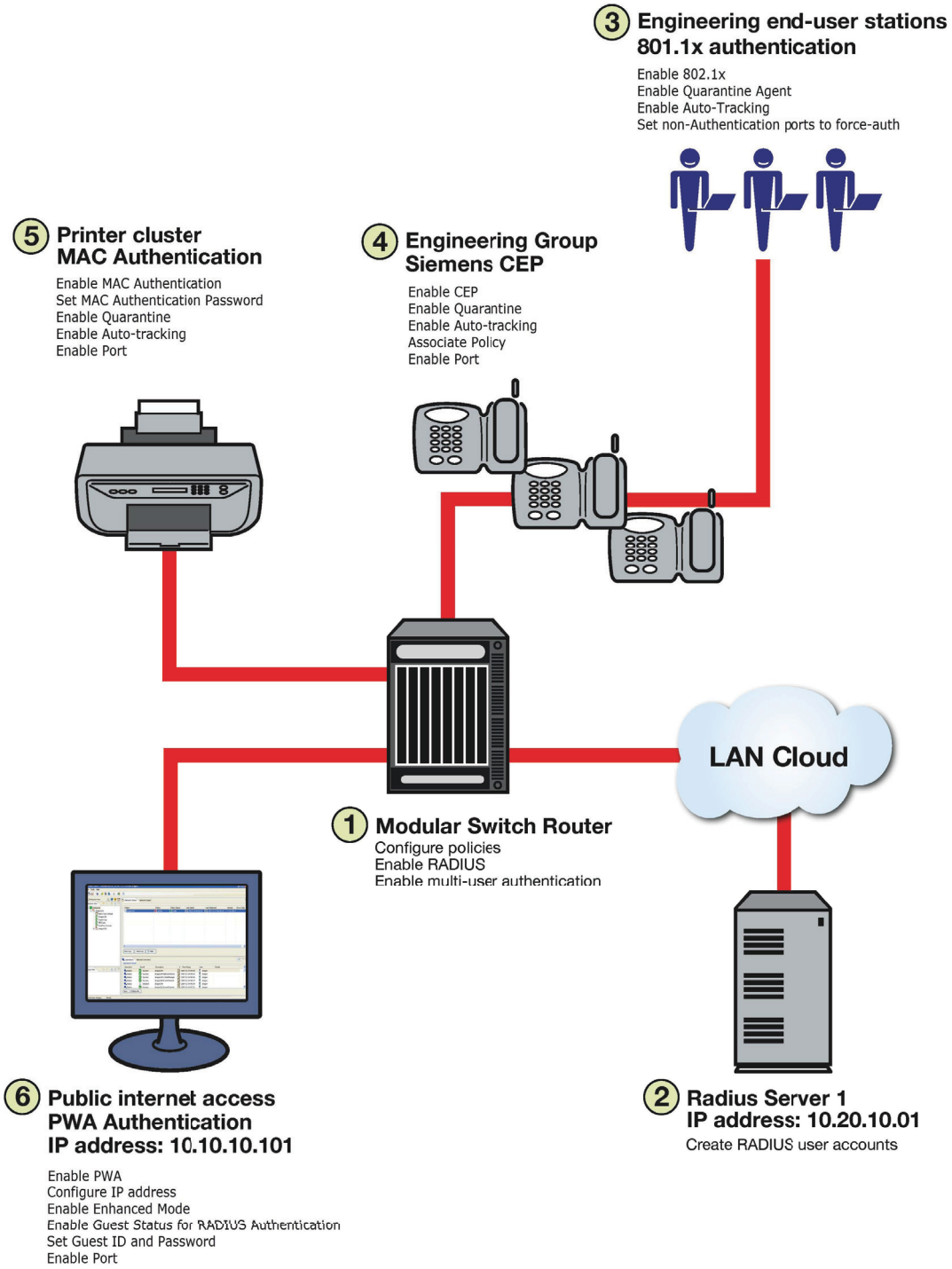
Procedure 56-17 RADIUS Accounting Configuration

Step	Task	Command(s)
1.	Set the minimum interval at which RADIUS accounting sends interim updates.	**set radius accounting intervalminimum** *interval*
2.	Set the number of seconds between each RADIUS accounting interim update.	**set radius accounting updateinterval** *interval*
3.	Set the number of times a switch will attempt to contact a RADIUS accounting server.	**set radius accounting retries** *retries*
4.	Set the amount of time to establish contact with a RADIUS accounting server before timing out.	**set radius accounting timeout** *timeout* {*index* | **all**}
5.	Configure the RADIUS accounting server.	**set radius accounting server** {*index* | **all**} *ip_address udp-port* [*server-secret*]
6.	Enable or disable RADIUS accounting on this switch.	**set radius accounting** {**enable** | **disable**}
7.	Reset RADIUS accounting parameters to default values or clear server definitions on this switch.	**clear radius accounting** {[**server**{*index* | **all**}] [**retries** {*index* | **all**}] [**timeout** {*index* | **all**}] [**intervalminimum**] [**updateinterval**]}
8.	Display RADIUS accounting configuration or statistics.	**show radius accounting** [**updateinterval** | **intervalminimum** | **state** | **server** {*index* | **all**}]

Authentication Configuration Example

Our example covers six supported authentication types being used in an engineering group context: an end-user station, an IP phone, a printer cluster, and public internet access, along with both the quarantine and auto-tracking agents turned on in appropriate places.

Figure 56-4 Authentication Configuration Example Overview



Our configuration example consists of the following steps as shown in Figure 56-4 and described in the sections that follow:

1. Configuring policies, RADIUS, and MultiAuth authentication on the switch.
2. Creating RADIUS user accounts on the authentication server.
3. Configuring for the engineering group 802.1x end-user stations.
4. Configuring for the engineering group Siemens CEP devices.
5. Configuring for the printer cluster MAC authentication.
6. Configuring for the public area internet access for PWA.

Configuring the Quarantine Agent

We will enable the quarantine agent on the switch, but only turn it on for the 802.1x, MAC, and CEP authentication contexts.

For a quarantine policy example we will create a rule for forwarding UDP source port 67 which is normally used for DHCP traffic and associate it with the dhcpQuarantine policy profile. We want to disable any port not connected to a DHCP server if attempts to forward any DHCP traffic occurs. The following CLI input

- Enables the quarantine agent on the switch
- Sets policy rule 1 to hit when forwarding traffic occurs on UDP source port 67 and specifies that the rule is assigned to policy profile 1 as a quarantine profile.
- Names the policy profile dhcpQuarantine and sets the action to disable the port should a hit occur for rule 1.

We enable the quarantine agent at the port level within the appropriate 802.1, MAC, and CEP authentication discussions.

```
S Chassis(rw)->set quarantine-agent enable
S Chassis(rw)->set policy rule 1 udpportsourceip 67 mask 16 forward
quarantine-profile 1
S Chassis(rw)->set policy profile 1 name dhcpQuarantine disable-port enable
```

Configuring the Auto-Tracking Agent

We will enable the auto-tracking agent on the switch, but only turn it on for the 802.1x, MAC, and CEP authentication contexts. We want to enable enhanced mode for PWA. PWA enhanced mode is not supported if auto-tracking is enabled on the port. If you configure an admin-policy, it will be used should auto-tracking be selected as the authenticator. If you do not configure an admin-policy, the device will authenticate, the session will be logged, and no action will occur.

The following CLI input enables the auto-tracking agent on the switch. We enable the auto-tracking agent at the port level within the appropriate 802.1, MAC, and CEP authentication discussions.

```
S Chassis(rw)->set auto-tracking enable
```

Setting MultiAuth Configuration On the Switch

MultiAuth authentication must be set to **multi** whenever multiple users of 802.1x need to be authenticated or whenever any non-802.1X authentication method is present. For ports where no authentication is present, such as switch to switch, or switch to router connections, you should also set MultiAuth port mode to force authenticate to assure that traffic is not blocked by a failed

authentication. For purposes of this example, we will limit authentication to a maximum of 6 users per port.

The following CLI input:

- Sets MultiAuth authentication to **multi**.
- Sets ports with switch to switch and switch to router connections to force authenticate.
- Sets the maximum number of users that can authenticate on each port to 6.

```
S Chassis(rw)->set multiauth mode multi
S Chassis(rw)->set multiauth port mode force-auth ge.1.5-7
S Chassis(rw)->set multiauth port numusers 6 ge.1.5-7
S Chassis(rw)->set multiauth port mode force-auth ge.1.19-24
S Chassis(rw)->set multiauth port numusers 6 ge.1.19-24
```

- Enables MultiAuth authentication system and module traps for the S-Series configuration.

```
S Chassis(rw)->set multiauth trap system enabled
S Chassis(rw)->set multiauth trap module enabled
```

This completes the MultiAuth authentication configuration piece for this example. Keep in mind that you would want to use the **set multiauth precedence** command, to specify which authentication method should take precedence, should you have a single user configured for multiple authentications on the same port.

Enabling RADIUS On the Switch

The switch needs to be informed about the authentication server. Use the following CLI input to

- Configure the authentication server IP address on the switch.
- Enable the RADIUS server.

```
S Chassis(rw)->set radius server 1 10.20.10.01
S Chassis(rw)->set radius enable
```

Creating RADIUS User Accounts On The Authentication Server

RADIUS account creation on the authentication server is specific to the RADIUS application you are using. Please see the documentation that comes with your RADIUS application. Create an account for all users to be authenticated.

Configuring the Engineering Group 802.1x End-User Stations

There are three aspects to configuring 802.1x for the engineering group:

- Configure EAP on each end-user station
- Set up an account in RADIUS on the authentication server for each end-user station
- Configure the quarantine agent, 802.1x, and the auto-tracking agent on the switch

Configuring EAP on the end-user station and setting up the RADIUS account for each station is dependent upon your operating system and the RADIUS application being used, respectively. The important thing the network administrator should keep in mind is that these two configurations should be in place before moving on to the 802.1x configuration on the switch. In an 802.1x configuration, policy is specified in the RADIUS account configuration on the authentication server using the RADIUS Filter-ID. See “The RADIUS Filter-ID” on page 56-10 for RADIUS

Filter-ID information. If a RADIUS Filter-ID exists for the user account, the RADIUS protocol returns it in the RADIUS Accept message and the firmware applies the policy to the user.

Note: Globally enabling 802.1x on a switch sets the port-control type to **auto** for all ports. Be sure to set port-control to **forced-auth** on all ports that will not be authenticating using 802.1x and no other authentication method is configured. Otherwise these ports will fail authentication and traffic will be blocked.

The following CLI input:

- Enables 802.1x on the switch
- Sets port-control to **forced-auth** for all connections between switches and routers, because they do not use authentication and would be blocked if not set to **forced-auth**
- Enables the quarantine agent on ports ge.1.5, ge.1.19, and ge.1.24
- Enables the auto-tracking agent on the switch and ports ge.1.5, ge.1.19, and ge.1.24

```
S Chassis(rw)->set dot1x enable
S Chassis(rw)->set dot1x auth-config authcontrolled-portcontrol forced-auth
ge.1.5
S Chassis(rw)->set dot1x auth-config authcontrolled-portcontrol forced-auth
ge.1.19
S Chassis(rw)->set dot1x auth-config authcontrolled-portcontrol forced-auth
ge.2.24
S Chassis(rw)->set quarantine-agent port enable ge.1.5
S Chassis(rw)->set quarantine-agent port enable ge.1.19
S Chassis(rw)->set quarantine-agent port enable ge.1.24
S Chassis(rw)->set auto-tracking enable
S Chassis(rw)->set auto-tracking port enable ge.1.5
S Chassis(rw)->set auto-tracking port enable ge.1.19
S Chassis(rw)->set auto-tracking port enable ge.1.24
```

This completes the 802.1x end-user stations configuration.

Configuring the Engineering Group Siemens CEP Devices

If a Siemens phone is inserted into a port enabled for Siemens CEP, the firmware detects communication on UDP/TCP port 4060. Use policy manager to configure a policy with a VLAN, CoS, and rate limit appropriate to VoIP. See Chapter 53, **Quality of Service (QoS) Configuration** for a QoS VoIP policy configuration example. Once an existing policy is configured, the **set cep policy command can be used to apply** the policy

The following CLI input:

- Enables CEP globally on the switch.
- Sets CEP policy to a previously configured policy named **siemens** with an index of **9**.
- Sets ports **ge.1.16-18** to only accept default Siemens type phones and applies the Siemens policy to the specified ports.
- Enable the quarantine and auto-tracking agents on ports **ge.1.16** through **ge.1.18**

```
S Chassis(rw)->set cep enable
S Chassis(rw)->set cep policy siemens 9
S Chassis(rw)->set cep port ge.1.16-18 siemens enable
S Chassis(rw)->set quarantine-agent port enable ge.1.16-18
```

```
S Chassis(rw)->set auto-tracking port enable ge.1.16-18
```

This completes the Siemens CEP end-user stations configuration.

Configuring the Printer Cluster for MAC-Based Authentication

Perform the following tasks to configure MAC-based authentication for the printer cluster in our example:

- Set up an account for each printer on the authentication server that contains the printer MAC address, the MAC authentication password configured on the switch, and a RADIUS filter ID entry specifying the printer policy.
- Configure a policy using the policy manager specifying the printer cluster VLAN and optionally configuring a CoS and rate limit.
- Enable MAC authentication globally on the switch.
- Enter the MAC authentication password as **enterasys** on the switch.
- Set the MAC authentication significant-bits to **24**.
- Enable MAC authentication on the ports used by the printer cluster: **ge.1.3-4**
- Enable the quarantine and auto-tracking agents on ports **ge.1.3-4**

With the authentication server configured with a RADIUS account for each printer, and the printer policy preconfigured, enter the following CLI input:

```
S Chassis(rw)->set macauthentication enable
S Chassis(rw)->set macauthentication password enterasys
S Chassis(rw)->set macauthentication significant-bits 24
S Chassis(rw)->set macauthentication port enable ge.1.3-4
S Chassis(rw)->set quarantine-agent port enable ge.1.3-4
S Chassis(rw)->set auto-tracking port enable ge.1.3-4
```

This completes the printer cluster MAC authentication configuration.

Configuring the Public Area PWA Station

The public area PWA station provides visitors to your business site with open access to the internet, while at the same time isolating the station from any access to your internal network. In order to provide a default set of network resources to communicate over HTTP, policy must be set to only allow DHCP, ARP, DNS, and HTTP. You may want to set a rate limit that would guard against excessive streaming. You will also need to set up RADIUS for the public station account on the authentication server. This configuration will include the guest name, password, and a RADIUS Filter-ID for the public policy. We will not enable auto-tracking because PWA enhanced mode is not supported with auto-tracking. We will also not enable quarantine.

Perform the following tasks to configure the public station for PWA authentication:

- Configure the policy appropriate to the public station.
- Setup the RADIUS user account for the public station on the authentication server.
- Enable PWA globally on the switch.
- Configure the IP address for the public station.
- Optionally set up a banner for the initial PWA screen.
- Enable PWA enhancemode so that any URL input will cause the PWA sign in screen to appear.

- Set PWA gueststatus to RADIUS authentication mode.
- Set the PWA login guest name.
- Set the PWA login password.
- Enable PWA on the switch port where the public station is connected.

Once the policy and RADIUS account are configured, enter the following CLI input on the switch:

```
S Chassis(rw)->set pwa enable
S Chassis(rw)->set pwa ipaddress 10.10.10.101
S Chassis(rw)->set pwa banner \"Extreme Networks Public Internet Access Station\"
S Chassis(rw)->set pwa enhancemode enable
S Chassis(rw)->set pwa gueststatus authradius
S Chassis(rw)->set pwa guestname guest
S Chassis(rw)->set pwa guestpassword password
S Chassis(rw)->set pwa portcontrol enable ge.1.6
```

This completes the Authentication configuration example.

Terms and Definitions

Table 56-4 lists terms and definitions used in this Authentication configuration discussion.

Table 56-4 Quality of Service Configuration Terms and Definitions

Term	Definition
Authentication Server (AS)	An entity providing authorization services to an authenticator using RADIUS. The authentication server may be on the same device or be at a remote location.
Authenticator	The switch seeking authentication from the authentication server for a supplicant.
Auto-tracking agent	A form of authentication that authenticates those sessions that are not captured by the other supported MultiAuth authentication agents (quarantine, 802.1x, PWA, MAC, CEP, and RADIUS snooping).
Convergence End Point (CEP)	A protocol capable of detecting an IP telephony or video device on a port and dynamically applying a specific policy to the port.
Domain Name System (DNS)	Serves as a means for the Internet to translate human-readable computer hostnames, e.g. www.example.com, into the IP addresses.
Dynamic Host Configuration Protocol (DHCP)	A protocol used by networked clients to obtain various parameters necessary for the clients to operate in an Internet Protocol (IP) network.
Extensible Authentication Protocol (EAP)	A protocol that provides the means for communicating the authentication information in an IEEE 802.1x context.
IEEE 802.1x	An IEEE standard for port-based Network Access Control that provides authentication to devices attached to a LAN port, establishing a point-to-point connection or preventing access from that port if authentication fails.
MAC-based Authentication	A means of authenticating a device attempting to gain access to the network based upon the device MAC address and a secret keyword known to the authenticator and the RADIUS application on the authentication server.
Multi-user Authentication	The ability to appropriately authenticate multiple supplicants on a single link and provision network resources, based upon policy associated with each supplicant.

Table 56-4 Quality of Service Configuration Terms and Definitions (continued)

Term	Definition
MultiAuth Authentication	The ability to authenticate multiple authentication modes for a user and applying the authentication mode with the highest precedence.
Port Web Authentication (PWA)	A means of authenticating a user by utilizing a web browser for the login process to authenticate to the network.
Quarantine agent	A form of authentication that depends upon the existence of one or more configured quarantine policy rules, with each rule associated with a policy profile that determine the action should the quarantine agent be used to authenticate the device.
RADIUS Filter ID	An Extreme Networks proprietary string formatted in the RADIUS Access-Accept packet sent back from the authentication server to the switch containing either the policy to apply to the supplicant, the management type for the port, or both.
RADIUS Protocol	An AAA (Authentication, Authorization, and Accounting) protocol for controlling access to network resources used by ISPs and corporations managing access to Internet or internal networks across an array of access technologies.
Supplicant	The user or device seeking access to network resources.