Enterasys N-Series®

Configuration Guide

Firmware Version 7.31

P/N 9034507-02

enterasys®

Notice

Enterasys Networks reserves the right to make changes in specifications and other information contained in this document and its web site without prior notice. The reader should in all cases consult Enterasys Networks to determine whether any such changes have been made.

The hardware, firmware, or software described in this document is subject to change without notice.

IN NO EVENT SHALL ENTERASYS NETWORKS BE LIABLE FOR ANY INCIDENTAL, INDIRECT, SPECIAL, OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING BUT NOT LIMITED TO LOST PROFITS) ARISING OUT OF OR RELATED TO THIS DOCUMENT, WEB SITE, OR THE INFORMATION CONTAINED IN THEM, EVEN IF ENTERASYS NETWORKS HAS BEEN ADVISED OF, KNEW OF, OR SHOULD HAVE KNOWN OF, THE POSSIBILITY OF SUCH DAMAGES.

Enterasys Networks, Inc.
50 Minuteman Road
Andover, MA 01810

© 2011 Enterasys Networks, Inc. All rights reserved.

Part Number: 9034507-02 September 2011

ENTERASYS, ENTERASYS NETWORKS, ENTERASYS SECURE NETWORKS, ENTERASYS MATRIX, ENTERASYS NETSIGHT, LANVIEW, WEBVIEW, and any logos associated therewith, are trademarks or registered trademarks of Enterasys Networks, Inc., in the United States and/or other countries. For a complete list of Enterasys trademarks, see http://www.enterasys.com/company/trademarks.aspx.

All other product names mentioned in this manual may be trademarks or registered trademarks of their respective companies.

Documentation URL: https://extranet.enterasys.com/downloads

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., 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) ("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 ENTERASYS NETWORKS, LEGAL DEPARTMENT AT (978) 684-1000.

You and Enterasys agree as follows:

1. **LICENSE.** You have the non-exclusive and non-transferable right to use only the 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 upon 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 Commonwealth of Massachusetts without regard to its conflicts of laws provisions. You accept the personal jurisdiction and venue of the Commonwealth of Massachusetts courts. 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 understand that Enterasys and its Affiliates are subject to regulation by agencies of the U.S. Government, including the U.S. Department of Commerce, which prohibit export or diversion of certain technical products to certain countries, unless a license to export the product is obtained from the U.S. Government or an exception from obtaining such license may be relied upon by the exporting party.

If the Program is exported from the United States pursuant to the License Exception CIV under the U.S. Export Administration Regulations, You agree that You are a civil end user of the Program and agree that You will use the Program for civil end uses only and not for military purposes.

If the Program is exported from the United States pursuant to the License Exception TSR under the U.S. Export Administration Regulations, in addition to the restriction on transfer set forth in Section 1 or 2 of this Agreement, You agree not to (i) reexport or release the Program, the source code for the Program or technology to a national of a country in Country

Groups D:1 or E:2 (Albania, Armenia, Azerbaijan, Belarus, Cambodia, Cuba, Georgia, Iraq, Kazakhstan, Laos, Libya, Macau, Moldova, Mongolia, North Korea, the People's Republic of China, Russia, Tajikistan, Turkmenistan, Ukraine, Uzbekistan, Vietnam, or such other countries as may be designated by the United States Government), (ii) export to Country Groups D:1 or E:2 (as defined herein) the direct product of the Program or the technology, if such foreign produced direct product is subject to national security controls as identified on the U.S. Commerce Control List, or (iii) if the direct product of the technology is a complete plant or any major component of a plant, export to Country Groups D:1 or E:2 the direct product of the plant or a major component thereof, if such foreign produced direct product is subject to national security controls as identified on the U.S. Commerce Control List or is subject to State Department controls under the U.S. Munitions List.

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 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 IMPLIED WARRANTIES OF MERCHANTABILITY, SATISFACTORY QUALITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT WITH RESPECT TO THE PROGRAM. IF 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 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.

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, 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 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 audit discovers non-compliance with this Agreement, including copies of the Program made, used or deployed in breach of this Agreement, You shall promptly pay to Enterasys the appropriate license fees. Enterasys reserves the right, to be exercised in its sole discretion and without prior notice, to terminate this 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 suppliers. All rights not specifically granted to You shall be reserved to Enterasys.

10. **ENFORCEMENT.** You acknowledge and agree that any breach of Sections 2, 4, or 9 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.

11. **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 which acquires substantially all of Your stock assets. Enterasys may assign this Agreement in its sole discretion. 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.

12. **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.

13. **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.

14. **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: Port Mirroring Configuration

Chapter 6: System Configuration

Chapter 7: Tracked Object Manager Configuration

Chapter 8: Power over Ethernet Configuration

Chapter 9: Discovery Protocol Configuration

Chapter 10: Simple Network Management Protocol (SNMP) Configuration

Chapter 11: Spanning Tree Configuration

Chapter 12: VLAN Configuration

Chapter 13: Link Aggregation Control Protocol (LACP) Configuration

Chapter 14: Policy Configuration

Chapter 15: Multicast Configuration

Chapter 16: System Logging Configuration

Chapter 17: Network Monitoring Configuration

Chapter 18: NetFlow Configuration

Chapter 19: Virtual Routing and Forwarding (VRF) Configuration

Chapter 20: IP Routing Configuration

Chapter 21: Routing Information Protocol (RIP) Configuration

Chapter 22: Open Shortest Path First (OSPFv2) Configuration

Chapter 23: Network Address Translation (NAT) Configuration

Chapter 24: Load Sharing Network Address Translation (LSNAT) Configuration

Chapter 25: Transparent Web Cache Balancing (TWCB) Configuration

Chapter 26: Virtual Router Redundancy Protocol (VRRP) Configuration

Chapter 27: Security Configuration

Chapter 28: Flow Setup Throttling Configuration

Chapter 29: Access Control List Configuration

Chapter 30: Route-Map Manager Configuration

Chapter 31: Quality of Service (QoS) Configuration

Chapter 32: RADIUS Snooping Configuration

Chapter 33: Authentication Configuration

Procedures

Figures

Tables

About This Guide

This manual explains how to configure Enterasys N-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 N-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:

https://extranet.enterasys.com/downloads

- *Enterasys Matrix N-Series CLI Reference* provides information on how to use the Command Line Interface for the N-Series switch/routers.
- The *Transitioning to Firmware v7.0* document provides a summary of the CLI changes from the S-Series firmware v6.11 and greater to N-Series firmware v7.0.

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 N-Series switch/router or to this document, contact Enterasys Networks using one of the following methods:

World Wide Web	www.enterasys.com/support
Phone	1-800-872-8440 (toll-free in U.S. and Canada) or 1-978-684-1000 For the Enterasys Networks Support toll-free number in your country: www.enterasys.com/support/
Internet mail	support@enterasys.com To expedite your message, please type **[N-SERIES]** in the subject line.

To send comments or suggestions concerning this document to the Technical Publications Department:

techpubs@enterasys.com

To expedite your message, include the document Part Number in the Email message.

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 N-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 N-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 N-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 Enterasys Networks' NetSight management application.
- Remotely using WebView™, Enterasys Networks' embedded web server application.

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

Initial Configuration

To initially configure the N-Series device, you must have connected a terminal to the local console port as described in the *Hardware Installation Guide* for your N-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 *Enterasys Matrix N-Series CLI Reference*.

For module placement rules and considerations for configuring local management on DFE Gold modules, refer to the installation guide that comes with the DFE Gold module.

For details on activating redundancy on a DFE Gold Series module, refer to "About Redundant Management on The DFE-Gold Series Modules" on page 6-9.

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 Enterasys 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 6-10.	**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 N-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 N-Series device for each area.

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

Table 1-1 Advanced Configuration

Task	Refer to page...
System Services	
Configure the Simple Network Time Protocol (SNTP) client.	6-10

Table 1-1 Advanced Configuration (continued)

Task	Refer to page...
Configure the Domain Name Server (DNS) client.	6-18
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.	6-16
Configure the Secure Shell V2 (SSHv2) client and server.	6-17
Configure the Dynamic Host Configuration Protocol (DHCP) client and server.	6-21
Configure the port parameters, such as speed and duplex mode.	4-1
Enable SNMP and create a community string. By default, the SNMP master agent is disabled and no defined public community string is configured.	10-1
Configure RMON to provide comprehensive network fault diagnosis, planning, and performance tuning information, and allow for interoperability between SNMP management stations and monitoring agents.	17-4
Change the interactive login authentication method, from local to remote (RADIUS authentication).	33-2
If RADIUS authentication is configured, configure the remote RADIUS servers to be used by the RADIUS client on the N-Series	33-24
Layer 2 Switching	
Enable desired ports for switching.	4-1
Set port configurations and port-based Virtual Local Area Networks (VLANs). VLANs can be created statically or dynamically.	12-1
Configure Spanning Trees using STP, RSTP, or MSTP.	11-1
Configure LLDP or CDP.	9-1
Layer 3 Routing	
Configure the router id. Refer to the **router id** command in the *Enterasys Matrix N-Series CLI Reference*.	
Configure interfaces for IP routing.	20-3
Configure the ARP table.	20-16
Configure UDP broadcast forwarding, including DHCP/BOOTP relay agent.	20-20
Configure routes.	20-1
Configure interior gateway protocols: RIP and OSPF.	21-1, 22-1
Configure multicast protocols IGMP, DVMRP, and PIM, and general multicast parameters.	15-1
Configure VRRP.	26-1
Configure policy-based routing.	14-1
Security and General Management	
Configure Access Control Lists (ACLs).	29-1
Configure RADIUS servers.	33-24
Manage user accounts and passwords.	6-5
Configure system logging.	16-1

Table 1-1 Advanced Configuration (continued)

Task	Refer to page...
Configure the N-Series using text files.	3-1
Upgrade system firmware.	3-1
Configure QoS features.	31-1
Configure policy.	14-1

2

Using the CLI

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

CLI Conventions

Getting Help with CLI Syntax

The N-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:

```
N 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.

```
N 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
N Chassis(rw)->show snmp
N 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>
N 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**:

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

```
N 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 N-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**.

```
N 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:

```
N Chassis(rw)->conf<SPACEBAR>
N 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 *Enterasys Matrix N-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.

```
N 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 *Enterasys Matrix N-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 N-Series devices.

Configuration and Image File Management on Your System

On N-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 *Enterasys Matrix N-Series CLI Reference* for more information about each command.

Saving a Configuration

You can save the N-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. For Standalone device, always specify slot1.

Example: Creating a Configuration Restore-Point

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

Example: Creating a Configuration File

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

Executing a Configuration

You can execute the N-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 N-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 N-Series device reboots. When the N-Series device reboots, any configuration changes made after the restore-point was set will be lost.	**configure restore-point** *index*
3.	Reboot the N-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**

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

`configure` *`filename`*

filename must include the slotN/ file path.

Example: Executing a Configuration Restore-Point

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

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

Example: Executing a Configuration File

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

Deleting a Configuration Restore-Point or File

You can delete the N-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

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

Example: Deleting a Configuration File

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

Example: Deleting an Image File

```
N 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 N-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 N-Series module to which a configuration file is downloaded must have the same hardware configuration as the N-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:

```
N Chassis(rw)->copy scp://doc@banshee.enterasys.com:22/myconfig slot3/myconfig
Password:
###########
N 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 3-7.

For information on downloading

Example: Downloading an Image File

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

Example: Downloading a Configuration File

```
N 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 N-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: 256 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. The N-Series supports a maximum of two images. If there are two images listed, use the **delete** *filename* command to remove one of the images.

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. 2011
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 N-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

```
N 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

```
N 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 N-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.

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

N 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

N 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 N-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

Port Configuration

This document describes port configuration on Enterasys N-Series devices.

Port Configuration Overview

The Enterasys N-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 or in the case of the Standalone device:

port type.port group.port number

Where **port type** can be:

- **fe** - 100-Mbps Ethernet
- **ge** - 1-Gbps Ethernet
- **tg** - 10-Gbps Ethernet (not supported on the Standalone device)
- **com** - COM (console) port
- **host** - the host port
- **vlan** - VLAN interfaces
- **lag** - IEEE802.3 link aggregation ports
- **lpbk** - loopback interfaces, or
- **lo** - the local (software loopback) interface
- **bp** - FTM1 backplane ports
- **pc** - the internal ports which connect to the on-board processor of an installed Matrix Security Module
- **rtr** - router 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

Slot location for modules installed in a N-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. The notion of a slot location is not appropriate to the Standalone device. See port group information below when using a Standalone device.

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 TGbE 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**.

For Standalone device only, **Port group** can be:
1 for the lower fixed front panel ports
2 for the middle fixed front panel ports, or
3 for the top fixed front panel ports and the Mini-GBIC uplink ports

Port number can be:
Any port number in a port group.

Examples

Note: You can use a wildcard (*) to indicate all of an item. For example, ge.3.* would represent all Gigabit Ethernet ports in the module in slot 3 or port group 3 in the case of the Standalone device.

This example shows the *port-string* syntax for specifying the 100-Mbps Ethernet ports 1 through 10 in the module in chassis slot 1, or in the case of the Standalone device port group 1.

```
ge.1.1-10
```

This example shows the *port-string* syntax for specifying the 1-Gigabit Ethernet port 14 in the module in chassis slot 3, or in the case of the Standalone device port group 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, or in the case of the Standalone device port group 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, or in the case of the Standalone device port group 1:

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

This example shows the *port-string* syntax for specifying the 10-Gigabit Ethernet port 2 of the module in chassis slot 3 (the 10-Gigabit Ethernet port is not supported on the Standalone device):

```
tg.3.2
```

This example shows the *port-string* syntax for specifying all 1-Gigabit Ethernet ports in the module in chassis slot 3, or in the case of the Standalone device port group 3:

```
ge.3.*
```

This example shows the *port-string* syntax for specifying all 10-Gbps Ethernet ports in the chassis (the 10-Gigabit Ethernet port is not supported on the Standalone device):

```
tg.*.*
```

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

```
*.*.*
```

Console Port Parameters

Each Enterasys N-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

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:

```
N 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:

```
N 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:

```
N 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:

```
N 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:

```
N 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:

```
N Chassis(rw)->set port enable ge.1.1
N Chassis(rw)->show port ge.1.1
Port ge.1.1 enabled
N 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 12, **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

```
N Chassis(rw)->set port ingress-filter ge.1.1 enable
N 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**

```
N Chassis(rw)->set port alias ge.1.1 documentation
N 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:

```
N Chassis(rw)->set forcelinkdown enable
N Chassis(rw)->show forcelinkdown
ForceLinkDown feature is globally enabled.
N 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, or 1000 Mbps.

Auto-negotiation is enabled by default.

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

```
N Chassis(rw)->set port speed ge.1.1 100
N Chassis(rw)->show port speed ge.1.1
```

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

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:

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

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

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 on port ge.1.1:

```
N Chassis(rw)->set port jumbo ge.1.1
N Chassis(rw)->show port jumbo ge.1.1

Port Number  Jumbo Oper Status  Jumbo Admin Status  Jumbo MTU
-----------  -----------------  ------------------  ---------
ge.1.1       Disabled           Enabled             10239

N 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:

```
N Chassis(rw)->set port negotiation ge.1.1 enable
N Chassis(rw)->set port advertise ge.1.1 10t 10tfd 100tx 100txfd bpause
N 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
N 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.

```
N Chassis(rw)->set port mdix mdix
N 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:

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

```
N Chassis(rw)->set port trap ge.1.1 enable
N Chassis(rw)->show port trap ge.1.1
Link traps enabled on port ge.1.1.
N 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 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 5 link flaps
- Sets the link flap interval to 10 seconds
- Sets the downtime period to 600 seconds

```
N Chassis(rw)->set linkflap action ge.1.1 all
N Chassis(rw)->set linkflap threshold ge.1.1 5
N Chassis(rw)->set linkflap interval ge.1.1 10
N Chassis(rw)->set linkflap downtime ge.1.1 600
N 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  5           10          600
1 port(s) found.
N 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:

```
N Chassis(rw)->set port broadcast ge.1.1 10000
N 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 Enterasys N-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 *Enterasys Matrix N-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:

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

Port Priority to Transmit Queue Mapping

N-Series module ports support up to 24 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 31-4 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

```
N Chassis(rw)->set port priority-queue ge.1.1 5 1
N 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
N Chassis(rw)->
```

Configuring Ports

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

Table 4-1 lists port parameters and their default values.

Table 4-1 Default Port Parameters

Parameter	Description	Default Value
console baud rate	Specifies the baud rate for the console port.	9600
console flow control	Specifies the flow control mechanism for the console port.	ctsrts (Clear to Send/Request to Send)
console bits	Specifies the number of bits per character on the console port.	8 bits
port state	Specifies the port state.	disabled
port ingress filter	Specifies that frame forwarding is limited to members of the port's VLAN egress list.	disabled
jumbo frame support	Specifies whether Ethernet frame with a payload greater than 1500 is supported on this port.	disabled
port traps	Specifies whether the sending of port traps is enabled on this port.	enabled
global link flap state	Specifies whether link flap is enabled globally on this device	disabled
port priority	Specifies the 802.1D priority for this port.	0

Table 4-1 Default Port Parameters (continued)

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)
port negotiation	Specifies whether auto-negotiation is enabled on this port.	enabled

Procedure 4-1 describes how to configure ports.

Procedure 4-1 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*] | [**flowcontrol** {**none** | **ctsrts** | **dsrdtr**}] | [**parity** {**none** | **odd** | **even** | **mark** | **space**}] | [**stopbits** {**one** | **oneandhalf** | **two**}] } [*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**}
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**}
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 4-2 describes how to configure link trap and link flap detection.

Procedure 4-2 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.	**set linkflap globalstate enable**
3.	Optionally, enable the link flap detection function on one or more ports.	**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 4-2 describes how to manage port configuration.

Table 4-2 Managing Port Configuration

Task	Command
To clear the properties set for one or more console ports to its default values:	**clear console** [**baud**] [**bits**] [**flowcontrol**] [**parity**] [**stopbits**] [*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**]
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**}

Table 4-2 Managing Port Configuration (continued)

Task	Command
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 4-3 describes how to display port configuration information and statistics.

Table 4-3 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**
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 4-4 lists terms and definitions used in this port configuration discussion.

Table 4-4 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.
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 (**.**).

5

Port Mirroring Configuration

This chapter provides the following information about configuring and monitoring port mirroring on N-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.

Note: Each DFE Gold module supports three mirroring destination ports, which can be configured in a many-to-one mirroring configuration (that is, many destinations mirrored to one source port).

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.

The N-Series does not support multicast port mirroring. A static IGMP managed multicast entry can be configured, which forces a multicast group out a specified port, the equivalence of a mirror destination. Use the **set igmp static** command to configure a static IGMP multicast entry. See "Multicast Configuration" on page 15-1 for static IGMP group configuration information.

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)

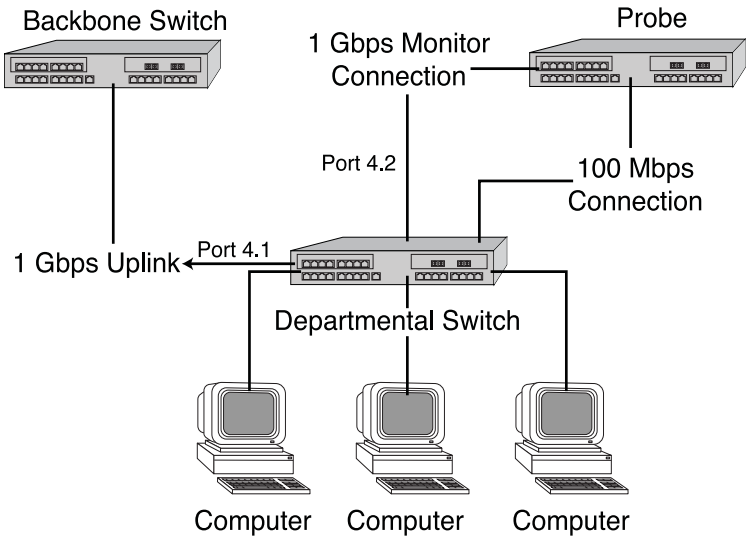
- Many-to-one
- One-to-many

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 5-4.

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 5-1.

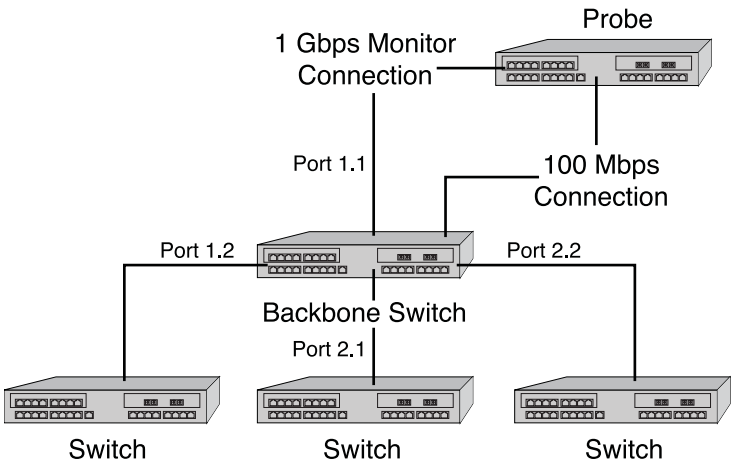
Figure 5-1 Using Port Mirroring to Monitor a Departmental Switch



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 5-2.

Figure 5-2 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 DFE Gold module supports three port mirrors. The Standalone device, DFE Diamond module, and DFE Platinum module supports 16 port mirrors. With the exception of the DFE Gold module which does not support VLANs and IDS port mirroring, these port mirrors can be a mixed variety of port, VLAN, and IDS combinations. Any or all port 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. Examples of port mirroring combinations on a DFE Platinum and Diamond module include:

- 16 port mirrors
- 16 VLAN mirrors
- 8 port and 6 VLAN mirrors
- 12 port and 4 VLAN mirrors
- 15 port and 1 IDS mirror (where the device mirrors to 10 ports)
- 15 VLAN and 1 IDS mirror (where the device mirrors to 10 ports)

Implementing Port Mirroring

You can implement port mirroring on N-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.

You can also use CLI to operationally disable mirroring, if necessary, and to specify whether to mirror received traffic, transmitted traffic, or both.

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 N-Series device. In addition, virtual ports and other types of port configurations can also participate in mirroring on Enterasys switching devices as described in the following sections:

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

LAG Mirrors

Each N-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 13, **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.

Note: The DFE Gold module does not support IDS mirrors.

An N-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 N-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 N-Series device, you must take into consideration the following:

- Only one IDS mirror is allowed per N-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 5-11 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.

Note: The DFE Gold module only supports the creation of port mirrors for destination ports. The DFE Gold module does not support the creation of port mirrors for VLANs.

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.

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
```

Examples

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

```
N Chassis(rw)->show port mirroring
No Port Mirrors configured.
IGMP Multicast Mirror status 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.

```
N 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
```

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.

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:

```
N 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.:

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

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

```
N Chassis(rw)->set vlan interface 5 create
N Chassis(rw)->set port mirroring create vtap.0.5 ge.1.1
N 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 12, **VLAN Configuration** for more information about configuring VLANs.

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 14, **Policy Configuration**.

Deleting Mirrors

Use this command to clear a port mirroring configuration:

```
clear port mirroring source destination
```

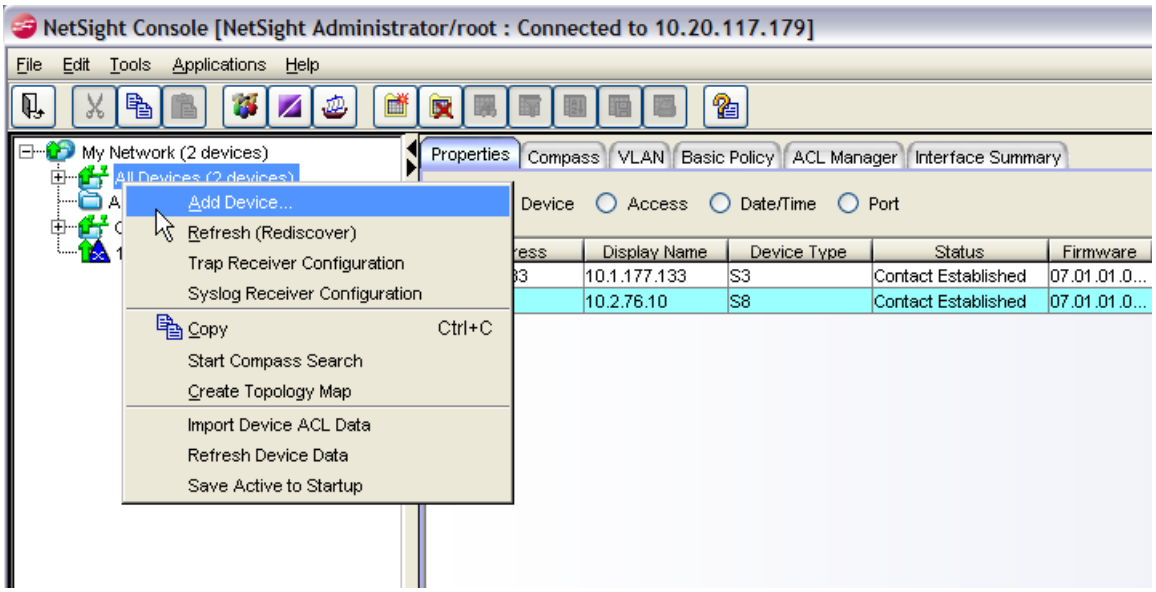
Use this command to clear a policy mirror destination:

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

Example: Configuring and Monitoring Port Mirroring

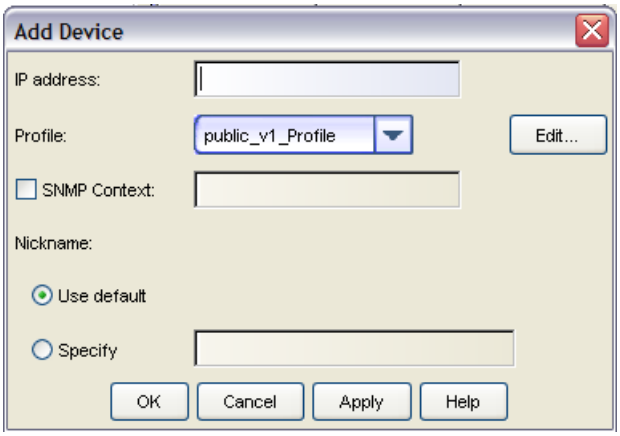
This section describes how to use Enterasys NetSight Console from a Network Management Station (NMS) to display RMON statistics for monitoring port mirroring.

1. Log onto Netsight Console.



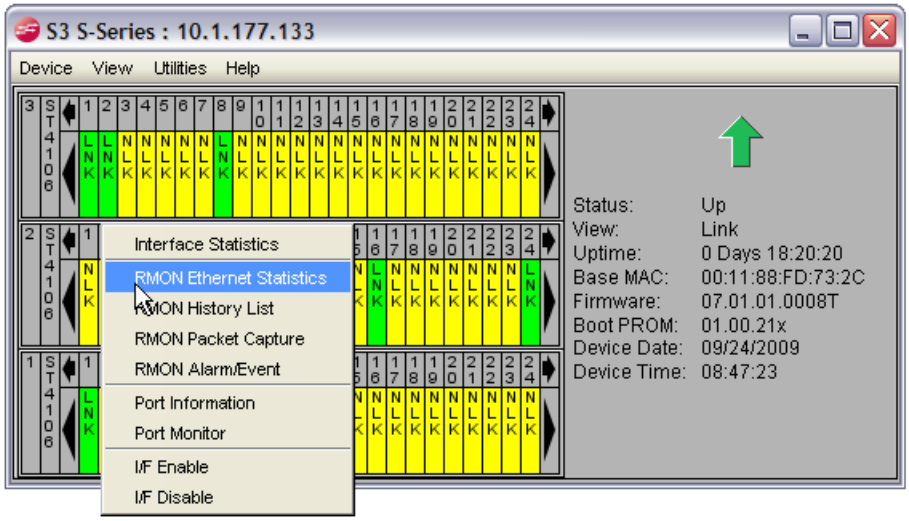
2. On the console main screen, expand **My Network** in the file directory tree, right-click **All Devices**, and select **Add Device**.

   The Add Device screen displays.



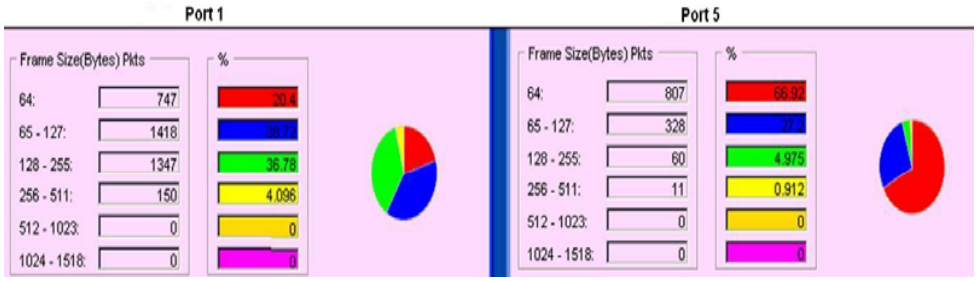
3. Model theN-Series device by entering its **IP address** in the field provided. Click **OK**.
4. On the console main screen, expand **All Devices** in the file directory tree to show the IP address(es) of the device(s) you just modeled.
5. Right click on the IP address of the N-Series device and select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

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

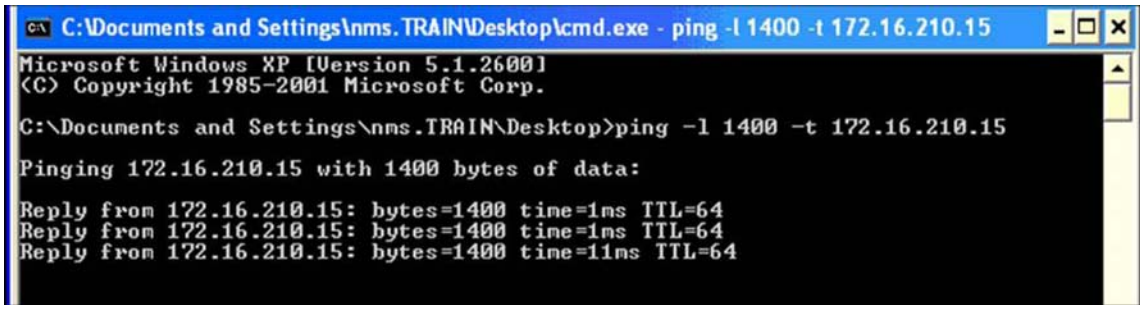


6. Right click on port 1 (ge.1.1) and select **RMON Ethernet Statistics**.
7. Repeat step 9 for port 5 (ge.1.5).

   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 N-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 N-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.

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

12. Verify the mirroring configuration.

```
N 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

N-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 5-1 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 5-1 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

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

6

System Configuration

This document provides the following information about system configuration on the Enterasys N-Series platforms.

System Properties Overview

Table 6-1 lists system parameter default values.

Table 6-1 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
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 N-Series device with an IP interface and an IP address. You can also configure other system properties on the N-Series device. See Table 6-2.

Table 6-2 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*]
Control the gratuitous ARP processing behavior. By default, gratuitous ARP is disabled.	**set ip gratuitous-arp [request|reply|both]**
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*]
(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*]

Table 6-2 System Properties Configuration (continued)

Task	Command
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 6-3 lists system properties management and display commands for N-Series devices.

Table 6-3 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**
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**

Table 6-3 System Properties Management and Display Commands (continued)

Task	Command
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

```
N Chassis(rw)->set ip interface vlan.0.5 default
N Chassis(rw)->set ip address 10.1.10.1 mask 255.255.128.0 gateway 10.1.10.1
N Chassis(rw)->set ip gratuitous-arp both
N Chassis(rw)->set system utilization threshold 1000
N Chassis(rw)->set time 7:50:00
N Chassis(rw)->set summertime enable
N Chassis(rw)->set summertime recurring second Sunday March 02:00 first Sunday
November 02:00 60
N Chassis(rw)->set system name “Information Systems”
N Chassis(rw)->set system location “Bldg N32-04 Closet 9”
N Chassis(rw)->set system contact “Joe Smith”
N Chassis(rw)->set physical alias chassis chassisone
N 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 6-1.

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

User management configuration also includes the following:

- "Setting the Authentication Login Method" on page 6-6
- "Using WebView" on page 6-7

Procedure 6-1 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*] [**allowed-interval** {*HH:MM HH:MM*}] [**allowed-days** {[**Sun**] [**Mon**] [**Tue**] [**Wed**] [**Thu**] [**Fri**] [**Sat**]}] [**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*]
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**}] [**substring-match-len** *characters*] [**allow-repeating-chars** {**yes | no**}] [**change-first-login** {**yes | no**}] [**change-frequency** *minutes*]
4.	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*] [**inactive** *days*]}

Table 6-4 lists user account management and display commands for N-Series devices.

Table 6-4 User Account Management and Display Commands

Task	Command
Display user login account information.	**show system login** [**-verbose**]
Display current password configuration settings.	**show system password**

Table 6-4 User Account Management and Display Commands (continued)

Task	Command
Display settings for locking out users.	**show system lockout**
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**] [**local-only**]
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**] [**substring-match-len**] [**allow-repeating-chars**] [**change-first-login**] [**change-frequency**]

User Management Example

This example includes the following:

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

```
N Chassis(su)->set system login netops read-write enable
N Chassis(su)->set password rw
Please enter new password: ********
Please re-enter new password: ********
Password changed.
N Chassis(su)->set system password age 60 length 6 allow-repeating-chars no
N 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 6-2 Authentication Configuration

Step	Task	Command(s)
1.	Change the default authentication login method.	**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).	**clear authentication login**

Procedure 6-2 Authentication Configuration (continued)

Step	Task	Command(s)
4.	Configure the chosen authentication login method. For more information, see Chapter 27, **Security Configuration** for TACACS+ and Chapter 33, **Authentication Configuration** for RADIUS.	

Using WebView

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

Procedure 6-3 describes how to configure WebView on an N-Series device.

Procedure 6-3 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 10, **Simple Network Management Protocol (SNMP) Configuration**.

Configuring Management Authentication Notification MIB

Procedure 6-4 describes how to configure the Management Authentication Notification MIB on an N-Series device. Management Authentication Notification MIB commands can be entered in any command mode.

By default, all Management Authentication Notification types are enabled.

Procedure 6-4 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**]

Procedure 6-4 Management Authentication Notification MIB Configuration (continued)

Step	Task	Command(s)
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:

```
N Chassis(su)->set mgmt-auth-notify disable
N 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:

```
N Chassis(su)->set mgmt-auth-notify enable console telnet
N 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:

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

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

N Chassis(su)->clear mgmt-auth-notify
```

```
N 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:

- Activation of advanced routing features on N-Series modules and Standalone device
- A DFE Platinum and Diamond module option that activates the N-EOS-PUC user-capacity license, which increases system capacity to 2024 users per N-Series system
- A DFE Platinum and Diamond module option that activates the N-EOS-PPC license, which increases port user capacity to 256 users per port, up to a maximum of 1024 users per N-Series access
- A DFE Gold module option that activates management module redundancy

You must activate the purchased license key.

The license is activated on an N-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.

About Redundant Management on The DFE-Gold Series Modules

Note: Interoperability of DFE Gold Series modules is dependent upon module placement rules during installation in the chassis. For details on these rules and their effects on system management, refer to the installation guide that comes with your DFE Gold series module.

The DFE Gold System Management Module (SMM) coordinates and controls the configuration of the entire chassis. By default, this is the module installed in slot 1. Access to the SMM is available through any console (COM) port on any module in the chassis. Only one CLI session can be active at any one time, and active status is granted to the first connection to any of the console ports.

In order to enable switch and routing redundancy on a DFE Gold Series module, you must purchase and activate a license key. If you have purchased a redundancy license, you can proceed to activate it as described in this section. If you wish to purchase a redundancy license, contact Enterasys Networks Sales.

When a redundancy license key is purchased and activated as described in this section, redundancy can be configured on the module in slot 2 of the chassis. Then, in the event module 1 fails, module 2 will assume chassis management.

Configuring a License

Procedure 6-5 describes how to configure the license on an N-Series device. License commands can be entered in any command mode.

Procedure 6-5 License Configuration

Step	Task	Command(s)
1.	Activate the license on an N-Series device—module or chassis—as applicable.	**Platinum and Diamond:** **set license** {**advanced** | **user-capacity** | **port-capacity**} *license-key* [**slot** *slot*] **Standalone:** **set license advanced** *license-key* [**slot** *slot*] **Gold:** **set license** {**advanced** | **redundancy**} *license-key* [**slot** *slot*]
2.	Display the license key.	**show license**

License Examples

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

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

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

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

License Type      Location  Status      Key
----------------  --------  ----------  ----------------------------------------
advanced          chassis   active      ABCDEF1234567890
user-capacity     chassis   active      ABCDEF1234567890
port-capacity     slot 1    active      ABCDEF1234567890
port-capacity     slot 2    active      ABCDEF1234567890
port-capacity     slot 3    active      ABCDEF1234567890
port-capacity     slot 4    active      ABCDEF1234567890
port-capacity     slot 5    active      ABCDEF1234567890
```

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

```
N 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 N-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 N-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:

```
N 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:

```
N Chassis(rw)->set sntp authentication key 1 md5 foobaraboof
N Chassis(rw)->set sntp authentication key 2 md5 DEADBEAFCAFEBABEDEADBEAFCAFEBAE
N 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:

```
N 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:

```
N Chassis(rw)->set sntp authentication trust 1 enable
N Chassis(rw)->set sntp authentication trust 3 disable
```

Configuring SNTP

This section provides details for the configuration of SNTP on the N-Series products.

Table 6-5 lists SNTP parameters and their default values.

Table 6-5 Default SNTP Parameters

Parameter	Description	Default Value
SNTP authentication mode	Specifies whether authentication for all SNTP client communications is enabled or disabled.	disabled
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 6-6 describes how configure SNTP. SNTP can be configured in any command mode.

Procedure 6-6 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*

Procedure 6-6 Configuring SNTP (continued)

Step	Task	Command(s)
5.	When operating in unicast mode, optionally change the number of poll retries to a unicast SNTP server.	**set sntp poll-retry** *retry*
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 6-6 describes how to manage and display SNTP.

Table 6-6 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

```
N Chassis(rw)->set sntp client broadcast
N Chassis(rw)->set sntp broadcastdelay 3500
N Chassis(rw)->set timezone EDT -4 0
N 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
N 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

```
N Chassis(rw)->set sntp client unicast
N Chassis(rw)->set sntp authentication mode enable
N Chassis(rw)->set sntp authentication key 1 md5 foobar
N Chassis(rw)->set sntp authentication trust 1 enable
N Chassis(rw)->set sntp server 10.21.1.100 key 1
N Chassis(rw)->set sntp poll-interval 600
N Chassis(rw)->set timezone EDT -4 0
N Chassis(rw)->set sntp poll-retry 2
N 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
N 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 6-7 describes how to configure and use Telnet on an N-Series device. Telnet commands can be entered in any command mode.

Procedure 6-7 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 6-7 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:

```
N Chassis(rw)->set telnet enable all
```

The following example shows how to verify the Telnet status:

```
N 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:

```
N 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 N-Series SSHv2 implementation includes:

- Data privacy
- Communication integrity

An SSH server resides on the N-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 N-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 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.

Configuring Secure Shell

Procedure 6-8 describes how to configure Secure Shell on an N-Series device. Secure Shell commands can be entered in any command mode.

Procedure 6-8 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.	Start an SSH session to a remote host.	**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*]
4.	Verify the SSH state.	**show ssh state**

Secure Shell Configuration Examples

The following commands enable and verify SSH:

```
N Chassis(rw)->set ssh enable
N Chassis(rw)->show ssh state
SSH Server state: Enabled
N Chassis(rw)->
```

The following command reinitializes the host key on the SSH server:

```
N 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 N-Series products.

Table 6-7 lists DNS parameters and their default values.

Table 6-7 Default DNS Parameters

Parameter	Description	Default Value
DNS resolver state	Specifies whether DNS resolver is enabled or disabled on the device.	enabled
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 6-9 describes how to configure DNS resolution. DNS can be configured in any CLI command mode.

Procedure 6-9 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*

Procedure 6-9 Configuring DNS Resolution (continued)

Step	Task	Command(s)
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 6-8 describes how manage DNS resolution on an N-Series switch. DNS commands can be configured in any CLI command mode.

Table 6-8 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**
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 **Enterasys.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**

```
N-Series(rw)->set ip dns domain Enterasys.Documentation
N-Series(rw)->set ip dns server 153.50.50.10 primary
N-Series(rw)->set ip dns server 153.50.50.20 secondary
N-Series(rw)->set ip dns timeout 4
N-Series(rw)->set ip dns query-retries 3
N-Series(rw)->show ip dns
Current State:             Enabled
Default DNS domain name:   Enterasys.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
N-Series(rw)->
```

DHCP Overview

The Dynamic Host Configuration Protocol (DHCP) provides services for allocating and delivering IP addresses and other configuration parameters 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 N-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 N-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.

DHCP Supported Options

Table 6-9 on page 6-22 lists the DHCP server option names and codes supported by the firmware. All options specified in Table 6-9 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**.

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.

Configuring DHCP

This section provides details for the configuration of DHCP on the N-Series products.

Table 6-9 lists DHCP parameters and their default values.

Table 6-9 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

Procedure 6-10 describes enabling the DHCP feature and client configuration.

Procedure 6-10 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*

Table 6-10 describes how to configure the router.

Table 6-10 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 6-11 describes client configuration.

Procedure 6-11 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*

Table 6-11 describes how to manage and display DHCP.

Table 6-11 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**

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 N-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**.

```
N Chassis(rw-config)->ip local pool pool1 10.60.1.0 255.255.255.0
N Chassis(rw-config-ip-local-pool)->exclude 10.60.1.10 20
N Chassis(rw-config-ip-local-pool)->exit
N Chassis(rw-config)->ip dhcp pool pool1
N Chassis(rw-config-dhcp-pool)->domain-name MyCompany.com
N Chassis(rw-config-dhcp-pool)->bootfile dhcpboot
N Chassis(rw-config-dhcp-pool)->option 72 ip 10.70.0.10 10.70.0.11 10.70.0.12
N Chassis(rw-config-dhcp-pool)->next-server 10.70.0.12
N Chassis(rw-config-dhcp-pool)->lease 100
N Chassis(rw-config-dhcp-pool)->default-router 1.1.1.1
```

```
N Chassis(rw-config-dhcp-pool)->dns-server 2.2.2.2
N Chassis(rw-config-dhcp-pool)->client-class class1
N Chassis(rw-config-dhcp-class)->default-router 3.3.3.3
N Chassis(rw-config-dhcp-class)->dns-server 4.4.4.4
N Chassis(rw-config-dhcp-class)->exit
NChassis(rw-config-dhcp-pool)->hardware-address 00:11:22:33:44:55 client-class
class1
N Chassis(rw-config-dhcp-host)->dns-server 5.5.5.5
N Chassis(rw-config-dhcp-host)->host 10.60.0.1
N Chassis(rw-config-dhcp-host)->exit
N Chassis(rw-config-dhcp-pool)->exit
N Chassis(rw-config)->
```

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 N-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 N-Series products.

Procedure 6-12 describes how to configure node alias on switch ports.

Procedure 6-12 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 6-12 describes how to display and manage the node alias on the N-Series device.

Table 6-12 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*]

Table 6-12 Managing Node Alias (continued)

Task	Command(s)
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*]
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:

```
N-Series(rw)->set nodealias enable ge.1.1
N-Series(rw)->set nodealias maxentries 100
N-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:

```
N-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.*.*

```
N-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:

```
N Chassis(rw)->set mac agetime 600
N Chassis(rw)->show mac agetime
Aging time: 600 seconds
N 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.

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:

```
N 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.

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:

```
N Chassis(rw)->set mac unicast 00:a4:01:ff:0e:01 1 ge.1.1 ageable
N 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
N 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:

```
N Chassis(rw)->set mac unicast-as-multicast enable
N Chassis(rw)->show mac unicast-as-multicast
Unicast as multicast: enabled
N 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 either globally on the device or on a specified port basis. 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:

```
N Chassis(rw)->set newaddrtrap ge.1.1 enable
N Chassis(rw)->
```

Use the **set movedaddrtrap** command in any command mode to enable SNMP trap messaging to report detection of a moved MAC address either globally on the device or on a specified port basis. 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:

```
N Chassis(rw)->set movedaddrtrap ge.1.1 enable
N Chassis(rw)->
```

Procedure 6-13 describes how to configure MAC address settings. All commands for this feature can be set in any command mode.

Procedure 6-13 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, enable SNMP trap messaging to report the detection of new MAC addresses for the specified port or all ports.	**set newaddrtrap** [*port-string*] {**enable** | **disable**}

Procedure 6-13 Configuring MAC Address Settings (continued)

Step	Task	Command(s)
6.	Optionally, enable SNMP trap messaging to report the detection of a moved MAC address for the specified port or all ports.	**set movedaddrtrap** [*port-string*] {**enable** | **disable**}

Terms and Definitions

Table 6-13 lists terms and definitions used in this system configuration discussion.

Table 6-13 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.
node alias	An N-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.

Table 6-13 System Configuration Terms and Definitions (continued)

Term	Definition
unicast polling	An SNTP operational mode for which the client directly requests updates from the SNTP server.

7

Tracked Object Manager Configuration

This document provides the following information about configuring the tracked object manager on the Enterasys N-Series platform.

Using Tracked Object Manager in Your Network

The tracked object manager is an application that determines the status of a remote service using a probe. Probes track the availability of a remote service by actively transmitting network packets to a specified remote host. Tracked object manager supports three probe protocols:

- 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 tracked object manager supports the configuration of a probe on four N-Series applications. The type of probe used depends upon the N-Series application using it. Configure a probe for:

- Policy Based Routing (PBR) to monitor a next hop IP address using an ICMP probe

- Server Load Balancing (SLB) to monitor an LSNAT real server IP address using an ICMP ping, or a port using TCP or UDP port verification, as well as verify an application running on the real server, by configuring the TCP or UDP probe for ACV
- Transparent Web Cache Balancing (TWCB) to monitor a cache IP address using an ICMP probe or perform port verification on the cache server, by configuring a TCP or UDP probe
- Virtual Router Redundancy Protocol (VRRP) to monitor a critical IP interface using an ICMP probe

Note: Prior to the N-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 N-Series firmware and have been replaced by the tracked object manager feature.

Implementing Probes

To configure a probe:

- Create the probe by specifying a probe name and type
- Optionally configure a description to be associated with this probe
- 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 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

Tracked Object Manager Overview

The three 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 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 the probe configuration mode from the global configuration mode.

Probe 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

Tracked object manager provides for the setting of content verification parameters:

Tracked object manager provides for the setting of content verification parameters:

- **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
- The time, in seconds, the track object manager waits for a response from the monitored service before declaring a failed probe
- The time, in seconds, the track object manger waits for the TCP 3-way handshake to complete

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

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 N-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 either gets data back from the request to the server or 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.

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 either a TCP or UDP probe.

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.enterasys.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 provides probe support for four system applications. A preset default ICMP probe for each supported application exists when you boot your system. Default ICMP probes can not be modified or deleted.

See Table 7-1 for a listing of default preset ICMP probe names, the local application the probe is associated with, and parameter settings.

Table 7-1 Preset Default ICMP Probes

Probe Name	Description
$pbr_default	The default policy based routing ICMP probe. Default values are: • Fail-detect count: 3 • Pass-detect count: 1 • Fail-detect interval: 10 • Pass-detect interval: 5 • 3-way TCP handshake wait time: 5 • Server response wait time: 10
$slb_default	The default server load balancing ICMP probe. Default values are: • Fail-detect count: 4 • Pass-detect count: 1 • Fail-detect interval: 5 • Pass-detect interval: 5 • 3-way TCP handshake wait time: 5 • Server response wait time: 2
$twcb_default	The default TWCB ICMP probe. Default values are: • Fail-detect count: 4 • Pass-detect count: 1 • Fail-detect interval: 5 • Pass-detect interval: 5 • 3-way TCP handshake wait time: 5 • Server response wait time: 2
$vrrp_default	The default VRRP ICMP probe. Default values are: • Fail-detect count: 3 • Pass-detect count: 3 • Fail-detect interval: 1 • Pass-detect interval: 10 • 3-way TCP handshake wait time: 5 • Server response wait time: 3

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.

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**:

```
N 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:

```
N Chassis(rw)->configure
N Chassis(rw-config)->interface vlan 20
N Chassis(rw-config-intf-vlan.0.20)->vrrp critical-ip 1 20.20.20.2 100 remote
probe-name $vrrp_default
N Chassis(rw-config-intf-vlan.0.20)->no shutdown
N 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 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 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:

```
N Chassis(su-config)->probe ICMP-PBR icmp
N Chassis(su-config-probe)->faildetect count 5 interval 5
N Chassis(su-config-probe)->inservice
N Chassis(su-config-probe)->exit
N Chassis(su-config)->route-map probe 101.10.1.252 probe-name ICMP-PBR
N 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 session
...
N 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 7-5 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

```
N Chassis(su)->configure
N Chassis(su-config)->probe TCP-HTTP tcp
N Chassis(su-config-probe)->faildetect interval 5
N Chassis(su-config-probe)->passdetect interval 5
N Chassis(su-config-probe)->acv request “GET / HTTP/1.1\\r\\nHost:
2.0.0.5\\r\\n\\r\\n”
N Chassis(su-config-probe)->acv reply “HTTP/1.1 200 OK\\r\\n”
N Chassis(su-config-probe)->inservice
N 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
N Chassis(su-config-probe)->exit
N Chassis(su-config)->ip slb serverfarm myproductHTTP
N Chassis(su-config-slb-sfarm)->real 10.1.2.3 port 80
N Chassis(su-config-slb-real)->faildetect probe one TCP-HTTP
N Chassis(su-config-slb-real)->inservice
N 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 7-5 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

```
N Chassis(su)->configure
N Chassis(su-config)->probe TCP-HTTP tcp
N Chassis(su-config-probe)->inservice
N Chassis(su-config-probe)->acv request “GET / HTTP/1.1\\r\\nHost:
2.0.0.5\\r\\n\\r\\n”
N Chassis(su-config-probe)->acv reply “HTTP/1.1 200 OK\\r\\n”
N 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
N Chassis(su-config-probe)->exit
N Chassis(su-config)->ip twcb wcserverfarm s1Server
N Chassis(config-twcb-wcsfarm)->cache 186.89.10.51
N Chassis(config-twcb-cache)->faildetect probe one TCP-HTTP
N Chassis(config-twcb-cache)->faildetect app-port 8080
N Chassis(config-twcb-cache)->inservice
N 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 7-5 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

```
N Chassis(su-config)->probe ICMP-VRRP icmp
N Chassis(su-config-probe)->faildetect interval 5
N Chassis(su-config-probe)->passdetect interval 5
N Chassis(su-config-probe)->inservice
N Chassis(su-config-probe)->exit
N Chassis(rw)->configure
N Chassis(rw-config)->interface vlan 20
N Chassis(rw-config-intf-vlan.0.20)->vrrp critical-ip 1 20.20.20.2 10 remote
probe-name ICMP-VRRP
N Chassis(rw-config-intf-vlan.0.20)->no shutdown
N Chassis(rw-config-intf-vlan.0.20)->
```

Configuring Tracked Object Manager

This section provides details for tracked object manager configuration on N-Series products.

Table 7-2 lists tracked object manager default values.

Table 7-2 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

Table 7-2 Default Tracked Object Manager Parameters (continued)

Parameter	Description	Default Value
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 7-1 describes how to configure probes.

Procedure 7-1 Probe Configuration

Step	Task	Command(s)
1.	Create a probe, specifying the probe name and protocol type.	**probe** *probe-name* {**icmp** | **tcp** | **udp**}
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.	Enable the probe by placing it inservice.	**inservice**

Terms and Definitions

Table 7-3 lists terms and definitions used in this tracked object manager configuration discussion.

Table 7-3 Tracked Object Manager Terms and Definitions

Term	Definition
tracked object manager	An application that determines the state of a remote service using a probe.
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.
server port verification	A fail detection method used by server load balancing and TWCB to assure that the remote server is up.
application content verification (ACV)	A 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.
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.
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 track object manager waits for a response from the remote service before declaring a failed probe.

8

Power over Ethernet Configuration

This chapter provides information about configuring and monitoring Power over Ethernet (PoE) on the N-Series devices.

Important Notice

This section applies only to PoE-equipped N-Series devices. Consult the *Hardware Installation Guide* shipped with your product to determine if it is PoE-equipped.

Note: Power over Ethernet is not supported on the N-Series Standalone platform.

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 Enterasys device through the CLI-based procedures presented in the section "Configuring PoE" on page 8-3. 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 Enterasys device can supply enough power to run your PDs. See Table 8-1 for power ranges based on each device class.

Table 8-1 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 Enterasys 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 Enterasys device turns amber. 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 N-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 N-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, with the exception that for class 0 and class 4 devices, actual power consumption will always be used. 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 8-1 on page 8-2.

Configuring PoE

Once you have determined how to implement PoE on your N-Series device, the following sections will help you configure PoE.

Default Settings

Table 8-2 lists PoE parameters and their default values.

Table 8-2 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
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 8-1 describes how to configure PoE. Unspecified parameters use their default values.

Procedure 8-1 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*]}
	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 N-Series device is a Standalone, specify 1 as the *module-number*.	**set inlinepower threshold** *usage-threshold module-number*

Procedure 8-1 PoE Configuration (continued)

Step	Task	Command(s)
	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, except classes 0 and 4, for which the actual power consumption is used. 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 N-Series device is an Standalone, specify 1 as the *module-number*.	**set inlinepower management** {**realtime** | **class**} *module-number*
	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**

Procedure 8-1 PoE Configuration (continued)

Step	Task	Command(s)
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 N-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 *Enterasys Matrix N-Series CLI Reference* for more information about each command.

PoE Display Commands

Table 8-3 lists PoE show commands for N-Series devices.

Table 8-3 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 *Enterasys Matrix N-Series CLI Reference* for a description of the output of each command.

9

Discovery Protocol Configuration

This chapter provides information about configuring and monitoring discovery protocols on the N-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. Enterasys 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 Enterasys 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 Enterasys 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 9-3 on page 9-6). When a new neighbor is discovered, LLDP automatically enters fast transmission state. In fast transmission state, an LLDPPDU packet is sent each fast transmission state interval for the number of intervals configured. 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.

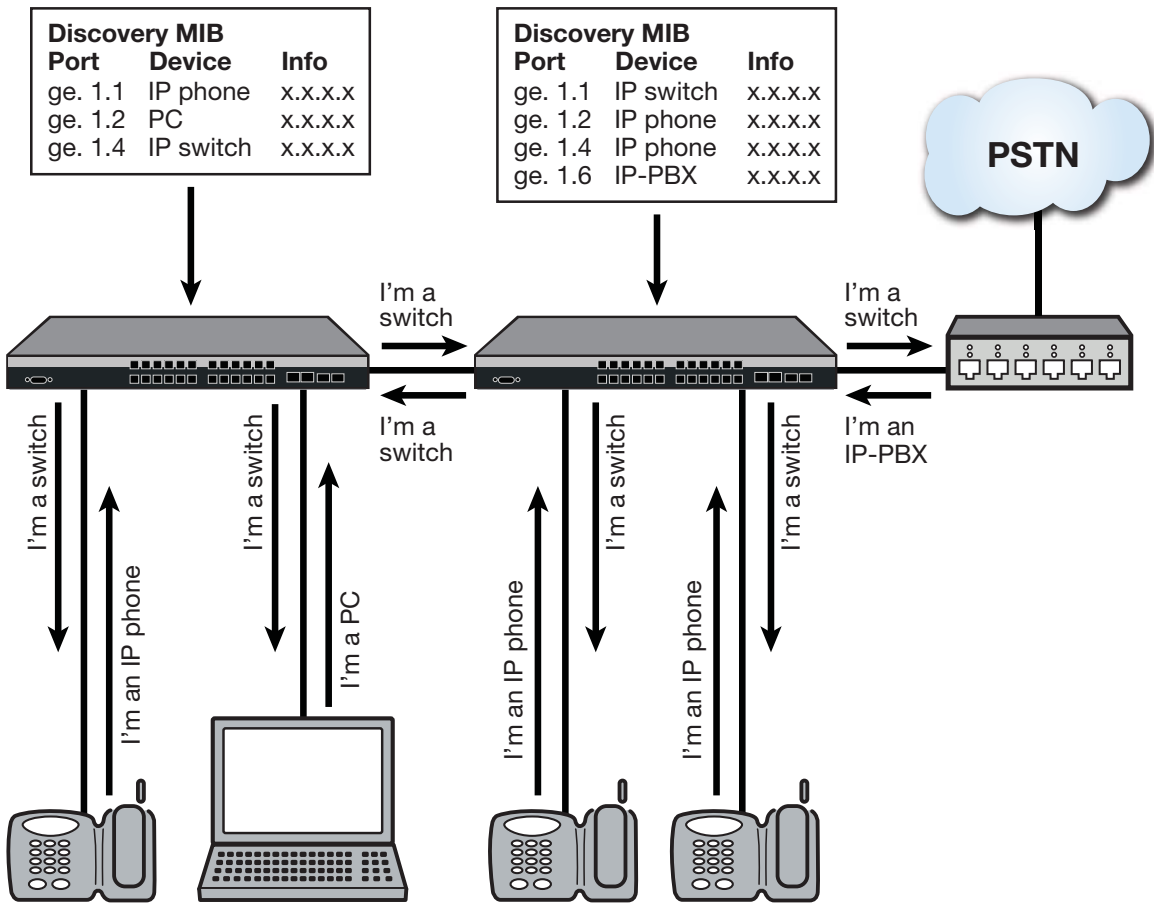
The LLDP agent operates only in an advertising mode, and hence does not support any means for soliciting information or keeping state between two LLDP entities.

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.

Figure 9-1 on page 9-3 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 9-1 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:

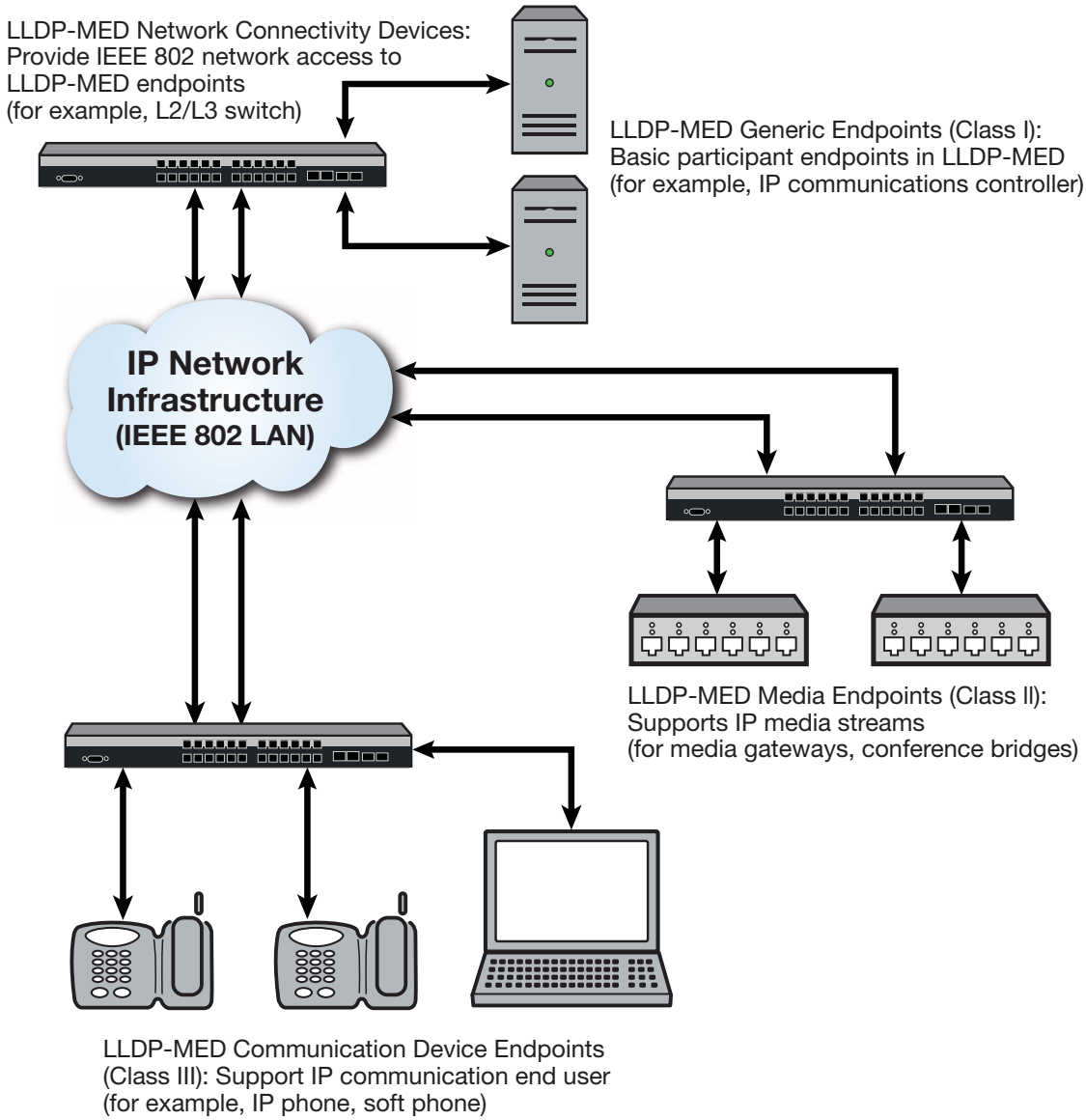
- Auto discovery of LAN policies, such as VLAN ID, 802.1p priority, and DiffServ codepoint settings, leading to plug-and-play networking.
- Device location and topology discovery, allowing creation of location databases and, in the case of VoIP, provision of E911 services.
- Extended and automated power 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 9-2 on page 9-5):

- 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 9-2 LLDP-MED

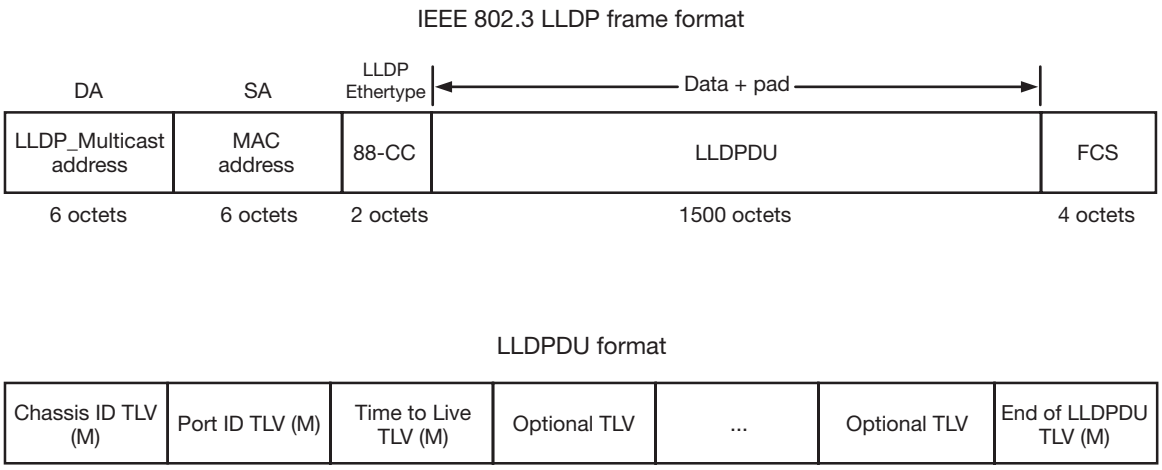


LLDPDU Frames

As shown in Figure 9-3, 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 9-3 Frame Format



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 Enterasys 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.

Configuring LLDP

LLDP Configuration Commands

Table 9-1 lists LLDP configuration commands. The table indicates which commands are device specific.

Table 9-1 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*
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*

Table 9-1 LLDP Configuration Commands (continued)

Task	Command
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**]} *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**}
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*

Table 9-1 LLDP Configuration Commands (continued)

Task	Command
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 *Enterasys Matrix N-Series CLI Reference* for more information about each command.

Basic LLDP Configuration

Procedure 9-1 describes the basic steps to configure LLDP on Enterasys N-Series devices.

Procedure 9-1 Configuring LLDP (Enterasys N-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.

```
N Chassis(rw)->set lldp tx-interval 20
N Chassis(rw)->set lldp hold-multiplier 5
```

Example LLDP Configuration: Location Information

On an N-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.

```
N Chassis(rw)->set lldp port location-info 5551234567 ge.1.1-6
N Chassis(rw)->set lldp port tx-tlv med-loc ge.1.1-6
```

LLDP Display Commands

Table 9-2 lists LLDP show commands. The table indicates which commands are device specific.

Table 9-2 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** [*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 *Enterasys Matrix N-Series CLI Reference* for a description of the output of each command.

Configuring Enterasys Discovery Protocol

Enterasys Discovery Protocol Configuration Commands

Table 9-3 lists Enterasys Discovery Protocol configuration commands.

Table 9-3 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 *Enterasys Matrix N-Series CLI Reference* for more information about each command.

Example Enterasys Discovery Protocol Configuration

This example shows how to globally enable CDP:

```
N Chassis(rw)->set cdp state enable
```

This example shows how to enable the CDP for port ge.1.2:

```
N Chassis(rw)->set cdp state enable ge.1.2
```

This example shows how to disable the CDP for port ge.1.2:

```
N Chassis(rw)->set cdp state disable ge.1.2
```

Enterasys Discovery Protocol Show Commands

Table 9-4 lists Enterasys Discovery Protocol show commands.

Table 9-4 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 *Enterasys Matrix N-Series CLI Reference* for a description of the output of each command.

Configuring Cisco Discovery Protocol

Cisco Discovery Protocol Configuration Commands

Table 9-5 lists Cisco Discovery Protocol configuration commands.

Table 9-5 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 *Enterasys Matrix N-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:

```
N Chassis(rw)->set ciscodp status enable
```

Cisco Discovery Protocol Show Commands

Table 9-6 lists Cisco Discovery Protocol show commands.

Table 9-6 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 *Enterasys Matrix N-Series CLI Reference* for a description of the output of each command.

10

Simple Network Management Protocol (SNMP) Configuration

This chapter provides information about configuring and monitoring SNMP on Enterasys N-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 Enterasys 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 Enterasys 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 10-1.

Table 10-1 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 10-6 for more information.

SNMP Support on N-Series Devices

By default, SNMP Version 1 (SNMPv1) is configured on Enterasys 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 Enterasys devices:

Versions Supported

Enterasys 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 Enterasys 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 Enterasys 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 Enterasys 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 10-2 lists common SNMP terms and defines their use on Enterasys devices.

Table 10-2 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.
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.

Table 10-2 SNMP Terms and Definitions (continued)

Term	Definition
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 Enterasys 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 10-3 identifies the levels of SNMP security available on Enterasys devices and authentication required within each model.

Table 10-3 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 Enterasys 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 Enterasys devices:

Configuration Basics

Completing an SNMP configuration on an Enterasys 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 Enterasys 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 Enterasys switches. Table 10-4 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 10-4 Default Enterasys 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 10-10.

To take advantage of the advanced security and other features available in SNMPv3, it is recommended that you add to the Enterasys default configuration by configuring SNMPv3 as described in "Configuring SNMPv3" on page 10-11.

Refer also to "Configuring Secure SNMP Community Names" on page 10-18 for a description of a recommended configuration that will prevent unsecured access to SNMP information.

Configuring SNMPv1/SNMPv2c

Creating a New Configuration

Procedure 10-1 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 10-1 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 10-1 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 N-Series device configuration using the steps in Procedure 10-1. 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**.

```
N Chassis(su)->set snmp community public
N Chassis(su)->set snmp group groupRW user public security model v1
N Chassis(su)->set snmp access groupRW security-model v1 read RW write RW notify RW
N Chassis(su)->set snmp view viewname RW subtree 1
N Chassis(su)->set snmp view viewname RW subtree 0.0
N Chassis(su)->set snmp view viewname RW subtree 1.3.6.1.6.3.13.1 excluded
N Chassis(su)->set snmp targetparams TVv1public user public security-model v1
message             processing v1
N Chassis(su)->set snmp targetaddr TVTrap 10.42.1.10 param TVv1public taglist
            TVTraptag
N Chassis(su)->set snmp notify TVTrap tag TVTrapTag
```

Adding to or Modifying the Default Configuration

By default, SNMPv1 is configured on Enterasys 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:

```
N Chassis(su)->set snmp community public
N Chassis(su)->set snmp group groupRW user public security-model v1
N Chassis(su)->set snmp access groupRW security-model v1 read All write All notify
            All
```

```
N 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:

```
N Chassis(su)->set snmp community newname
N Chassis(su)->set snmp group groupRW user newname security-model v1
```

Use this command to remove the **public** community name from the default configuration:

```
N 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 10-18 for a description of a recommended configuration that will prevent unsecured access to SNMP information.

Configuring SNMPv3

Procedure 10-2 shows how to complete a basic SNMPv3 configuration.

Procedure 10-2 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 10-2 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 N-Series device configuration using the steps in Procedure 10-2. It shows how to:

- Create the user **Enterasys_user**, specifying authentication, encryption, and security credentials.
- Assign **Enterasys_user** to the **Enterasys** 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 **Enterasys_user** using the USM security model.
- Create a target address entry named **Enterasys_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**.

```
N Chassis(su)->set snmp user Enterasys_user authentication md5 my_authentication
           privacy my_privacy
N Chassis(su)->set snmp group Enterasys user Enterasys_user security-model usm
N Chassis(su)->set snmp access Enterasys security-model usm privacy read readView
           write writeView
N Chassis(su)->set snmp view viewname readView subtree 1
N Chassis(su)-> set snmp view viewname writeView subtree 1
N Chassis(su)-> set snmp view viewname writeView subtree 1.3.6.1.4.1.5624.1.2.16
            excluded
N Chassis(su)-> set snmp targetparams matrixn user Enterasys_user security-model
usm             message-processing v3
N Chassis(su)-> set snmp targetaddr Enterasys_Networks 172.29.10.1 param matrixn
            taglist v3TrapTag
N 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 10-8, 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, (**Enterasys_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 10-3 on page 10-14 add to the following configuration example:

```
N Chassis(su)->set snmp view viewname All subtree 1
N Chassis(su)->set snmp user v3user authentication md5 md5passwd privacy despasswd
N Chassis(su)->set snmp group v3group user v3user security-model usm
N Chassis(su)->set snmp access v3group security-model usm privacy exact read All
           write All notify All
N Chassis(su)->set snmp notify v3notify tag v3tag inform
N Chassis(su)->set snmp targetaddr v3TA 134.141.209.73 param v3TP taglist v3tag
```

```
N Chassis(su)->set snmp targetparams v3TP user v3user security-model usm
message-            processing v3 privacy
```

Inform EngineIDs

In the Enterasys 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 10-3. This example assumes that NetSight Console is the receiver, and an N-Series switch is the sender.

Note: The following file location and EngineID are provided as examples. Your settings will vary.

Procedure 10-3 adds to the configuration example shown in "Configuring an SNMPv3 Inform or Trap Engine ID" on page 10-13.

Procedure 10-3 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 10-13.
2.	On the management station, navigate to and display the Netsight Console SNMP trap configuration file.	**C:\Program Files\Enterasys 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\Enterasys 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**.:

```
N Chassis(su)->clear snmp view All 1
N Chassis(su)->clear snmp view All 0.0
N Chassis(su)->set snmp view viewname All subtree 1.3.6.1.2.1
N Chassis(su)->set snmp view viewname All subtree 1.3.6.1.2.1.2 excluded
N 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 N-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.

```
N 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

N Chassis(su)->clear snmp view All 1
N Chassis(su)->set snmp view viewname All subtree 1.3.6.1.2.1.16.1.1.1.0.1012 mask
ff:df
N 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 10-4 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.

Enterasys 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 10-4 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** *read-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**
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*

Procedure 10-4 Configuring Secure Community Names (continued)

Step	Task	Command(s)
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 N-Series device configuration using the steps in Procedure 10-4.

```
N Chassis(su)->set snmp access gAdmin security-model usm privacy exact read
vSecured
            write vSecured notify vSecured
N Chassis(su)->set snmp access gReadOnlyV1V2C security-model v1   exact read
            vUnsecured
N Chassis(su)->set snmp access gReadOnlyV1V2C security-model v2c  exact read
            vUnsecured
N Chassis(su)->set snmp access gReadWriteV1V2C security-model v1  exact read
            vUnsecured write vUnsecured
N Chassis(su)->set snmp access gReadWriteV1V2C security-model v2c exact read
            vUnsecured write vUnsecured
N Chassis(su)->set snmp community cnPrivate securityname sn_v1v2c_rw
N Chassis(su)->set snmp community cnPublic securityname sn_v1v2c_ro
N Chassis(su)->set snmp group gReadOnlyV1V2C user sn_v1v2c_ro security-model v1
N Chassis(su)->set snmp group gReadWriteV1V2C user sn_v1v2c_rw security-model v1
N Chassis(su)->set snmp group gReadOnlyV1V2C user sn_v1v2c_ro security-model v2c
N Chassis(su)->set snmp group gReadWriteV1V2C user sn_v1v2c_rw security-model v2c
N Chassis(su)->set snmp group gAdmin user it-admin security-model usm
N Chassis(su)->set snmp user it-admin authentication sha auth_key privacy priv_key
N Chassis(su)->set snmp view viewname vSecured subtree 1
N Chassis(su)->set snmp view viewname vSecured subtree 0.0
N Chassis(su)->set snmp view viewname vUnsecured subtree 1
N Chassis(su)->set snmp view viewname vUnsecured subtree 0.0
N Chassis(su)->set snmp view viewname vUnsecured subtree 1.3.6.1.6.3.15 excluded
N Chassis(su)->set snmp view viewname vUnsecured subtree 1.3.6.1.6.3.16 excluded
N 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

```
N 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

```
N 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

```
N 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

```
N 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

```
N 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

```
N 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

```
N Chassis(su)-> show snmp targetparams matrix

Target Parameter Name = matrix
Security Name         = Enterasys_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

```
N Chassis(su)-> show snmp targetaddr
Target Address Name = Enterasys_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

```
N 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

```
N 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

```
N 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

```
N Chassis(su)->show snmp user Enterasys_user

EngineId                = 80:00:15:f8:03:00:e0:63:9d:cb:89
Username                = Enterasys_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

```
N Chassis(su)->show snmp view readView

View Name       = readView
Subtree OID     = 1
```

```
Subtree mask    =
View Type       = included
Storage type    = nonVolatile
Row status          = active
```

11

Spanning Tree Configuration

This chapter provides the following information about configuring and monitoring Spanning Tree protocols on Enterasys N-Series devices:

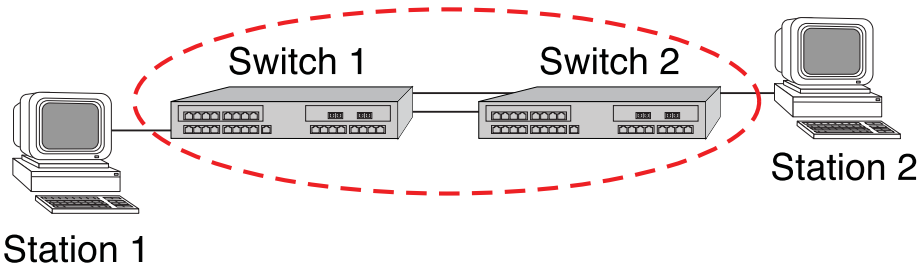
Using the Spanning Tree Protocol in Your Network

The Spanning Tree Protocol (STP) resolves the problems 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 original path fails, at which point duplicate paths can be brought into service. Should a bridge be added creating a redundant path, STP blocks one of the paths. Basically, the “tree” in the Spanning Tree Protocol is the optimal branching of network paths that STP-enabled devices keep in memory for efficient and fault-tolerant data forwarding.

Managing Redundant Links

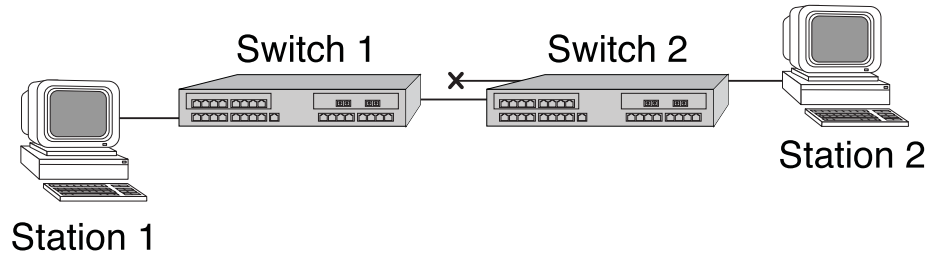
Redundant links must be factored into even the simplest of topologies to protect against data loss and downtime due to any single point of failure. However, redundant links can also set an endless loop in motion, significantly stressing your network’s speed and efficiency. As shown in Figure 11-1, a 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 memory. As administrator, you would be forced to manually disable one of the links between Switch 1 and 2 for the Figure 11-1 network to operate.

Figure 11-1 Redundant Link Causes a Loop in a Non-STP Network



With Spanning Tree running on your network devices, there would be no need for you to manually disable links. STP would automatically block one of the redundant paths, as shown in Figure 11-2, restoring a smooth data transfer between Switch 1 and 2 and end users. In the event that the primary (unblocked) path failed, STP would place the blocked path back into service and block the failed link. When enabled, it would do this automatically, without administrative intervention.

Figure 11-2 Loop Avoided When STP Blocks a Duplicate Path



Spanning Tree On N-Series Switches

By default, Spanning Tree is enabled globally on N-Series devices and enabled on all ports. The default version is set to MSTP mode, an enhancement of the standard 802.1D (see "Multiple Spanning Trees" on page 11-3). In most networks, these defaults should not be changed. However, if you are knowledgeable about Spanning Trees and configuring STP algorithms, you will be able to adjust parameters to fine tune STP performance in your network as described in this document. By using the Spanning Tree monitoring commands described here, you will also be able to better understand the default STP configuration on your Enterasys device and how it operates in your network.

STP Overview

N-Series 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 this document:

- 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)

Note: MSTP and RSTP are fully compatible and interoperable with each other and with legacy STP.

As described previously, STP resolves the problems of physical loops in a network by establishing one primary path between any two devices. It does this by enabling switching devices to exchange information using Bridge Protocol Data Unit (BPDU) messages. STP uses BPDUs to designate a bridge for each switched LAN segment, and one root bridge for the Spanning Tree. The root bridge is the logical center of the Spanning Tree and is used to determine which paths to block and which to open.

If you are familiar with STP operation and wish to adjust the defaults in your network, you can determine the topology of the Spanning Tree by adjusting the bridge priority, port priority, and path cost. The bridge priority assigns the bridge's relative priority compared to other bridges. The port priority assigns the port's priority in relation to the other ports on the same bridge. By default, the port cost is a value assigned to the 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.

Each bridge has a Bridge Identification (BID), which is derived from the bridge's MAC address and bridge priority. The bridge with the lowest BID becomes the root bridge.

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 provides rapid connectivity following the failure of a switching device, switch port, or a LAN.

Multiple Spanning Trees

Multiple Spanning Tree (MSTP) optimizes the utilization of redundant links between switching devices in a network. It assigns each VLAN present on the network to a particular Spanning Tree instance, allowing each switch port to be in a distinct state for each such instance: blocking for one Spanning Tree while forwarding for another. Thus, traffic associated with one set of VLANs can traverse a particular inter-switch link, while traffic associated with another set of VLANs can be blocked on that link. If VLANs are assigned to Spanning Trees wisely, no inter-switch link will be completely idle, maximizing network utilization.

MSTP enhances STP and RSTP with the following features:

- Backwards compatibility with STP and RSTP.
- Ability to create a single Common and Internal Spanning Tree (CIST) that represents the connectivity of the entire network.
- Users can group any number of devices into individual regions, with each region behaving and presenting itself as a single switching device to the rest of the network.
- A region can contain multiple instances of the Spanning Tree, where each instance can support multiple VLANs.

MSTP can automatically detect the version of Spanning Tree being used on a LAN and send out the equivalent type of BPDU. In addition, MSTP incorporates a force version feature that allows you to administratively force MSTP to behave as STP or RSTP.

Functions and Features Supported on N-Series Devices

Maximum STP Capacities

By default, Multiple Spanning Tree mode is globally enabled on Enterasys switching devices and one Spanning Tree is configured as Spanning Tree ID (SID) 0.

Maximum SID capacities for N-Series switches, including the default Spanning Tree (SID 0) are:

- 9 on Matrix DFE Gold Series devices
- 65 on Matrix DFE Platinum and Diamond Series devices with 256 MB of memory installed

Enterasys switching devices support a default 20-bridge span from the root bridge. You can configure support for a maximum diameter of up to 40 bridges from the Spanning Tree root as described in "Defining the Maximum Age Time" on page 11-16.

STP Features

Enterasys 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 11-20.
- Further protecting your network from loop formation with Loop Protect, as described below and in "Understanding and Configuring Loop Protect" on page 11-22.
- Supporting more port density and faster port speeds as described in Updated 802.1t on page 11-5.
- Supporting the Restricted TCN feature as described in "Restricted TCN" on page 11-5.
- Supporting the Restricted Role feature as described in "Restricted Role" on page 11-6.

SpanGuard

The Enterasys SpanGuard feature helps protect your network from two situations that can cause a Denial of Service 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 edge (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 N-Series devices and must be globally enabled to operate on all user ports. For configuration information, refer to Understanding and Configuring SpanGuard on page 11-20.

Loop Protect

The Loop Protect feature prevents or short circuits loop formation caused by redundant paths in your network 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 listening 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, and will revert to listening if it fails to get a response. This protects against misconfiguration and protocol failure by the connected bridge.

By default, the Loop Protect feature is globally disabled on N-Series devices and must be globally enabled to operate on all ports. For configuration information, refer to Understanding and Configuring Loop Protect on page 11-22.

Updated 802.1t

IEEE 802.1t is enabled by default on N-Series 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 (switch between old and new default values)

Restricted TCN

Restricted TCN is a Spanning Tree protocol feature that allows or disallows Topology Change Notification (TCN) propagation on specified ports.

When Restricted TCN is set to true, the port does not propagate received TCNs and topology changes to other ports. When set to true, temporary loss of connectivity can occur after changes in a Spanning Tree's active topology, due to persistent, incorrectly learned, station location information. You set this command to true to prevent unnecessary address flushing in the core region of the network, caused by activation of bridges external to the 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 for the attached LANs transitions frequently.

TCN propagation is set to **false** by default: the port propagates received TCNs and topology changes to other ports.

Example

The following example sets the Restricted TCN feature to true on port ge.2.1 and verifies the configuration:

```
N Chassis(rw)->set spantree restrictedtcn ge.2.1 true
N Chassis(rw)->show spantree restrictedtcn port ge.2.1
Port ge.2.1     has restrictedTcn set to True
```

Restricted Role

Restricted Role is a Spanning Tree protocol feature that allows or disallows the root role on specified ports.

When Restricted Role is set to true, the port will not to be selected as the root port for the Common Instance Spanning Tree (CIST) or any Multiple Spanning Tree Instance (MSTI), even if it has the best Spanning Tree priority. A port with Restricted Role set to true is selected as an alternate port after the root port has been selected. If set to true, Restricted Role can cause lack of Spanning Tree connectivity. Setting Restricted Role to true 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.

Restricted role is set to **false** (root role is allowed) by default.

Example

The following example sets the Restricted Role feature to true on port ge.2.1 and verifies the configuration:

```
N Chassis(rw)->set spantree restrictedrole ge.2.1 true
N Chassis(rw)->show spantree restrictedrole port ge.2.1
Port ge.2.1     has restrictedRole set to True
```

Understanding How Spanning Tree Operates

Electing the Root Bridge

A root bridge is the logical center of the Spanning Tree topology. Root is determined when bridges periodically advertise their STP information to their neighbors by transmitting BPDUs containing their root ID and bridge ID. Each receiving bridge analyzes this information, electing the bridge with the lowest bridge ID as root.

Assigning Path Costs

Spanning Tree assigns each LAN segment a path cost, which is a port speed-based value associated with each link and the relative “cost” to traverse that link.

Determining the Designated Bridge

Spanning Tree calculates a designated bridge, which is the bridge offering the lowest path cost to the root bridge. If path costs are equal, the designated bridge is the one with the lower bridge ID. Each bridge is serviced by only one designated bridge. The root bridge serves as the designated bridge for all bridges to which it is directly attached. For each bridge, Spanning Tree calculates all possible paths back to the root bridge. If the path cost is equal from multiple paths, the designated bridge will be determined by the lowest bridge ID.

Identifying Port Roles and Assigning Port States

On each bridge, Spanning Tree identifies ports which provide a path to the root bridge and selects the best path among these as the root port. Other ports with a path to the root bridge take the alternate port role. The remainder of the ports provide a path to root for attached devices. These remaining ports are designated ports, except in the condition where the port is either directly connected to another port on the bridge or connected to a shared LAN, which is also connected to another port on the bridge. In this case the port providing the best path to root is the designated port and the other port or ports have the backup role. Spanning tree uses the information in BPDUs to calculate the port roles. This information includes:

- Root bridge ID
- Root path cost
- Designated bridge ID
- Designated port ID

STP's main goal is to insure a fully connected, optimized, loop-free topology. The port role calculation makes this possible. A secondary goal, realized with the introduction of RSTP, is to move root and designated ports to the forwarding state as quickly as possible. A root port may move to forwarding as soon as any recent former root port is put into blocking. A designated port may move to forwarding once the connected device acknowledges agreement with the new topology information. This is typically an exchange of two BPDUs. For all transitions from blocking to forwarding, the port will move through the listening and learning states.

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. New designated ports will only forward after receiving an acknowledgement or, in the case of a port being connected to a 802.1d device, after a sufficient time has passed. This prevents transient loops as the network reconverges.

STP port roles are described in Table 11-1. Port states are described in Table 11-2.

Table 11-1 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).
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.

Table 11-2 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.

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 ISL in the topology to be forwarding for at least one Spanning Tree

MSTP is the default Spanning Tree mode on all N-Series 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.

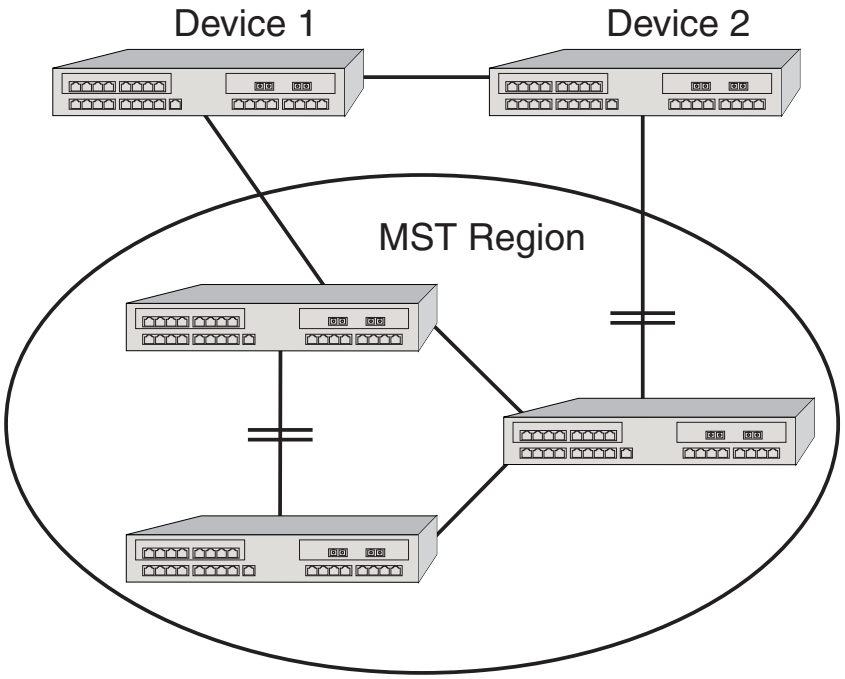
The MSTP enabled network contains one CIST and a minimum of at least one MST region. A typical network may contain numerous MST regions as well as separate LAN segments running legacy STP and RSTP Spanning Tree protocols.

The CIST contains a root bridge, which is the root of the Spanning Tree for the network. The CIST root is not necessarily located inside an MST region. Each region contains a CIST regional root, unless the CIST root is part of the region. 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.

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 11-3 shows that the MST region appears as a single switching device to Devices 1 and 2, but really consists of three devices.

Figure 11-3 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 the following four separate 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.
- 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.

The MST region designates one CIST regional root bridge for the region, regardless of the number of MSTIs. The regional root provides the connectivity from the region to the CIST root when the CIST root lies outside the region.

Multiple Spanning Tree Instances (MSTI)

Inside the MST region, a separate topology is maintained from the outside world. Each MST region may contain up to 64 different MSTIs with 256 MB of memory installed on the module. The N-Series device maps VLAN IDs (VIDs) and Filtering IDs (FIDs) to each other in a one-to-one correlation in the case of an individual VLAN learning (IVL) bridge and a one-to-many correlation in the case of an shared VLAN learning (SVL) bridge; for example, FID 3 = VID 3. VID/FIDs are mapped to different MSTIs to create a type of load balancing.

Determining FID-to-SID Mappings

VLANs are mapped to MSTIs through a FID-to-SID mapping correlation which is the key element in MSTP configuration. Each VLAN is associated to a FID and, during MSTI creation, VLANs are mapped to Spanning Tree IDs using their FID association. This mapping is contained 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 this command to determine MSTI configuration identifier information, and whether or not there is a misconfiguration due to non-matching configuration identifier components:

```
show spantree mstcfgid
```

Example

This example shows how to display MSTI configuration identifier information. In this case, this bridge belongs to "Region1":

```
N Chassis->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 only default value that must be changed for this to happen is the configuration name setting.

Use this command to change the configuration name from the default bridge MAC address value to "Region1":

```
set spantree mstcfgid cfgname Region1
```

Since an MSTI is a separate Spanning Tree, each MSTI has its own root inside the MST region. Figure 11-4 and Figure 11-5 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 power needed to maintain MSTIs, you should only create as many MSTIs as needed.

Figure 11-4 MSTI 1 in a Region

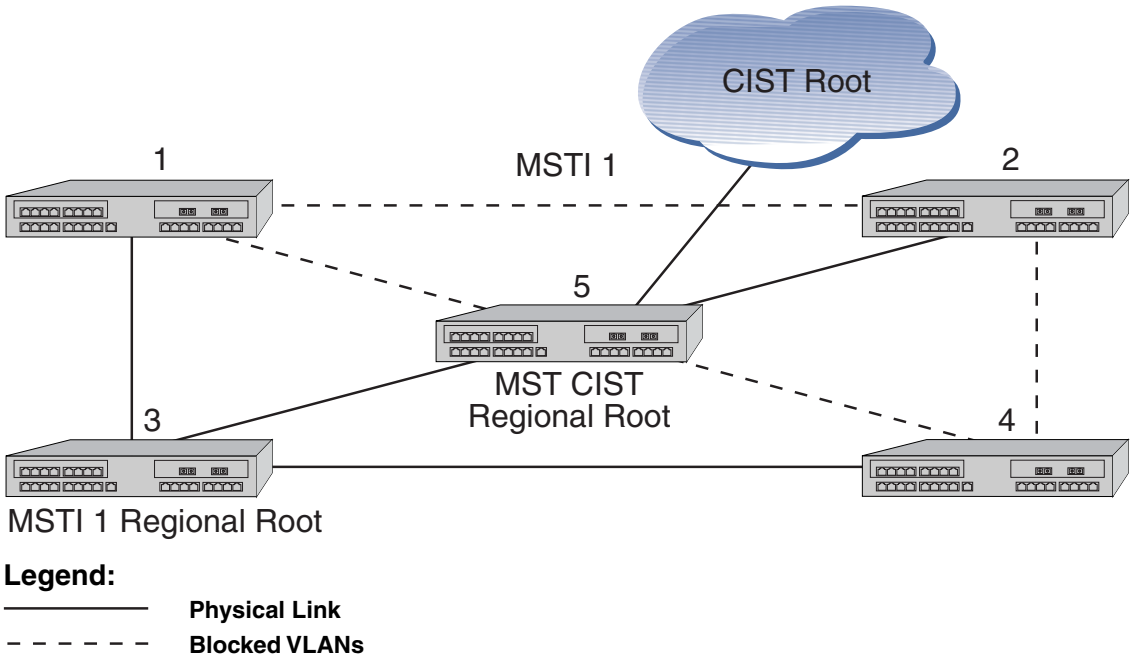


Figure 11-5 MSTI 2 in the Same Region

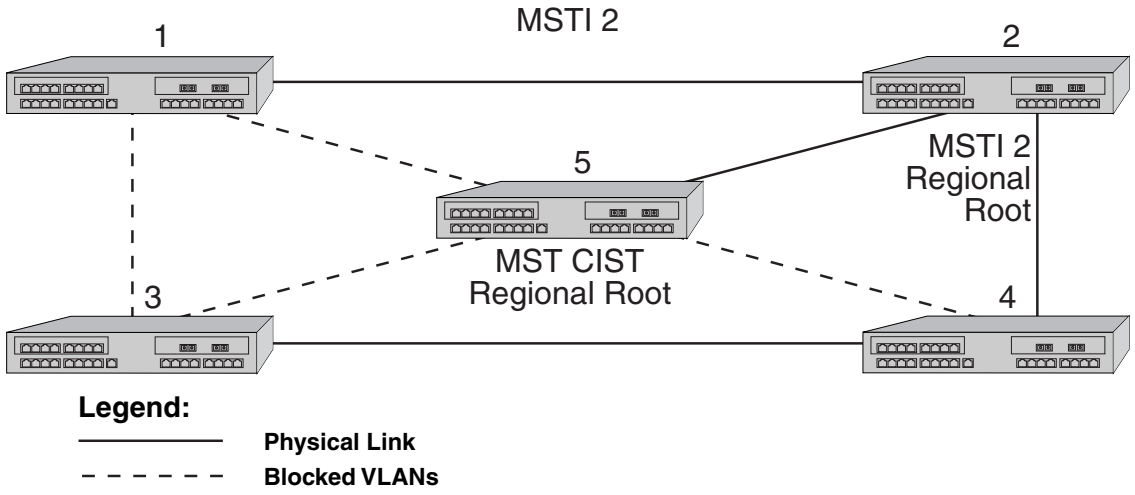


Figure 11-6 shows 3 regions with five MSTIs. Table 11-3 defines the characteristics of each MSTI. Ports connected to PCs from devices 1, 3, 9, and 11 will be automatically detected as edge ports. Devices 4 and 10 are the CIST regional roots and, because they contain the master port for their regions, are also the regional root devices. Each MSTI can be configured to forward and block various VLANs.

Figure 11-6 Example of Multiple Regions and MSTIs

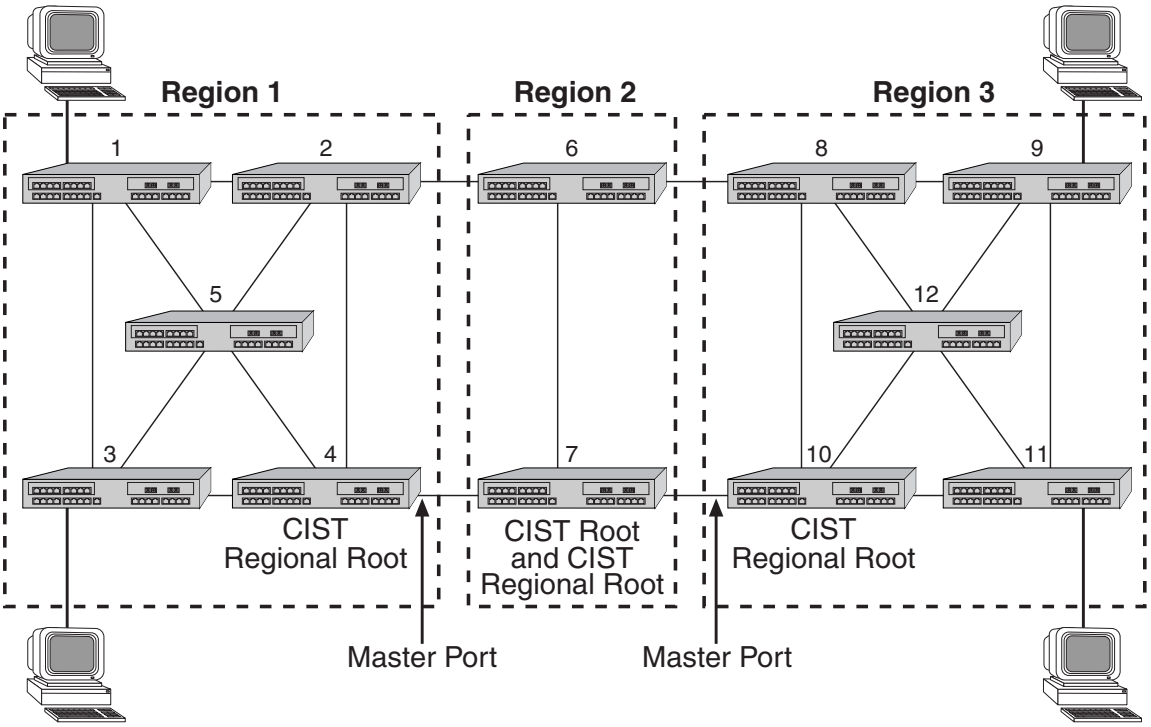


Table 11-3 MSTI Characteristics for Figure 11-6

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

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 N-Series 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.

   This example shows how to display the device's Spanning Tree configuration:

```
N Chassis(rw)->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
```

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**

Notes: By default, Spanning Tree is enabled globally on N-Series devices and enabled 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. See Table 11-4 for a listing of Spanning Tree port default settings.

Table 11-4 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
Maximum Spanning Tree instances allowed	• 5 on Matrix DFE Gold Series and Standalone devices • 33 on Matrix DFE Platinum and Diamond devices

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 11-5. 802.1t mode uses bridge priority values of 0 to 61440, in increments of 4096, with 0 indicating high priority and 61440 low priority. Legacy (802.1D) priority values are 0 to 65535.

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, a value to be 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*]

Va1id *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 11-5 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 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 1 – 10

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, Enterasys 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 Enterasys switching devices and one Spanning Tree is configured as Spanning Tree ID (SID) 0. As described in “Maximum STP Capacities” on page 11-4, devices support up to 65 Spanning Tree instances (including SID 0), depending on their model type and memory installed.

N-Series devices allow you set the maximum number of user configured STPs.

Use this command to adjust the maximum number of user configured Spanning Trees allowed on the device:

```
set spantree maxconfigurablestps numstps
```

Valid *numstps* values for N-Series devices are:

- 1-8 for DFE Gold Series devices
- 1-64 for DFE Platinum and Diamond Series devices

Enabling the Backup Root Function

Disabled by default on N-Series devices, the backup root function works only when the backup root-enabled bridge is directly connected to the root bridge. It then 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 a 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 Enterasys 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 Enterasys 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:

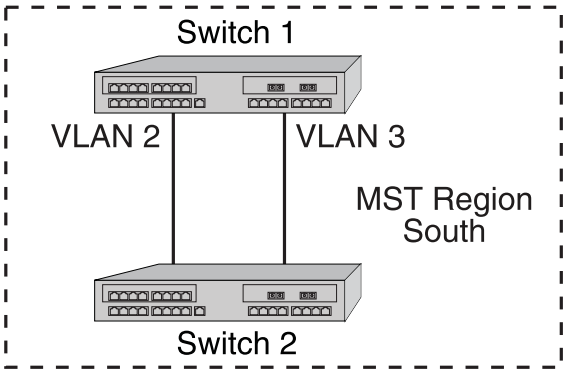
- The configuration identifier must match on all bridges within the region.
- All bridges must be within the same region.
- All bridges must be connected to MSTP-aware bridges. (They can be connected using a shared media such as a repeater provided that a single Spanning Tree device does not reside on that LAN).

Note: A single Spanning Tree device between two MSTP bridges will terminate the ability to have multiple forwarding paths. An MSTP region is bounded by a single STP device and can not contain a device in the region's interior that is not part of the region.

Example: Simple MSTP Configuration

The following example describes setting up the simple MSTP network shown in Figure 11-7. By default, each switching device will be in its own MST region using its own MAC address as the MST configuration ID. This configuration groups Switch 1 and Switch 2 into a single MST region with an MSTI configuration name of "South." It maps VLAN 2 to MSTI SID 2 and VLAN 3 to MSTI SID 3.

Figure 11-7 MSTP Sample Network Configuration



Procedure 11-1 shows how to configure Switches 1 and 2 for MSTP.

Procedure 11-1 Configuring Switches 1 and 2 for Simple MSTP

Step	Task	Command(s)
1.	Create VLANs 2 and 3.	**set vlan create 2-3**
2.	Set each switch's configuration name to South.	**set spantree mstcfgid cfgname South**
3.	Create MSTI SID 2.	**set spantree msti sid 2 create**
4.	Create MSTI SID 3.	**set spantree msti sid 3 create**

Procedure 11-1 Configuring Switches 1 and 2 for Simple MSTP (continued)

Step	Task	Command(s)
5.	Create a FID-to-SID mapping for VLAN 2 to SID 2.	**set spantree mstmap 2 sid 2**
6.	Create a FID-to-SID mapping for VLAN 3 to SID 3.	**set spantree mstmap 3 sid 3**

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 11-13 and Adjusting RSTP Parameters on page 11-17 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 11-6 to monitor MSTP statistics and configurations on N-Series devices. You can also use the show commands described in "Reviewing and Enabling Spanning Tree" on page 11-13 to review information related to all Spanning Tree protocol activity.

Table 11-6 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

For information about...	Refer to page...
What Is SpanGuard?	11-21
How Does It Operate?	11-21
Configuring SpanGuard	11-21

What Is SpanGuard?

As described previously in the overview of "SpanGuard" on page 11-4, this feature enables Enterasys 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 on N-Series devices and must be globally enabled to operate on all user ports. For configuration information, refer to "Configuring SpanGuard" on page 11-21.

How Does It Operate?

SpanGuard helps protect against Spanning Tree Denial of Service (DoS) SpanGuard attacks as well as unintentional/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 either:

- 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: In order to utilize 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 11-18, 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 11-18 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 11-7 to review SpanGuard status and settings.

Table 11-7 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 11-5, 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 is comprised of 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 helloTime. 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, then 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 a SID/port when one or more events occurs. 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 11-8, 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 11-8 Basic Loop Protect Scenario

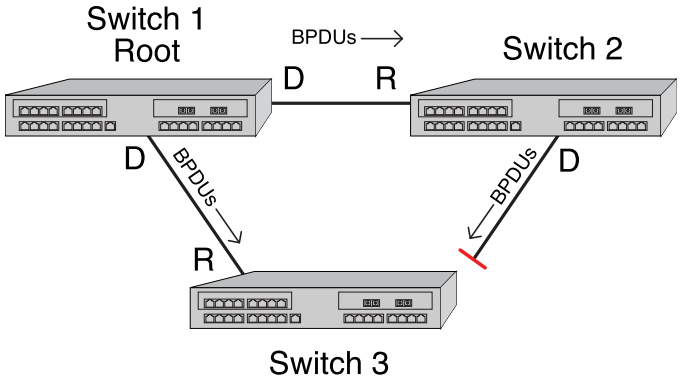


Figure 11-9 on page 11-24 shows that, without Loop Protect, a failure could be as simple as someone accidentally disabling Spanning Tree on the port between Switch 2 and 3. Switch 3's blocking port eventually transitions to a forwarding state which leads to a looped condition.

Figure 11-9 Spanning Tree Without Loop Protect

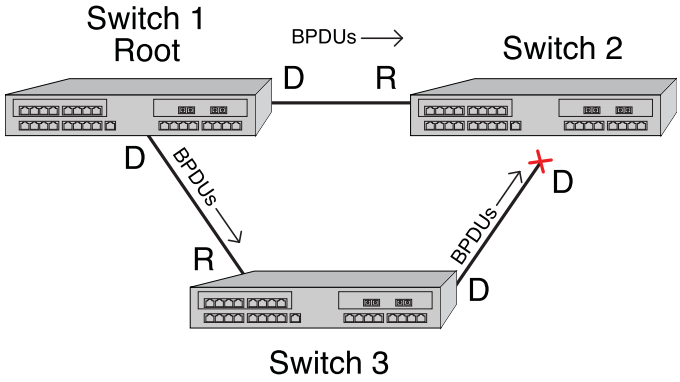
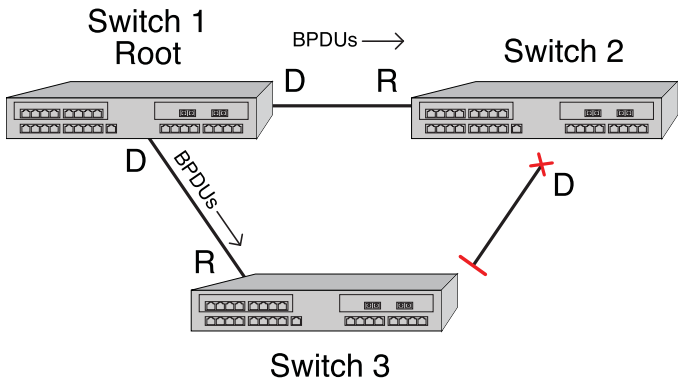


Figure 11-10 shows that, with Loop Protect enabled, Switch 3 will not go to a forwarding state until it has received a BPDU from Switch 2.

Figure 11-10 Spanning Tree with Loop Protect



Configuring Loop Protect

This section provides information about Loop Protect configuration.

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), then 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), then 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/SID until a disputed BPDU trap is sent and a syslog message is issued. For example, if the threshold is 10, then 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 11-8 to monitor Loop Protect settings.

Table 11-8 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*]

Table 11-8 Commands for Monitoring Loop Protect (continued)

Task	Command
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:

```
N Chassis->show spantree lp port lag.0.2 sid 56
LoopProtect is enabled on port lag.0.2, SID 56
N Chassis->show spantree lplock port lag.0.2 sid 56
LoopProtect Lock status for port lag.0.2, SID 56_ is UNLOCKED
N Chassis->show spantree lpcapablepartner port lag.0.2
Link partner of port lag.0.2_is LoopProtect-capable.
N Chassis->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 11-9 lists terms and definitions used in Spanning Tree configuration.

Table 11-9 Spanning Tree Terms and Definitions

Term	Definition
Alternate port	Acts as an MSTP alternate path to the root bridge than that provided by the root port.
Backup port	Acts as an MSTP 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.
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.

Table 11-9 Spanning Tree Terms and Definitions (continued)

Term	Definition
Hello time	Time interval (in seconds) at which the bridge sends BPDUs.
ISLs	Inter-Switch Links.
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. N-Series devices support up to 64 MSTIs.
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.
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.

12

VLAN Configuration

This chapter provides the following information about configuring and monitoring 802.1Q VLANs on Enterasys N-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.

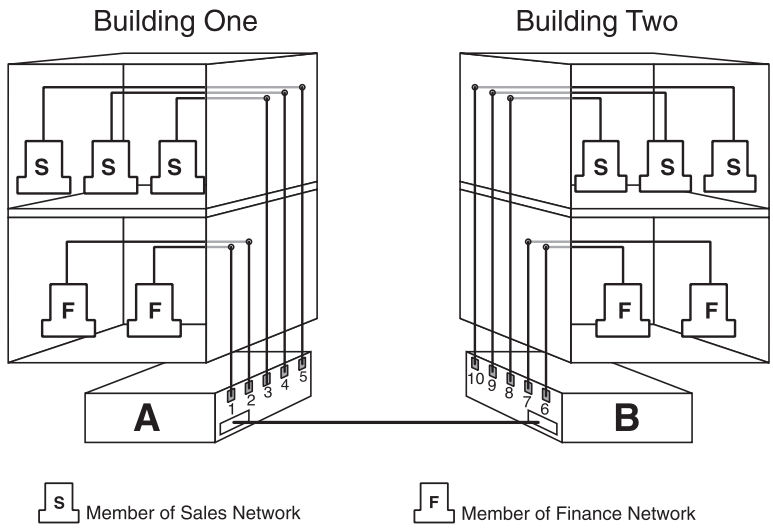
To accomplish this logical grouping, the network administrator uses 802.1Q VLAN-capable switching devices and assigns each switch port in a particular group to a VLAN. Ports in a VLAN share broadcast traffic and belong to the same broadcast domain. Broadcast traffic in one VLAN is not transmitted outside 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 12-1 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 12-1 VLAN Business Scenario



Without any VLANs configured, the entire network in the example in Figure 12-1 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 Enterasys switches run in 802.1Q VLAN operational mode. All ports on all Enterasys 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 GARP VLAN Registration Protocol (GVRP) aware ports?

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 12-9.

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).

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. This is the default mode on Enterasys 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.

VLAN Assignment and Forwarding

Receiving Frames from VLAN Ports

By default, Enterasys 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 12-5.

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 12-14.
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 12-2, 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:

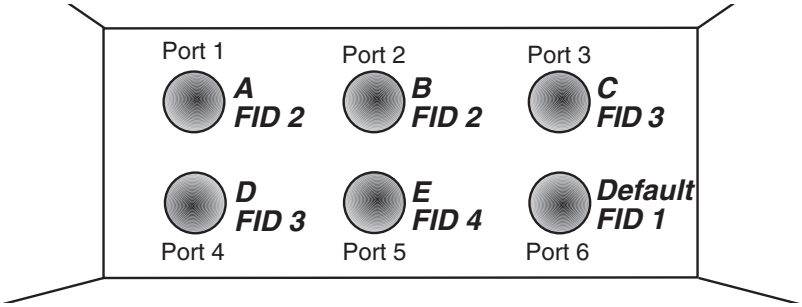
```
N Chassis(rw)->set vlan interface 1 create
N Chassis(rw)->show vlan interface 1
VLAN MIB-II Interfaces

Max Interfaces     : 16
Current Interfaces : 1

VLAN       Port          Storage Type

-------------------------------------
  1      vtap.0.1        non-volatile
N 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 12-2 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 12-2 Inside the Switch



Assume a unicast untagged frame is received on Port 3 in the example in Figure 12-2. 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 Enterasys N-Series Switches

Maximum Active VLANs

The total number of active VLANs supported on Enterasys N-Series switches is up to 4094.

Configurable Range

The allowable user-configurable range for VLAN IDs (VIDs) on Enterasys N-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 Enterasys switch assumes you intend to modify the existing VLAN.

VLAN Types

Enterasys switches support traffic classification for the following VLAN types:

Static and Dynamic VLANs

All VLANs on an Enterasys 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 and its effect on egress membership as described in "GARP VLAN Registration Protocol (GVRP) Support" on page 12-8.

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.

GARP VLAN Registration Protocol (GVRP) Support

The purpose of the GARP (Generic Attribute Registration Protocol) VLAN Registration Protocol (GVRP) is to dynamically create VLANs across a switched network. GVRP allows GVRP-aware devices to dynamically establish and update their knowledge of the set of VLANs that currently have active members.

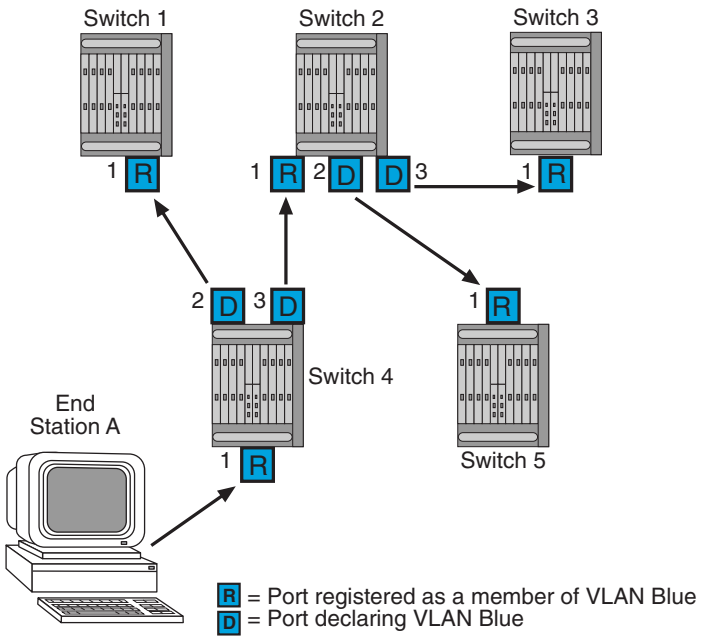
By default, GVRP is enabled both globally and at the port level. To allow GVRP to dynamically create VLANs, it must be enabled globally as well as on each individual port as described in "Configuring Dynamic VLANs" on page 12-13.

How It Works

When a VLAN has egress, the information is transmitted out GVRP configured ports on the device in a GARP formatted frame using the GVRP multicast MAC address. A switch that receives this frame examines the frame and extracts the VLAN IDs. GVRP 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 configured ports of the device.

Figure 12-3 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 12-3 Example of VLAN Propagation Using GVRP



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-created dynamic 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 GVRP 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 12-3, you can begin configuring VLANs as described in this section.

Default Settings

Table 12-1 lists VLAN parameters and their default values.

Table 12-1 Default VLAN Parameters

Parameter	Description	Default Value
garp timer	Configures the three GARP timers. The setting is critical and should only be done 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. GVRP must be enabled to allow creation of dynamic VLANs.	• Enabled at the port level • Enabled at the global level
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

Table 12-1 Default VLAN Parameters (continued)

Parameter	Description	Default Value
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.	Enabled
port vlan ID (PVID)	802.1Q VLAN/port association.	VLAN1/ Default VLAN
vlan constraint (Not supported on the DFE Gold module and Standalone device)	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 12-1 describes how to create and configure a static VLAN. Unspecified parameters use their default values.

Procedure 12-1 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*

Procedure 12-1 Static VLAN Configuration (continued)

Step	Task	Command(s)
	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.	
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. (Not supported on the DFE Gold module and Standalone device). 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**}

Procedure 12-1 Static VLAN Configuration (continued)

Step	Task	Command(s)
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**

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 N-Series device configuration using the steps in Procedure 12-1. 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.

```
N Chassis(rw)->set vlan create 100
N Chassis(rw)->set vlan name 100 VLANRED
N 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.
N Chassis(rw)->router
N Chassis(rw)-router)->configure terminal
N Chassis(rw)-router-config)->interface vlan 100
N Chassis(rw)-router-config-intf-Vlan-100)->ip address 120.20.20.1/24
N 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:

```
N 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:

```
N Chassis(rw)->set port discard ge.1.2-4 tagged
```

Creating a Secure Management VLAN

If you are configuring an Enterasys 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 12-2 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 12-2 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 12-3 describes how to enable the GARP (Generic Attribute Registration Protocol) VLAN Registration Protocol (GVRP), which is needed to create dynamic VLANs. By default, GVRP is enabled globally and at the port level. GVRP must be globally enabled and also enabled on specific ports in order to generate and process GVRP advertisement frames.

Note: Refer to "GARP VLAN Registration Protocol (GVRP) Support" on page 12-8 for conceptual information about GVRP.

Procedure 12-3 Dynamic VLAN 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.	If necessary, enable GVRP on those ports assigned to a VLAN. GVRP is enabled at the port level by default.	**set gvrp enable** *port-string*
3.	Display the existing GARP timer values.	**show garp timer** [*port-string*]
4.	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*

Caution: The setting of GARP timers is critical and should only be changed by personnel familiar with 802.1Q standards.

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 12-4 describes how to define protocol-based frame filtering policies to assign frames to particular VLANs. Refer to your Enterasys policy configuration and CLI documentation for more information.

Note: Depending on your Enterasys 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 12-4 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**]
3.	Disable ingress filtering on the ingress ports on which the policy will be applied.	**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 N-Series device configuration using the steps in Procedure 12-4. 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).

```
N Chassis(rw)->set vlan create 2, 3
N Chassis(rw)->set vlan egress 2 ge.4.1-2
N Chassis(rw)->set vlan egress 3 ge.4.3-5
N Chassis(rw)->set port ingress-filter ge.5.8-10 disable
N Chassis(rw)->set policy profile 1 name protocol_based_vlan pvid-status enable
   pvid 3
N Chassis(rw)->set policy rule admin-profile port ge.5.8 port-string ge.5.8
   admin-pid 1
N Chassis(rw)->set policy rule admin-profile port ge.5.9 port-string ge.5.9
   admin-pid 1
N Chassis(rw)->set policy rule admin-profile port ge.5.10 port-string ge.5.10
   admin-pid 1
N Chassis(rw)->set policy rule 1 ether 0x0800 mask 16 vlan 2
N Chassis(rw)->set policy rule 1 ether 0x0806 mask 16 vlan 2
N Chassis(rw)->set policy rule 1 ether 0x8035 mask 16 vlan 2
```

Configuring IGMP VLAN Snooping

IGMP Layer 2 snooping allows the Enterasys 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 *Enterasys Matrix N-Series CLI Reference*.

Procedure 12-5 describes how to configure IGMP snooping for a VLAN.

Procedure 12-5 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, or **2** for IGMPV2.	**set igmp config** *vlan-id* **igmp-version 1**|**2**
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*

Procedure 12-5 IGMP Snooping for a VLAN Configuration (continued)

Step	Task	Command(s)
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 12-2 describes the **show** commands that display information about VLAN configurations. Refer to *Enterasys Matrix N-Series CLI Reference* for a description of the output of each **show** command.

Table 12-2 Displaying VLAN Information

Task	Command
Display all existing VLANs.	**show vlan**
Display the VLAN constraint setting. (Not supported on the DFE Gold module and Standalone device).	**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 GARP timer vlaues for one or more ports.	**show garp 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 12-3 lists terms and definitions used in VLAN configuration.

Table 12-3 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.

Table 12-3 VLAN Terms and Definitions (continued)

Term	Definition
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.
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.
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.

13

Link Aggregation Control Protocol (LACP) Configuration

This document describes the link aggregation feature and its configuration on Enterasys N-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 13-3).

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
- 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:

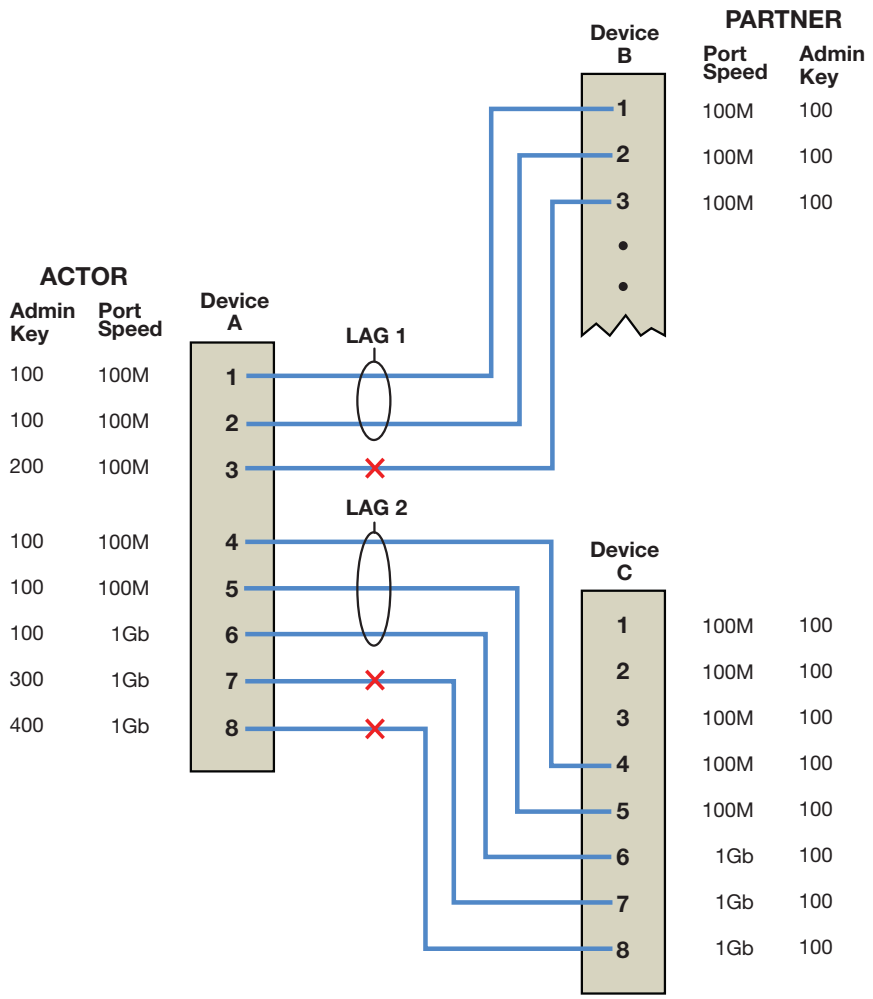
1. Operate in full duplex mode.
2. Have matching local LAG and physical port admin keys for the device controlling LAG formation.
3. Operate in parallel in that a LAG can have only two devices associated with it.
4. Consist of two or more physical actor to partner port pairings unless the single port LAG feature is enabled.

Figure 13-1 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 13-1 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 13-1 on page 13-4 as an example, if the LAG2 member port priorities are set as shown in Table 13-1 on page 13-5, ports 4 and 5 are moved to the attached state.

Table 13-1 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.

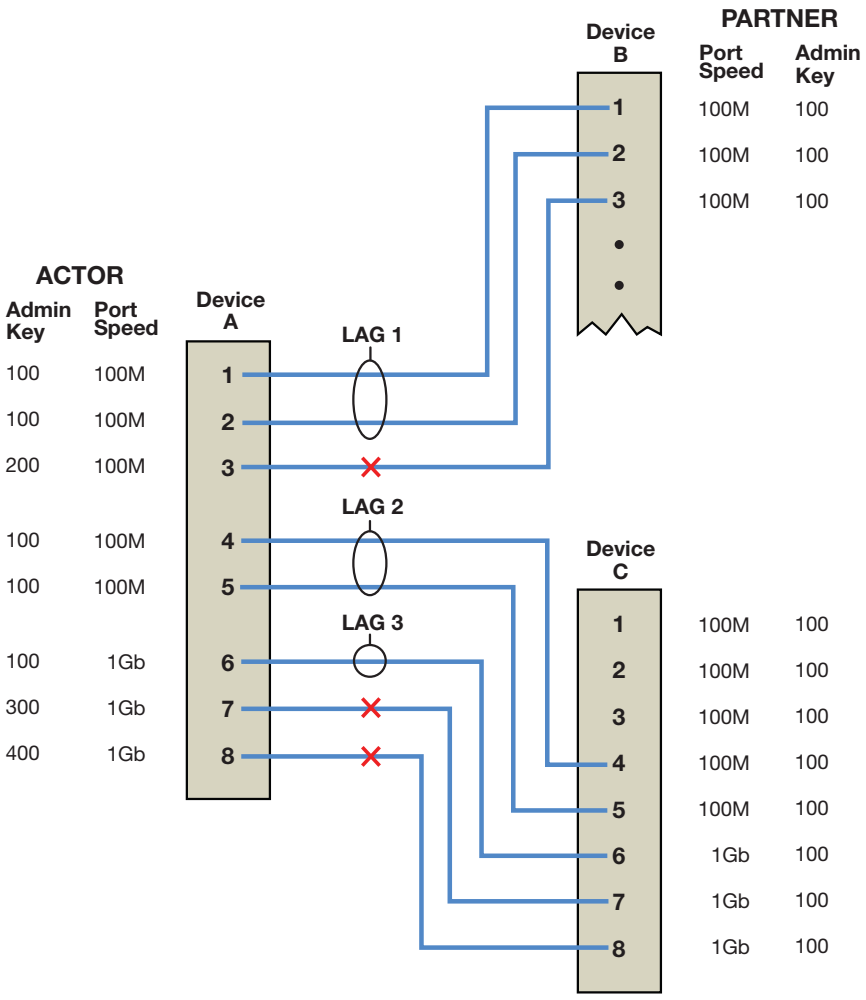
If LAG members with different port speeds should tie for the lowest port priority, the LAG member with the lowest port number breaks the tie. In our example, should all three ports have

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 13-1 on page 13-4, 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 13-2 illustrates the three LAGs described in this example.

Figure 13-2 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 13-2 specifies the LACP port parameters that can be changed.

Table 13-2 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 13-2 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 Enterasys 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 13-3 for a listing of the number of LAGs and the number of ports per LAG supported for your platform.

Table 13-3 Enterasys Platform LAG Support

Enterasys Platform	Number of LAGs Supported	Number of Ports in a LAG
N-Series DFE Platinum and Diamond modules	62	No Limitation
N-Series DFE Gold modules	4	4
N Standalone (NSA)	48	No Limitation

Configuring Link Aggregation

This section provides details for the configuration of link aggregation on the N-Series products.

Table 13-4 lists link aggregation parameters and their default values.

Table 13-4 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.	Enabled
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 13-1 describes how to configure link aggregation.

Procedure 13-1 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, modify the LAG port parameters. See Table 13-2 on page 13-7 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 13-5 describes how to manage link aggregation.

Table 13-5 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 13-5 Managing Link Aggregation (continued)

Task	Command

Reset a link aggregation port setting to the default value for one or more ports. See Table 13-2 on page 13-7 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 13-6 describes how to display link aggregation information and statistics.

Table 13-6 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:

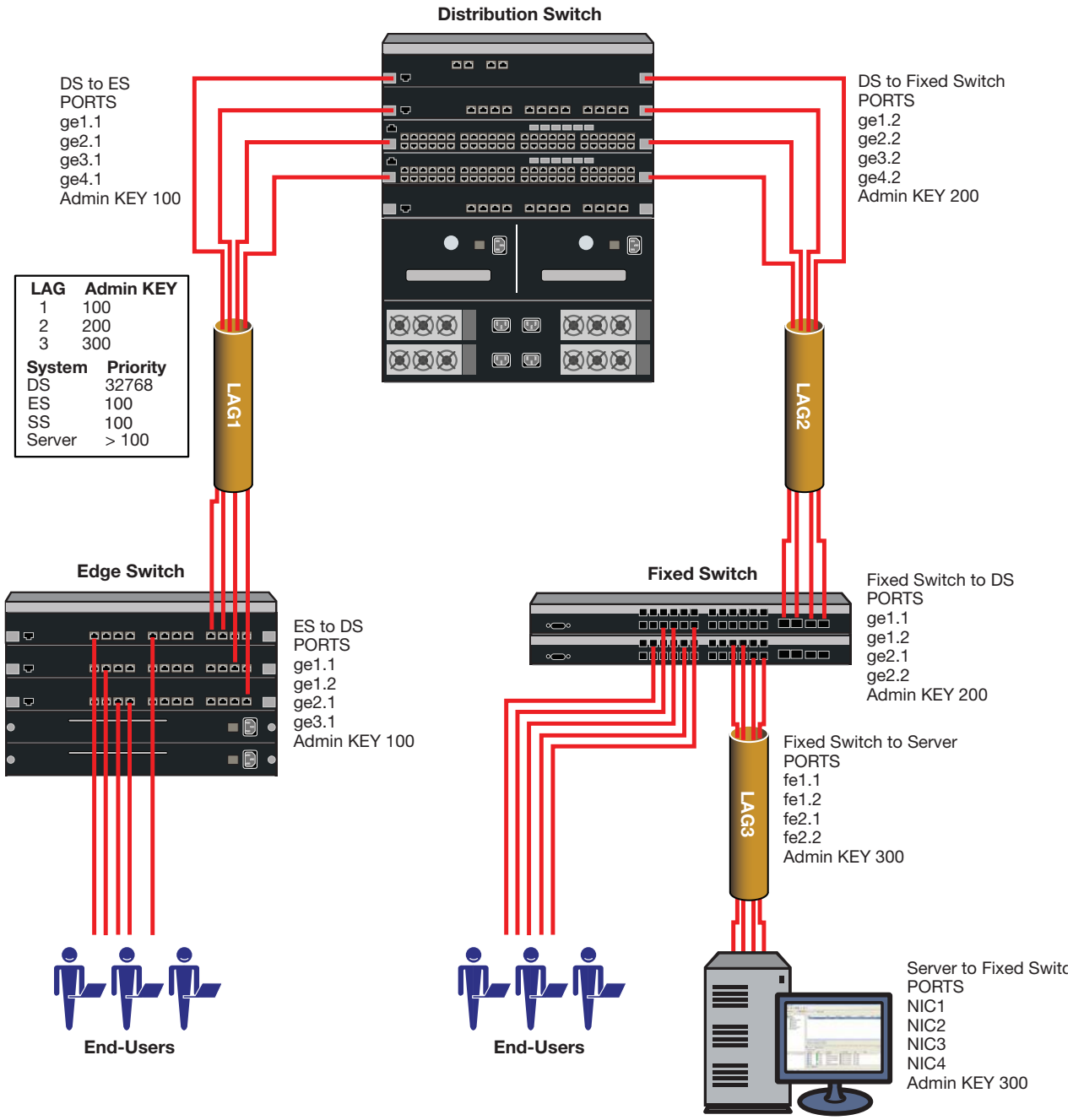
- An example of link aggregations between multiple devices
- An example of link aggregation when a LAG contains physical ports with different speeds

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 13-3 on page 12 for an illustration of this example, including port, key, and system priority assignments.

Figure 13-3 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 13-7 to ensure that the LAGs form only for the desired ports.

Table 13-7 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 13-7 on page 13-13.

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. The Fixed Switch does 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:

```
N 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:

```
N Chassis(rw)->set lacp aadminkey lag.0.1 100
N Chassis(rw)->set lacp aadminkey lag.0.2 200
```

We next want to set the admin keys for the distribution switch physical ports:

```
N Chassis(rw)->set port lacp port ge.1.1 aadminkey 100
N Chassis(rw)->set port lacp port ge.2.1 aadminkey 100
N Chassis(rw)->set port lacp port ge.3.1 aadminkey 100
N Chassis(rw)->set port lacp port ge.4.1 aadminkey 100
N Chassis(rw)->set port lacp port ge.1.2 aadminkey 200
N Chassis(rw)->set port lacp port ge.2.2 aadminkey 200
N Chassis(rw)->set port lacp port ge.3.2 aadminkey 200
N Chassis(rw)->set port lacp port ge.4.2 aadminkey 200
```

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:

```
N 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:

```
N 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:

```
N Chassis(rw)->set lacp aadminkey lag.0.1 100
```

We next want to set the admin keys for the edge switch physical ports:

```
N Chassis(rw)->set port lacp port ge.1.1 aadminkey 100
N Chassis(rw)->set port lacp port ge.1.2 aadminkey 100
N Chassis(rw)->set port lacp port ge.2.1 aadminkey 100
N Chassis(rw)->set port lacp port ge.3.1 aadminkey 100
```

Next we want to change the system priority for the edge switch so that it will be in charge of port selection on LAG1:

```
N Chassis(rw)->set lacp asyspri 100
```

We next enable flow regeneration on the edge switch:

```
N 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 set the admin keys for the Fixed Switch physical ports:

```
FixedSwitch(rw)->set port lacp port ge.1.1 aadminkey 200
FixedSwitch(rw)->set port lacp port ge.1.2 aadminkey 200
FixedSwitch(rw)->set port lacp port ge.2.1 aadminkey 200
FixedSwitch(rw)->set port lacp port ge.2.2 aadminkey 200
FixedSwitch(rw)->set port lacp port ge.1.1 aadminkey 300
FixedSwitch(rw)->set port lacp port ge.1.2 aadminkey 300
FixedSwitch(rw)->set port lacp port ge.2.1 aadminkey 300
FixedSwitch(rw)->set port lacp port ge.2.2 aadminkey 300
```

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 13-3 on page 13-9 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 13-4 on page 13-17 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.

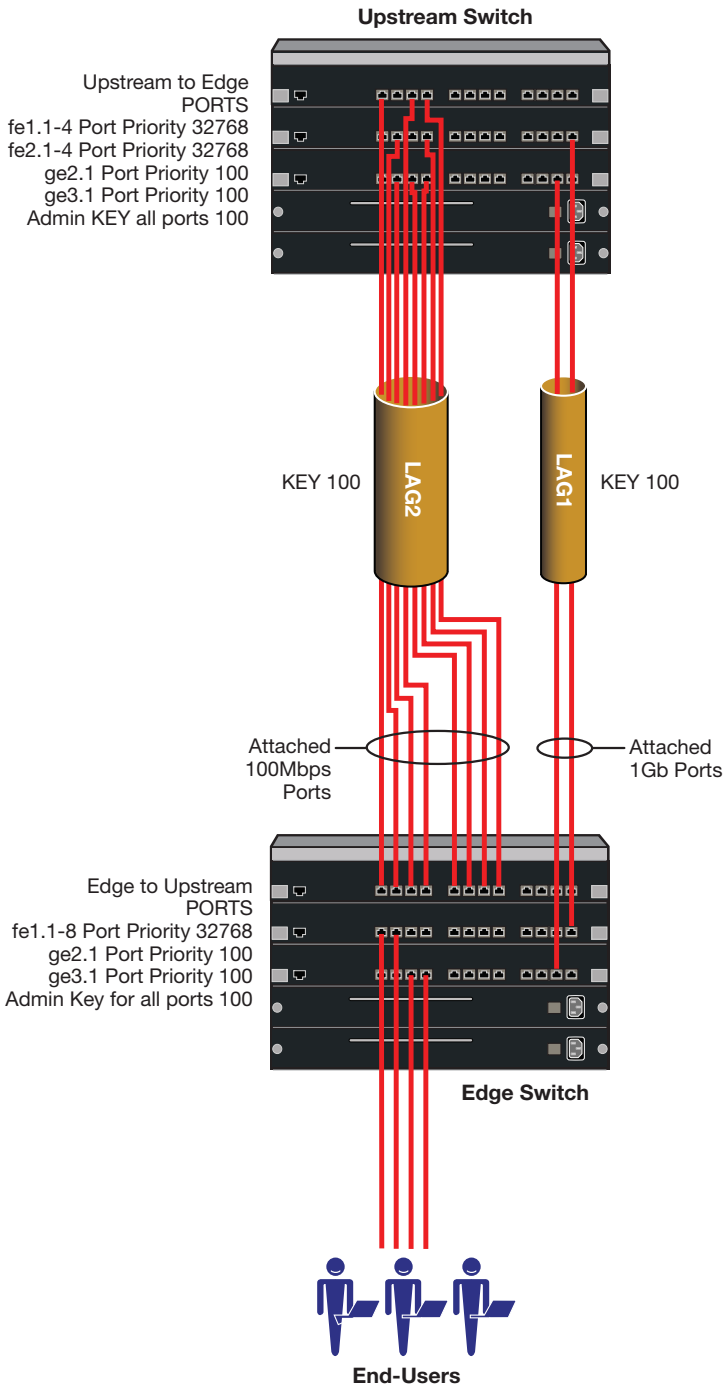
The admin key for each physical port and LAG in the example is set to 100. This ensures that LAGs will form for each set of ports.

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 13-4 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 the admin key for these LAGs to 100:

```
N Chassis(rw)->set lacp aadminkey lag.0.1-2 100
```

We next want to set the admin keys for the edge switch physical ports associated with LAG1 and LAG2:

```
N Chassis(rw)->set port lacp port ge.2.1 aadminkey 100
N Chassis(rw)->set port lacp port ge.3.1 aadminkey 100
N Chassis(rw)->set port lacp port ge.1.1 aadminkey 100
N Chassis(rw)->set port lacp port ge.1.2 aadminkey 100
N Chassis(rw)->set port lacp port ge.1.3 aadminkey 100
N Chassis(rw)->set port lacp port ge.1.4 aadminkey 100
N Chassis(rw)->set port lacp port ge.1.5 aadminkey 100
N Chassis(rw)->set port lacp port ge.1.6 aadminkey 100
N Chassis(rw)->set port lacp port ge.1.7 aadminkey 100
N Chassis(rw)->set port lacp port ge.1.8 aadminkey 100
```

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:

```
N Chassis(rw)->set port lacp port ge.2.1 aportpri 100
N Chassis(rw)->set port lacp port ge.3.1 aportpri 100
```

We next enable single port LAGs on this device:

```
N Chassis(rw)->set lacp singleportlag enable
```

We next enable flow regeneration on the edge switch:

```
N 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 for these LAGs to 100:

```
N Chassis(rw)->set lacp aadminkey lag.0.1-2 100
```

We next want to set the admin keys for the upstream switch physical ports associated with LAG1:

```
N Chassis(rw)->set port lacp port ge.2.1 aadminkey 100
N Chassis(rw)->set port lacp port ge.3.1 aadminkey 100
N Chassis(rw)->set port lacp port ge.1.1 aadminkey 100
N Chassis(rw)->set port lacp port ge.1.2 aadminkey 100
N Chassis(rw)->set port lacp port ge.1.3 aadminkey 100
N Chassis(rw)->set port lacp port ge.1.4 aadminkey 100
N Chassis(rw)->set port lacp port ge.2.1 aadminkey 100
N Chassis(rw)->set port lacp port ge.2.2 aadminkey 100
N Chassis(rw)->set port lacp port ge.2.3 aadminkey 100
N Chassis(rw)->set port lacp port ge.2.4 aadminkey 100
```

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:

```
N Chassis(rw)->set port lacp port ge.2.1 aportpri 100
N Chassis(rw)->set port lacp port ge.3.1 aportpri 100
```

We next enable single port LAGs on this device:

```
N Chassis(rw)->set lacp singleportlag enable
```

We next enable flow regeneration on the upstream switch:

```
N Chassis(rw)->set lacp flowRegeneration enable
```

This completes the example 2 configuration.

Terms and Definitions

Table 13-8 lists terms and definitions used in this link aggregation configuration discussion.

Table 13-8 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 13-3 on page 13-9 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.

14

Policy Configuration

This document describes the Enterasys policy feature and its configuration on Enterasys N-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 Enterasys Secure Networks policy in your network are provisioning and control of network resources, security, and centralized operational efficiency using the Enterasys 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 23 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:

- 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

Enterasys NetSight Policy Manager provides a centralized point and click configuration, and one click pushing of defined policy out to all network elements. Use the Enterasys 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

Policy Overview

Introduction

This section provides an overview of policy configuration. Policy is implemented on an Enterasys 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 Enterasys NetSight Policy Manager

Enterasys 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 Enterasys 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.enterasys.com/products/visibility-control/netsight-policy-manager.aspx.

Understanding Roles in a Secure Network

The capacity to define roles is directly derived from the ability of the Enterasys N-Series, SecureStack, 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 14-5), 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:

```
N 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: Enterasys 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 Enterasys N-Series Switches" on page 12-6 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 profile with a profile-index value of **2**.

```
N Chassis(rw)->set policy profile 1 pvid-status enable 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 31*, **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**:

```
N Chassis(rw)->set policy profile 1 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.

Applying a Destination Mirror to a Role

Destination mirrors can be created for one or more IP addresses or VLANs. See the Chapter 5, **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.

The following example configures profile **3** to add any port assigned to this profile to the VLAN **100** egress list formatted as untagged:

```
N Chassis(rw)->set policy profile 3 untagged-vlans 100
```

The following example changes profile **3** to add any port assigned to this profile to the VLAN **100** egress list formatted as tagged. Any other existing profile 3 configuration remains unchanged:

```
N Chassis(rw)->set policy profile 3 egress-vlans 100 append
```

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 must be enabled for both the policy role and the port the role is applied to.

TCI overwrite is not supported by the DFE Gold module.

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 Enterasys 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.

VLAN-to-Policy mapping is not supported by the DFE Gold module.

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**:

```
N 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 N-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 33-11 for a detailed RADIUS response discussion.

Please see *"Configuring RADIUS" on page 33-24* 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.

Note: Dynamic policy assignment is not supported by the DFE Gold module.

The following example specifies that the RADIUS filter-ID, if it is present, should be included in the RADIUS response message when a user authenticates:

```
N 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** 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 33-23 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 in the RADIUS response message can be included in the RADIUS response message:

```
N 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:

```
N 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 14-11.

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. Use the **clear policy disabled-ports** command to clear the disabled port.

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 14-1 on page 14-8 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 14-1 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. Not supported by the DFE Gold module.	**3**
ipxdest	Classifies based on destination IPX address. Not supported by the DFE Gold module.	**4**
ipxsourcesocket	Classifies based on source IPX socket. Not supported by the DFE Gold module.	**5**
ipxdestsocket	Classifies based on destination IPX socket. Not supported by the DFE Gold module.	**6**
ipxclass	Classifies based on transmission control in IPX. Not supported by the DFE Gold module.	**7**
ipxtype	Classifies based on IPX packet type. Not supported by the DFE Gold module.	**8**
ipsourcesocket	Classifies based on source IP address with optional post-fixed port.	**12**
ipdestsocket	Classifies based on destination IP address with optional post-fixed port.	**13**
ip frag	Classifies based on IP fragmentation value.	**14**
udpsourceportip	Classifies based on UDP source port and optional post-fix IP address.	**15**

Table 14-1 Administrative Policy and Policy Rule Traffic Classifications (continued)

Traffic Classification	Description	Attribute ID
udpdestportip	Classifies based on UDP destination port and optional post-fix IP address.	**16**
tcpsourceportip	Classifies based on TCP source port and optional post-fix IP address.	**17**
tcpdestportip	Classifies based on TCP destination port and optional post-fix IP address.	**18**
icmptype	Classifies based on ICMP type. Not supported by the DFE Gold module.	**19**
iptos	Classifies based on Type of Service field in IP packet.	**21**
ipproto	Classifies based on protocol field in IP packet.	**22**
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. Not supported by the DFE Gold module.	**27**
tci	Classifies based on Tag Control Information. Not supported by the DFE Gold module.	**28**
port	Classifies based on port-string.	**31**

Note: The optional post-fixed port traffic classification listed in Table 14-1 for IP, UDP, and TCP source and destination port traffic classifications is supported on DFE blades only.

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**:

```
N Chassis(rw)->set policy profile 1 tci-overwrite enable
N 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 lower 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 14-1 on page 14-8 for a listing of classification rule attributes and their associated attribute ID values.

Traffic classification precedence is not supported by the DFE Gold module.

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 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 N-Series example sets the port (**31**) attribute to the highest precedence and leaves the remaining attributes in the default ordering:

```
N Chassis(rw)->set policy profile 200 precedence 31,1-8,12-19,21-22,25-28
N Chassis(rw)->show policy profile 200
Profile Index           :200
Profile Name            :
.
.
.
Rule Precedence         :31,1-8,12-19,21-22,25-28
                        :Port (31), MACSource (1), MACDest (2), 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)
.
.
.
N Chassis(rw)->
```

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 14-1 on page 14-8 for a listing of supported traffic classification rule-types. Use the attribute ID value, specified in Table 14-1, 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:

```
N 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 is not supported by the DFE Gold module.

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.

Policy syslog rule usage is not supported on the DFE Gold module.

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 14-4. 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 31, **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:

```
N 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 14-7.

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 due to first profile rule use.

To clear the disabled port use the **clear policy disabled-ports** command.

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 14-24 for an example of how to configure a subset of these recommended IP services to drop traffic at the edge.

Table 14-2 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.
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 14-3 provides a listing of policy capabilities.

Table 14-3 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). Not supported by the DFE Gold module.
TCI Overwrite	The ability to overwrite user priority and other VLAN tag TCI field classification information. Not supported by the DFE Gold module.
Rule-Use Accounting	The ability to enable policy accounting.
Rule-Use Notification	The ability to enable syslog and traps for rule hit notification. Not supported by the DFE Gold module.
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. Not supported by the DFE Gold module.
Precedence Reordering	The ability to reorder traffic classification precedence for a policy role. Not supported by the DFE Gold module.

Configuring Policy

This section presents configuration procedures and tables including command description and syntax in the following policy areas: profile, classification, and display.

Procedure 14-1 describes how to configure policy roles and related functionality.

Procedure 14-1 Configuring Policy Roles

Step	Task	Command(s)
1.	In switch 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. • **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.	**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**}]

Procedure 14-1 Configuring Policy Roles (continued)

Step	Task	Command(s)
	• **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. • **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.	
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**}]

Procedure 14-1 Configuring Policy Roles (continued)

Step	Task	Command(s)
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 14-2 describes how to configure classification rules as an administrative profile or to assign policy rules to a policy role.

Procedure 14-2 Configuring Classification Rules

Step	Task	Command(s)
1.	In switch command mode, optionally set an administrative profile to assign traffic classifications to a policy role. See Table 14-1 on page 14-8 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 administratively-assigned rule to a specific ingress port. N-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.	**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 14-2 Configuring Classification Rules (continued)

Step	Task	Command(s)
	• **clear-mirror** - (Optional) Clears mirroring on this profile. • **prohibit-mirror** - (Optional) Prohibits mirroring on this profile.	
2.	In switch command mode, optionally configure policy rules to associate with a policy role. See Table 14-1 on page 14-8 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. N-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**]
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**]

Procedure 14-2 Configuring Classification Rules (continued)

Step	Task	Command(s)
5.	(Optional) Enable or disable the the ability to clear rule usage information if operational status "up" is detected on any port. The autoclear rule usage feature is not supported by the DFE Gold module.	**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. Dynamic policy assignment is not supported by the DFE Gold module.	**set policy dynamic** [**syslog-default** {**enable** | **disable**}] [**trap-default** {**enable** | **disable**}]}

Table 14-4 describes how to display policy information and statistics.

Table 14-4 Displaying Policy Configuration and Statistics

Task	Command(s)
In switch command mode, display policy role information.	**show policy profile** {**all** | *profile-index* [*consecutive-pids*] [**-verbose**]}
In switch 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 switch command mode, display the current control status of the collection of rule usage statistics.	**show policy accounting**
In switch command mode, display syslog parameters for policy rule entries.	**show policy syslog** [**machine-readable**] [**extended-format**]
In switch command mode, display VLAN-ID to policy role mappings table.	**show policy maptable** *vlan-list*
In switch command mode, display TCI overwrite tag control information on one or more ports.	**show port tcioverwrite** [*port-string*]
In switch command mode, display policy classification and admin rule information.	**show policy rule** [**attribute**] | [**all**] | [**admin-profile**] | [*profile-index*] [**port-hit**] *classification-type* [*data*] [**mask** *mask*] [**port-string** *port-string*] [**rule-status** {**active** | **not-inservice** | **not-ready**}] [**storage-type** {**non-volatile** | **volatile**}] [**vlan** *vlan*] | [**drop** | **forward**] [**dynamic-pid** *dynamic-pid*] [**cos** *cos*] [**admin-pid** *admin-pid*] [**syslog** {**enable** | **disable**}] [**-verbose**] [**trap** {**enable** | **disable**}] [**disable-port** {**enable** | **disable**}] [**usage-list**] [**display-if-used**]
In switch command mode, display all policy classification capabilities for this device.	**show policy capability**
In switch 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**]

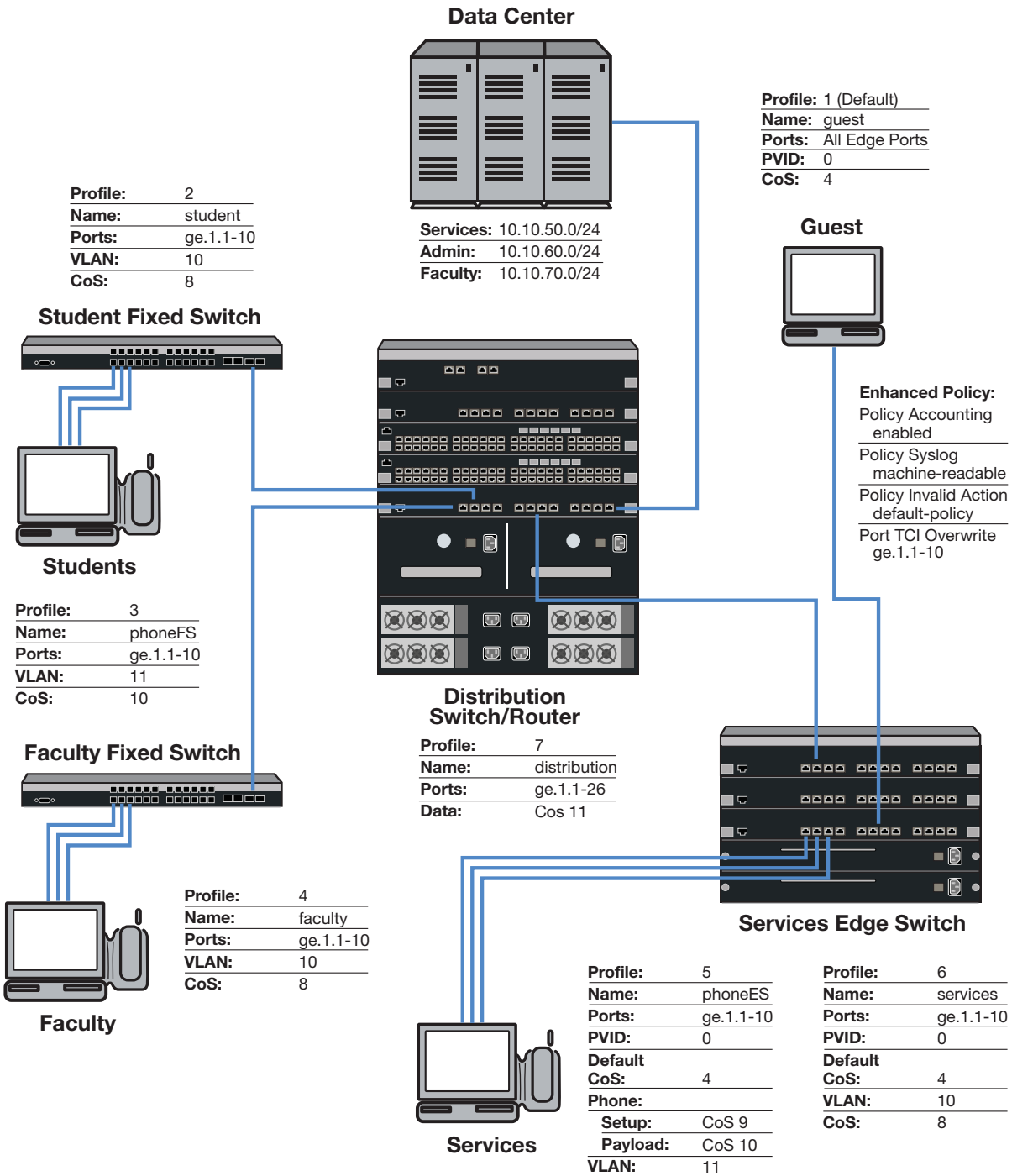
Table 14-4 Displaying Policy Configuration and Statistics (continued)

Task	Command(s)
In switch 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 switch command mode, display disabled ports for all rule entries.	**show policy disabled-ports**
In switch command mode, display the current state of the autoclear feature.	**show policy autoclear** {**all** | **link** | **interval** | **profile** | **ports**}
In switch command mode, display status of dynamically assigned roles.	**show policy dynamic** {[**syslog-default**] [**trap-default**]}

Policy Configuration Example

This section presents a college-based policy configuration example. Figure 14-1 displays an overview of the policy configuration. This overview display is followed by a complete discussion of the configuration example.

Figure 14-1 College-Based Policy Configuration



Note: For purposes of this discussion, Edge Switch and Distribution Switch refer to N-Series platforms.

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 SecureStack 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.

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 four domains: basic edge, standard edge on the Fixed Switch, premium edge on the Services Edge Switch, and premium distribution on the Distribution 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 14-12 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

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

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 N-Series" on page 31-8*.

Note: CLI command prompts used in this configuration example have the following meaning:

- Enterasys(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 N-Series device.
- Distribution(rw)-> - Input on the distribution N-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:

```
Enterasys(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).

```
Enterasys(rw)->set policy rule 1 udpsourceport 68 mask 16 forward
Enterasys(rw)->set policy rule 1 udpdestportIP 53 mask 16 forward
Enterasys(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.

```
Enterasys(rw)->set policy rule 1 tcpdestportIP 80 mask 16 forward
Enterasys(rw)->set policy rule 1 tcpdestportIP 443 mask 16 forward
Enterasys(rw)->set policy rule 1 tcpdestport 8080 mask 16 forward
```

ARP forwarding is required on ether port 0x806.

```
Enterasys(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.

```
Enterasys(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 incase 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 incase 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
```

This completes the policy configuration for this school example.

Terms and Definitions

Table 14-5 lists terms and definitions used in this policy configuration discussion.

Table 14-5 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 Enterasys 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 14-5 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.

15

Multicast Configuration

This document describes the multicast feature and its configuration on N-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 Enterasys 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. For more information, see "Internet Group Management Protocol (IGMP)" on page 15-2.

Multicast routers communicate among themselves using a multicast routing protocol, such as DVMRP or PIM-SM. These protocols calculate a multicast distribution tree 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 15-5 and "Protocol Independent Multicast (PIM)" on page 15-11.

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.

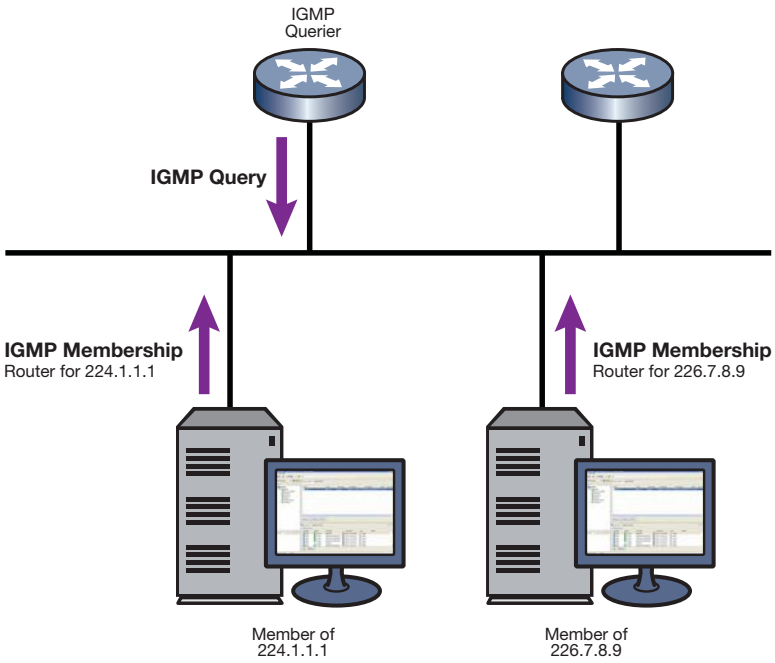
The purpose of IP multicast group management is to optimize a switched network's performance so multicast packets will only be forwarded to those ports containing multicast group hosts or multicast switch devices instead of flooding to all ports in the subnet (VLAN).

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 15-1 IGMP Querier Determining Group Membership



As shown in Figure 15-1, a multicast-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 Enterasys Devices

Enterasys 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 Enterasys device, IGMP can be configured independently at the switch level (Layer 2) and at the router level (Layer 3).

Enterasys 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 snooping is disabled by default on Enterasys devices. You can automatically enable it using the **set igmp enable** command as described in "Configuring IGMP" on page 15-18.

- 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 IGMP messages to the device. You can configure a static IGMP group using the **set igmp static** command as described in "Configuring IGMP" on page 15-18.

Example: Sending a Multicast Stream

Figure 15-2 Sending a Multicast Stream with No Directly Attached Hosts

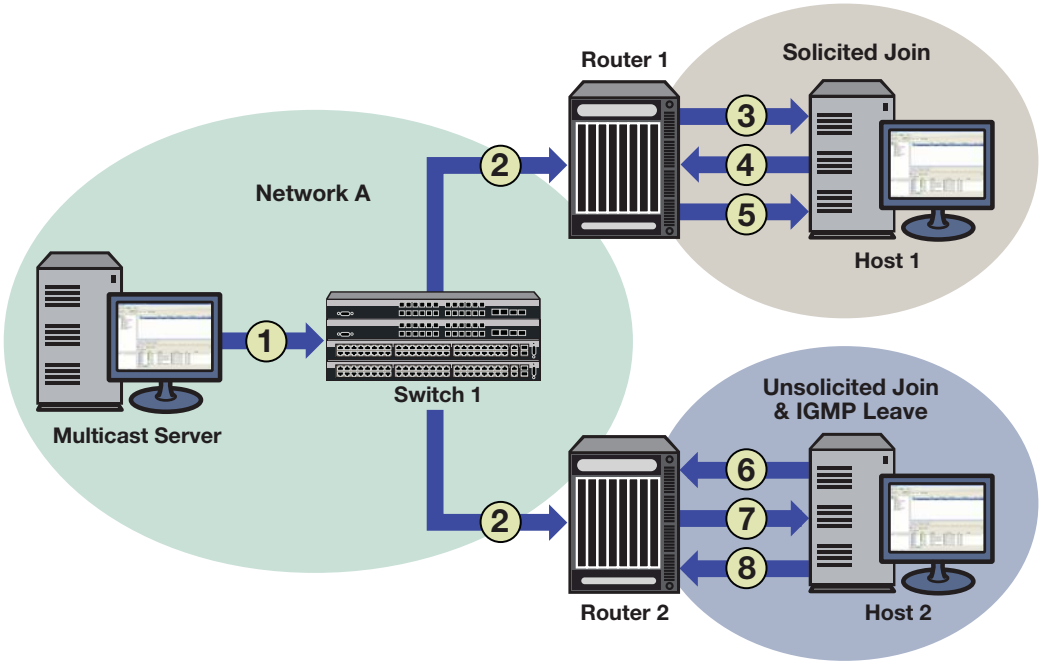


Figure 15-2 provides an example of IGMP processing on Enterasys devices when there are no directly attached hosts.

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 which are linked to Router 1 and Router 2.

   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 15-2, 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 15-2, 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 Enterasys Devices

DVMRP routing is implemented on Enterasys devices as specified in RFC 1075 and *draft-ietf-idmr-dvmrp-v3-10.txt*.

Enterasys 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 example, the DVMRP route table contains a source network, hop count, route uptime, neighbor expiration time, associated interface, and associated IP address.

```
N 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.

```
N 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).

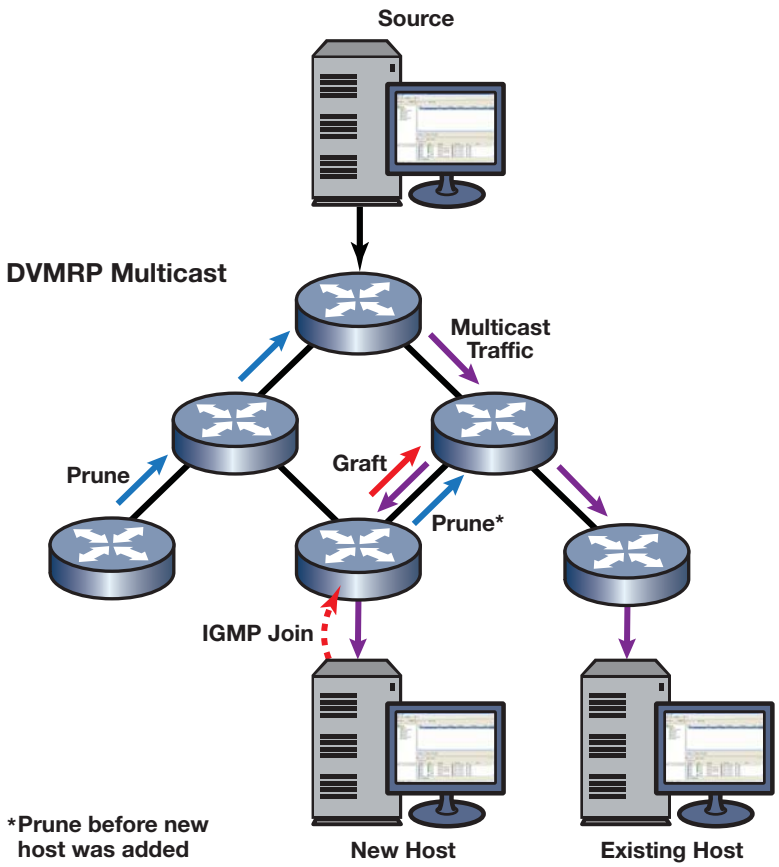
3. Adds the graft to the retransmission timer list awaiting an acknowledgment.
4. Formulates and transmits the graft packet.

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 15-3 shows the DVMRP pruning and grafting process.

Figure 15-3 DVMRP Pruning and Grafting



Protocol Independent Multicast (PIM)

Overview

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.

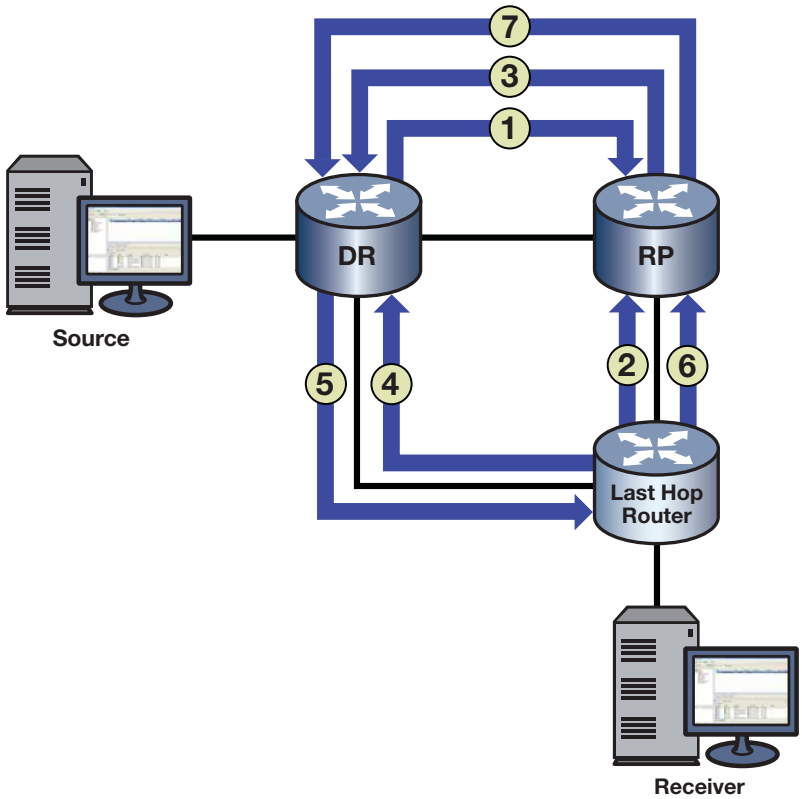
PIM relies on IGMP technology to determine group memberships and uses existing unicast routes to perform reverse path forwarding (RPF) checks, which are, essentially, a route lookup on the source. Its routing engine then returns the best interface, regardless of how the routing table is constructed. 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, a shared-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 15-4 illustrates the PIM traffic flow.

Figure 15-4 PIM Traffic Flow



1. The source's DR registers (that is, 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 Enterasys Devices

Enterasys 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. Enterasys devices support sparse mode (PIM-SM) 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.

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 interlace
- 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

PIM-SM Message Types

Enterasys 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.

Enterasys 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.

Anycast-RP

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. The purpose of anycast-RP is to provide a means of fast convergence when a PIM RP router fails.

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 15-5 on page 15-16 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

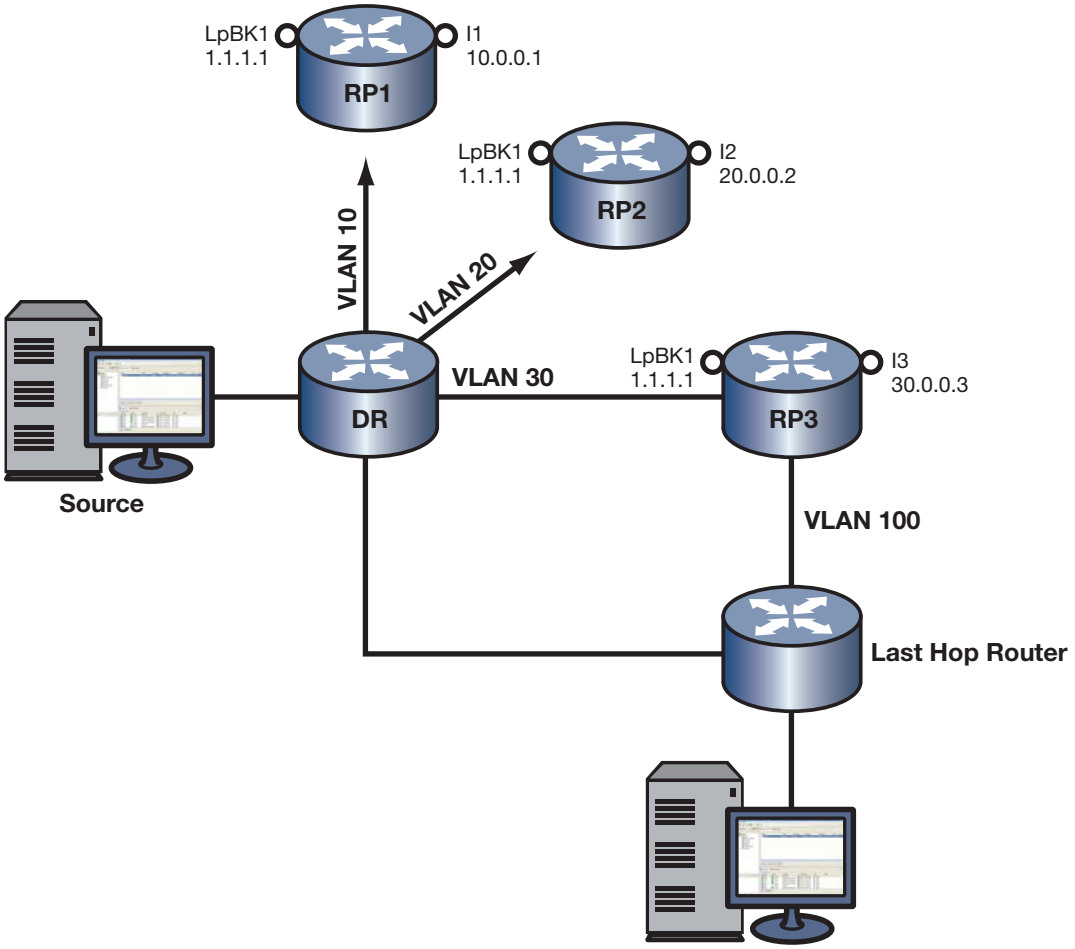
- Create and enable VLAN 20 with IP interfaces
- Configure the underlying unicast routing protocol (OSPF)
- Enable IGMP on VLAN 20
- Configure interface loopback 1 with the anycast-RP address 1.1.1.1/32
- Configure interface loopback 2 with the peer-address 20.0.0.1/32
- Configure 1.1.1.1 as either a 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 15-5 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 Terms and Definitions

Table 15-1 lists terms and definitions used in PIM configuration.

Table 15-1 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.
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.
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.

Table 15-1 PIM Terms and Definitions (continued)

Term	Definition
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.

Configuring Multicast

This section provides the following information about configuring multicast.

Configuring IGMP

IGMP is configured in system mode on N-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 15-2 lists the IGMP configuration commands for N-Series devices.

Table 15-2 IGMP Configuration Commands

Task	Command
Enable IGMP on one or more VLANs.	**set igmp enable** *vlan-list*
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*]
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*]

Table 15-2 IGMP Configuration Commands (continued)

Task	Command
Set the number of multicast groups supported by the N-Series device. Default = 4048, Maximum = 16384 (Not supported on the DFE Gold module and Standalone device).	**set igmp number-flows** {**default** | **maximum**}
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*]

Basic IGMP Configurations

Procedure 15-1 describes the basic steps to configure IGMP on N-Series devices. This procedure assumes that the VLANs on which IGMP will run have been configured and enabled with IP interfaces.

Procedure 15-1 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*]}
2.	In switch mode, enable IGMP on each VLAN interface.	**set igmp enable** *vlan-list*
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

```
N Chassis(su)->set igmp enable 2, 3
N Chassis(su)->set igmp query-enable 2, 3
```

IGMP Display Commands

Table 15-3 lists Layer 2 IGMP show commands for N-Series devices.

Table 15-3 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**]

Table 15-3 Layer 2 IGMP Show Commands (continued)

Task	Command
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 the number of multicast flows supported by the Enterasys N-Series device.	**show igmp number-flows**

Table 15-4 lists Layer 3 IGMP show commands for N-Series devices.

Table 15-4 Layer 3 IGMP Show Commands

Task	Command
Display IGMP information regarding multicast group membership.	**show ip igmp groups**
Display multicast-related information about a specific interface or all interfaces.	**show ip igmp interface** [**vlan** *vlan-id*]

Configuring DVMRP

DVMRP Configuration Commands

Table 15-5 lists the DVMRP configuration commands for N-Series devices.

Table 15-5 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 Enterasys N-Series devices and attached interfaces.

Procedure 15-2 describes the basic steps to configure IGMP on N-Series devices. This procedure assumes that the VLANs on which DVMRP will run have been configured and enabled with IP interfaces.

Procedure 15-2 Basic DVMRP Configuration

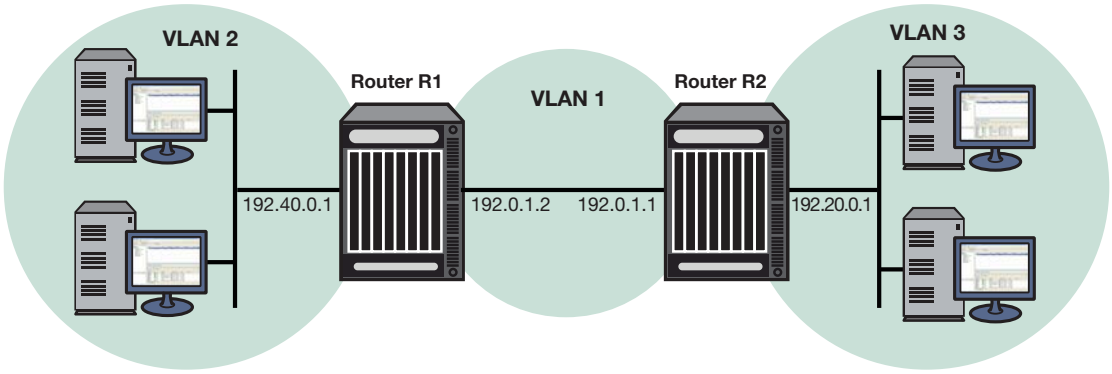
Step	Task	Command
1.	Configure IGMP for each VLAN interface.	**set igmp config** *vlan-list* {[**query-interval** *query-interval*] [**igmp-version** *igmpversion*] [**max-resp-time** *max-resp-time*] [**robustness** *robustness*] [**last-mem-int** *last-mem-int*]}
2.	Enable IGMP on each VLAN interface.	**set igmp enable** *vlan-list*
3.	Enable DVMRP on each of the VLANs specified in step 2.	**ip dvmrp**

Example DVMRP Configuration

Figure 15-6 illustrates the DVMRP configuration of two N-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 15-6 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 15-6 lists the DVMRP show commands for N-Series devices.

Table 15-6 DVMRP Show Commands

Task	Command
Display information about the routes in the DVMRP routing table.	**show ip dvmrp route**
Display the IP multicast routing table.	**show ip mroute** [*unicast-source-address* | *multicast-group-address*] [**summary**]

Refer to the *Enterasys Matrix N-Series CLI Reference* for an example of each command's output.

Configuring PIM

Important Notice

PIM is an advanced routing feature that must be enabled with a license key. If you have purchased an advanced license key, and have enabled routing on the device, you must activate your license as described in the *Enterasys Matrix N-Series CLI Reference* in order to enable the PIM command set. If you wish to purchase an advanced routing license, contact Enterasys Networks Sales.

PIM Configuration Commands

Table 15-7 lists the PIM configuration commands for Enterasys N-Series devices.

Table 15-7 IPv4 PIM 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**

Table 15-7 IPv4 PIM Commands

Task	Command
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*

Table 15-8 lists the PIM IPv6 configuration commands for Enterasys N-Series devices.

Table 15-8 IPv6 PIM 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**
Enable PIM-SSM on a routing interface. 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**

Table 15-8 IPv6 PIM Commands

Task	Command
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*

Basic PIM 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 N-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 Version 3 for interfaces with IGMP reporters.
4. Configuring PIM-SM and/or PIM-SSM on the VLANs.

Procedure 15-3, which describes the basic steps the configure PIM-SM on an N-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 15-18.

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 15-3 Basic PIM Configuration

Step	Task	Command(s)
1.	If desired, change the DR priority of one or more interfaces on the Enterasys N-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*

Procedure 15-3 Basic PIM Configuration

Step	Task	Command(s)
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 Enterasys N-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 Enterasys N-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 N-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 15-9 lists the PIM IPv4 and IPv6 display commands for Enterasys N-Series devices.

Table 15-9 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**
Display RP anycast information for all or a specified RP.	**show** {**ip** | **ipv6**} **pim anycast-rp** [**rp-address** *rp-address*]
Display BootStrap Router (BSR) information.	**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**]

Table 15-9 PIM IPv4 and IPv6 Display Commands

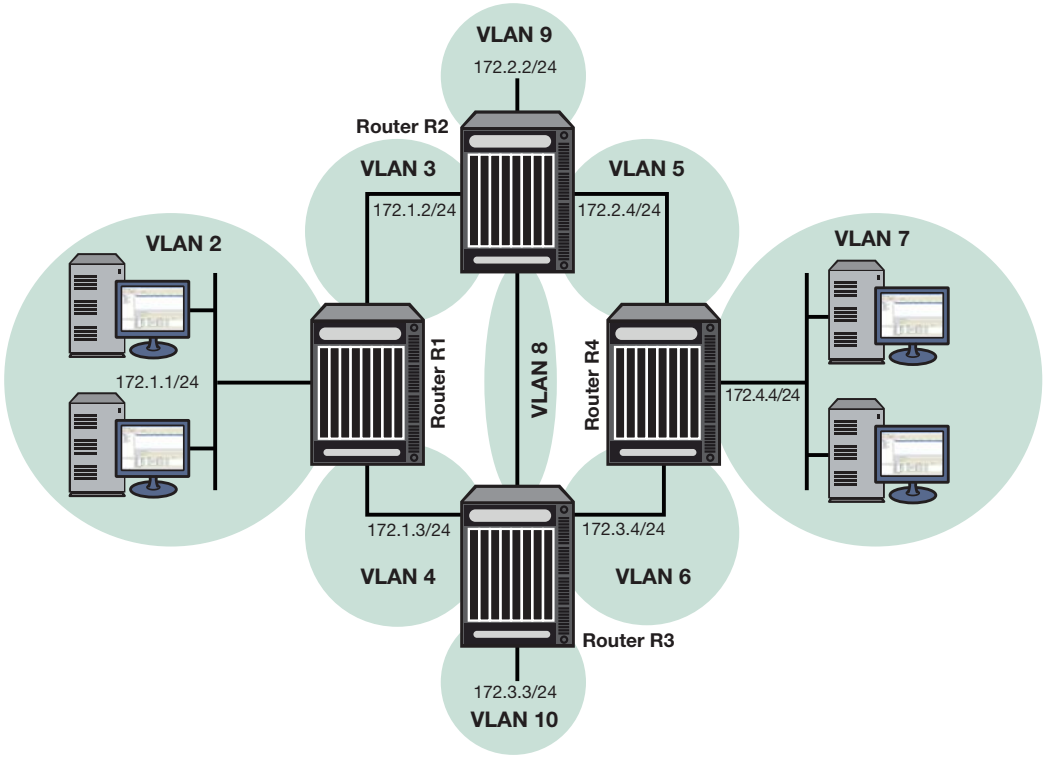
Task	Command
Display the PIM multicast route (*,G and S,G) table.	**show** {**ip**|**ipv6**} **pim mrt** [**source** *source* | **group** *group*] [**interface**] [**detail**] [**brief**] [**summary**]
Display the PIM multicast route (*,G and S,G) table by type.	**show** {**ip**|**ipv6**} **mrt type** {**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.	**show** {**ip**|**ipv6**} **pim rp** [**mapping**]
Display the rendezvous point (RP) selected for a specified group.	**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 *Enterasys Matrix N-Series CLI Reference* for a description of the output of each command.

Example PIM-SM Configuration

Figure 15-7 illustrates the PIM-SM configuration of four N-Series routers shown in the example scripts below. This 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 15-7 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 N-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

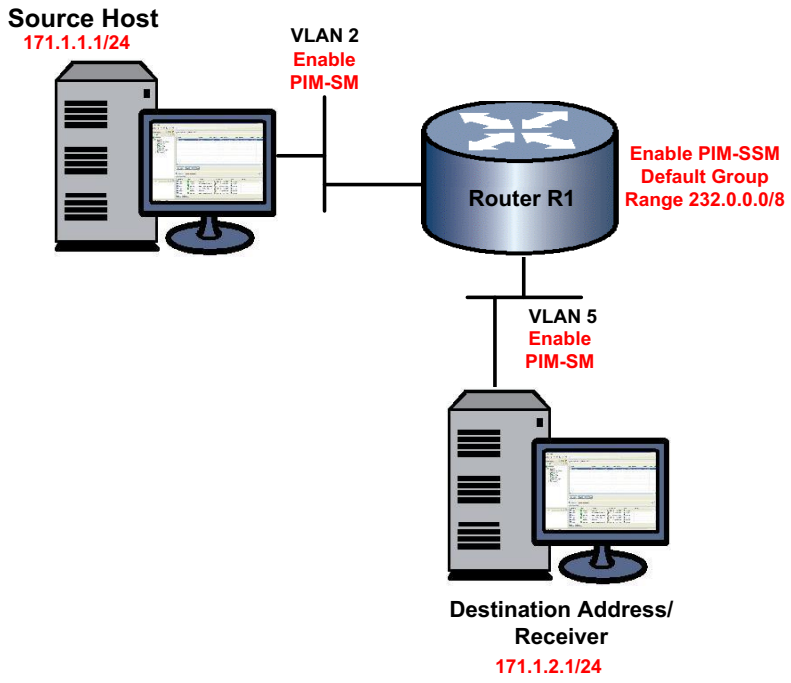
This router does not play any special role in PIM-SM, except that it has hosts directly connected to it. IGMP is enabled on the interface that connects to hosts and PIM-SM is enabled on all interfaces.

```
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 15-8 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 15-8 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 on VLAN 2 and VLAN 5. IGMP is used to determine host group membership on directly attached subnets. Note that IGMP is enabled in switch mode on N-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
R1(su)->set igmp enable 5
R1(su)->set igmp query-enable 5
```

16

System Logging Configuration

This chapter provides the following information about configuring and monitoring Syslog on Enterasys N-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 N-Series Switches

By default, Syslog is operational on N-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 Enterasys 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 N-Series platforms.

Configuring Syslog Message Disposition

The Syslog implementation on N-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 Enterasys 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 16-1 describes the Enterasys implementation of key Syslog components.

Table 16-1 Syslog Terms and Definitions

Term	Definition	Enterasys Usage
Facility	Categorizes which functional process is generating an error message. Syslog combines this value and the severity value to determine message priority.	Enterasys 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.	Enterasys 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 Enterasys 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.

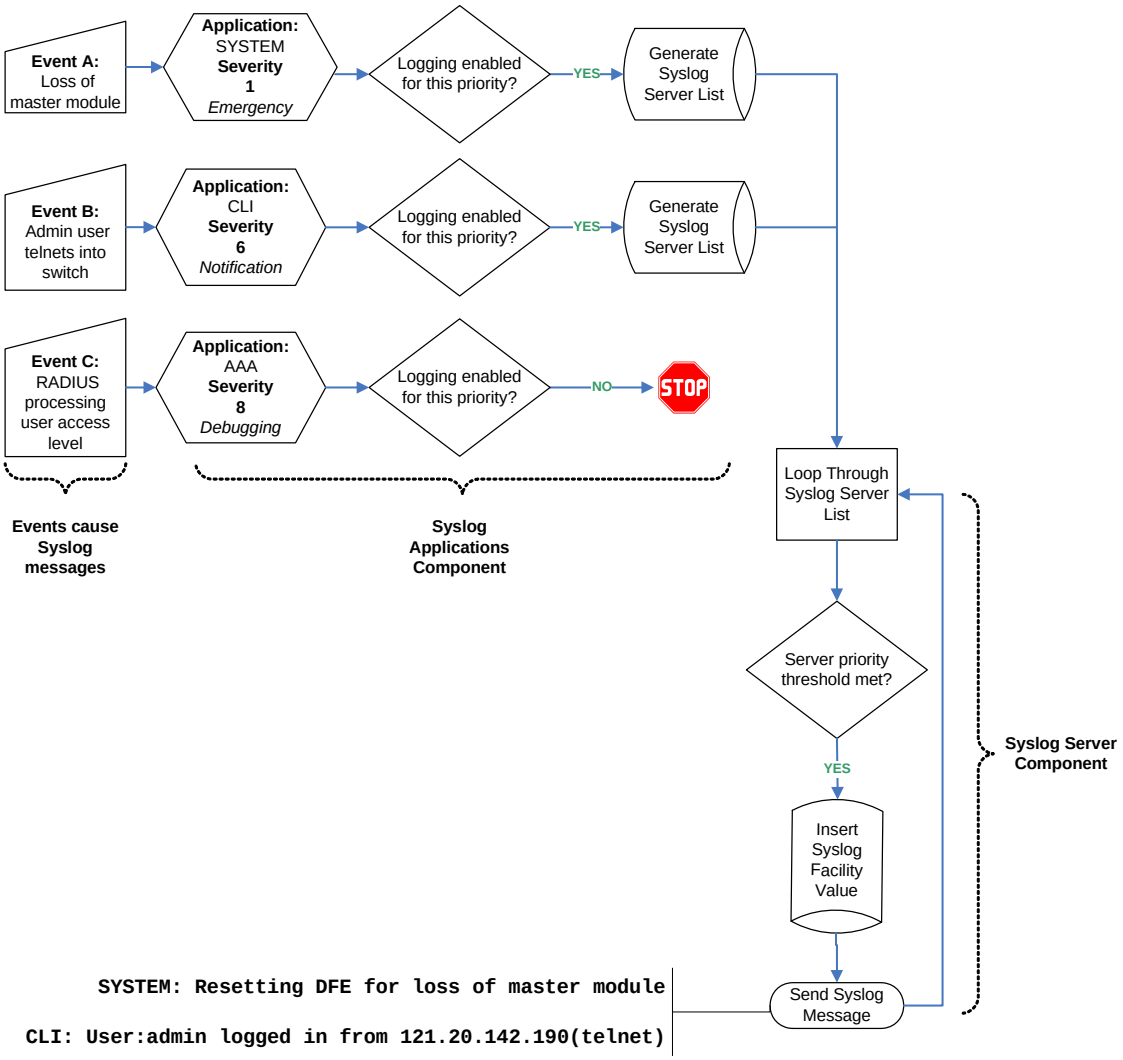
Table 16-1 Syslog Terms and Definitions (continued)

Term	Definition	Enterasys Usage
Application	Client software applications running on devices that can generate Syslog messages.	Enterasys supported applications and their associated CLI mnemonic values include: **CLI** - Command Line Interface **SNMP** - Simple Network Management Protocol **Webview** - Enterasys 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.	Enterasys 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 16-1 on page 16-5 shows a basic scenario of how Syslog components operate on an Enterasys switch. By default, all applications running on the Enterasys 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 16-1 Basic Syslog Scenario



Default application settings in the example in Figure 16-1 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 16-7, and the “Configuration Examples” on page 16-11.

Interpreting Messages

Every system message generated by the N-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.

```
N 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 16-2 describes the components of these messages.

Table 16-2 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 Enterasys 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 16-3 lists basic Syslog commands and their order of precedence on Enterasys switches.

Table 16-3 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* [**severity** *severity*]	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 16-4 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.

```
N 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 16-4 for its configured Syslog servers:

Table 16-4 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 16-4. Refer back to Filtering by Severity and Facility and to Table 16-1 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:

```
N 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:

```
N Chassis(rw)->set logging default facility local2 severity 4
```

Reviewing and Configuring Logging for Applications

By default, all applications running on N-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.

```
N 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 16-1 for sample CLI mnemonic values.

Modifying Severity Levels and Assigning Syslog Servers for Applications

Applications running on Enterasys 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.

```
N Chassis(rw)->set logging application SSH level 5 server 1
```

Enabling Console Logging and File Storage

N-Series devices allow you to display logging messages to the console and save to a persistent file. In addition, N-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**}

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**.

Displaying to the Current CLI Session

To display logging to the current CLI console session on a N-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.

Configuration Examples

Enabling a Server and Console Logging

Procedure 16-1 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 16-1 Configuring a Server and Console Logging

Step	Task	Command(s)
1.	Configure Syslog server 1 and accept default settings (listed in Table 16-4 on page 16-8).	**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 16-2 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 16-2 Adjusting Settings for an Application

Step	Task	Command(s)
1.	Configure Syslog server 2 and accept default settings (listed in Table 16-4 on page 16-8).	**set logging server 2 ip-addr 10.1.1.3 state enable**

Procedure 16-2 Adjusting Settings for an Application (continued)

Step	Task	Command(s)
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**

17

Network Monitoring Configuration

This document describes the network monitoring features and their configuration on Enterasys N-Series devices.

Using Network Monitoring in Your Network

N-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 <xx> 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.

```
N Chassis(rw)->set history 25
```

Use the **show history** command in any command mode to display the currently configured size of the history buffer.

```
N Chassis(rw)->show history
History size currently set to: 25
N Chassis(rw)->
```

Use the **history** command in any command mode to display the contents of the history buffer.

```
N Chassis(rw)->history
 1 history
 2 show gvrp
 3 show vlan
 4 show igmp
N Chassis(rw)->
```

Network Diagnostics

N-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.

```
N 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.

```
N 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.enterasys.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.

```
N 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.

```
N 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
N 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.

```
N Chassis(rw)->show users
  Session  User  Location
  -------- ----- -------------------------
* console  admin console (via com.1.1)
  telnet   rw    134.141.192.18
N Chassis(rw)->
```

Use the **tell** command to send a message to one or all users on the switch.

```
N 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.

Table 17-1 lists:

- The nine RMON monitoring groups supported on N-Series devices
- Each group's function
- The elements it monitors
- The group's associated commands

Table 17-1 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**
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 17-1 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**
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**
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 equation. These matched packets form a data stream or “channel” that may be captured or may generate events.	Packets matching the filter configuration.	**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 configuration.	**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 N-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.

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:

```
N Chassis(rw)->set smon priority create ge.1.1
N Chassis(rw)->set smon priority enable ge.1.1
N 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

N 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:

```
N Chassis(rw)->set smon vlan create ge.1.1
N Chassis(rw)->set smon vlan enable ge.1.1
N 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

N Chassis(rw)->
```

Configuring Network Monitoring

This section provides details for the configuration of network monitoring on the N-Series products.

Table 17-2 lists network monitoring parameters and their default values.

Table 17-2 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
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.

Table 17-2 Default Network Monitoring Parameters (continued)

Parameter	Description	Default Value
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:

```
N 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:

```
N Chassis(rw)->show users
  Session  User  Location
  -------- ----- -------------------------
* console  admin console (via com.1.1)
  telnet   rw    134.141.192.18
N 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 17-3 describes network diagnostics commands.

Table 17-3 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.	**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 17-3 Network Diagnostics Commands (continued)

Task	Command
• **-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.	
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.	**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*

Table 17-3 Network Diagnostics Commands (continued)

Task	Command
-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.	
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 17-1 describes how to configure SMON. SMON commands can be entered from any command mode.

Procedure 17-1 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 port(s).	**set smon vlan** {**create** | **enable**} *port-string* [*owner*]

Procedure 17-2 describes how to configure RMON. RMON commands can be entered from any command mode.

Procedure 17-2 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	**set rmon stats** *index* [*port-string*] [*owner*]

Procedure 17-2 Configuring Remote Network Monitoring (continued)

Step	Task	Command(s)
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	**set rmon history** *index* [*port-string*] [**buckets** *buckets*] [**interval** *interval*] [**owner** *owner*]
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**

Procedure 17-2 Configuring Remote Network Monitoring (continued)

Step	Task	Command(s)
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**
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**

Procedure 17-2 Configuring Remote Network Monitoring (continued)

Step	Task	Command(s)
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**
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*]

Procedure 17-2 Configuring Remote Network Monitoring (continued)

Step	Task	Command(s)
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*]
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	**set rmon capture** *index* {*channel* [**action** {**lock** | **wrap**}] [**slice** *slice*] [**loadsize** *loadsize*] [**offset** *offset*] [**asksize** *asksize*] [**owner** *owner*]}

Table 17-4 describes how to manage network monitoring.

Table 17-4 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 port(s), 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*
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 17-5 describes how to display network monitoring information and statistics.

Table 17-5 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 port(s) and 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**}]

Table 17-5 Displaying Network Monitoring Information and Statistics (continued)

Task	Command
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**]

18

NetFlow Configuration

This document describes the NetFlow feature and its configuration on Enterasys N-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 N-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:

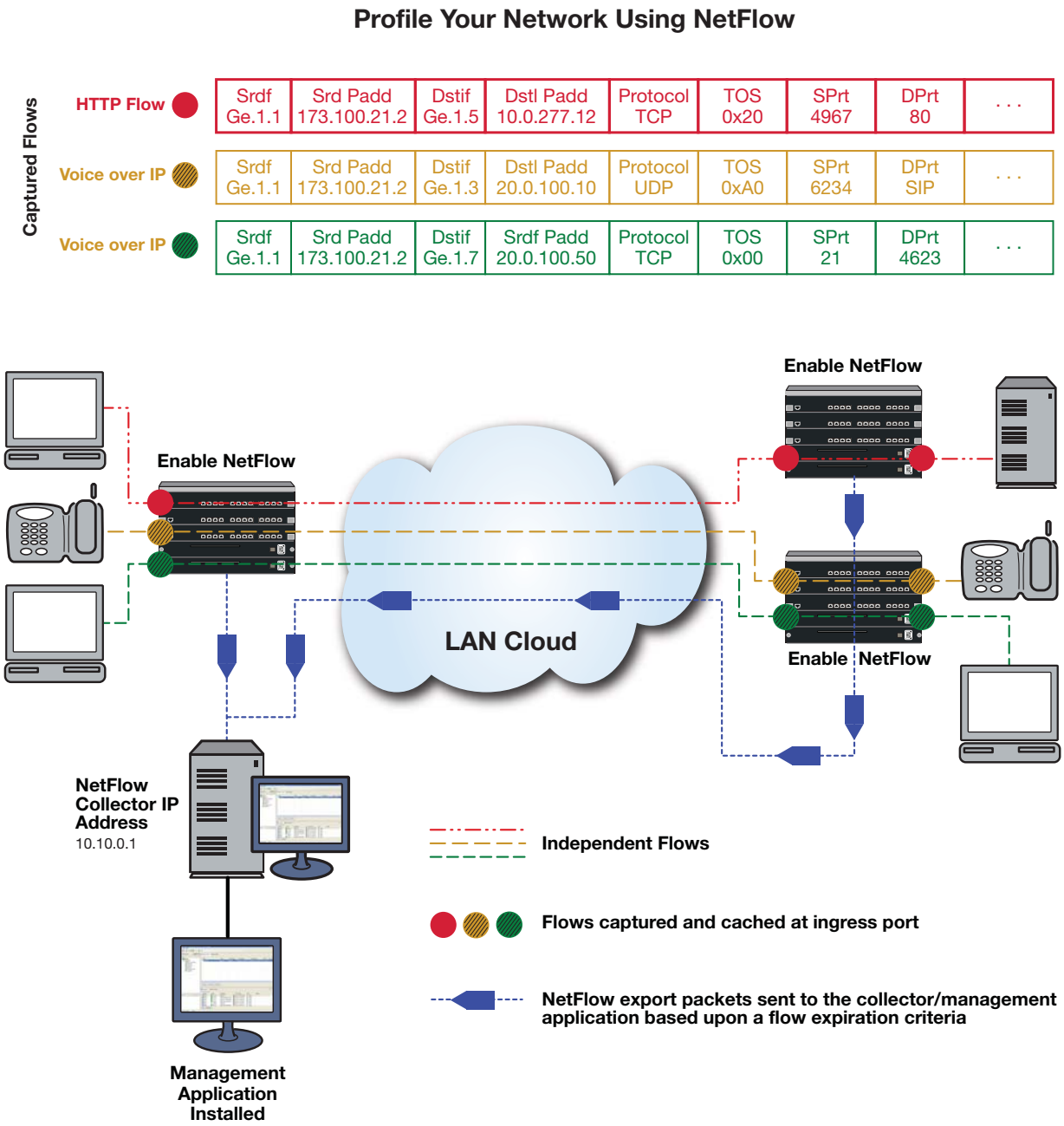
- Provides a means to profile all flows on your network over a period of time. A network profile provides the granularity of insight into your network necessary for such secure network functionality as establishing roles with policy and applying QoS to policy.
- Provides a means of isolating the source 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 18-1 illustrates an example of a NetFlow network profile setup.

Figure 18-1 NetFlow Network Profile Example



To complete a NetFlow network profile, enable NetFlow on all ports where packet flows aggregate. At the top of Figure 18-1 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 Enterasys 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 18-11 for NetFlow Version 5 template data field descriptions and "NetFlow Version 9 Templates" on page 18-12 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 18-6).

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.

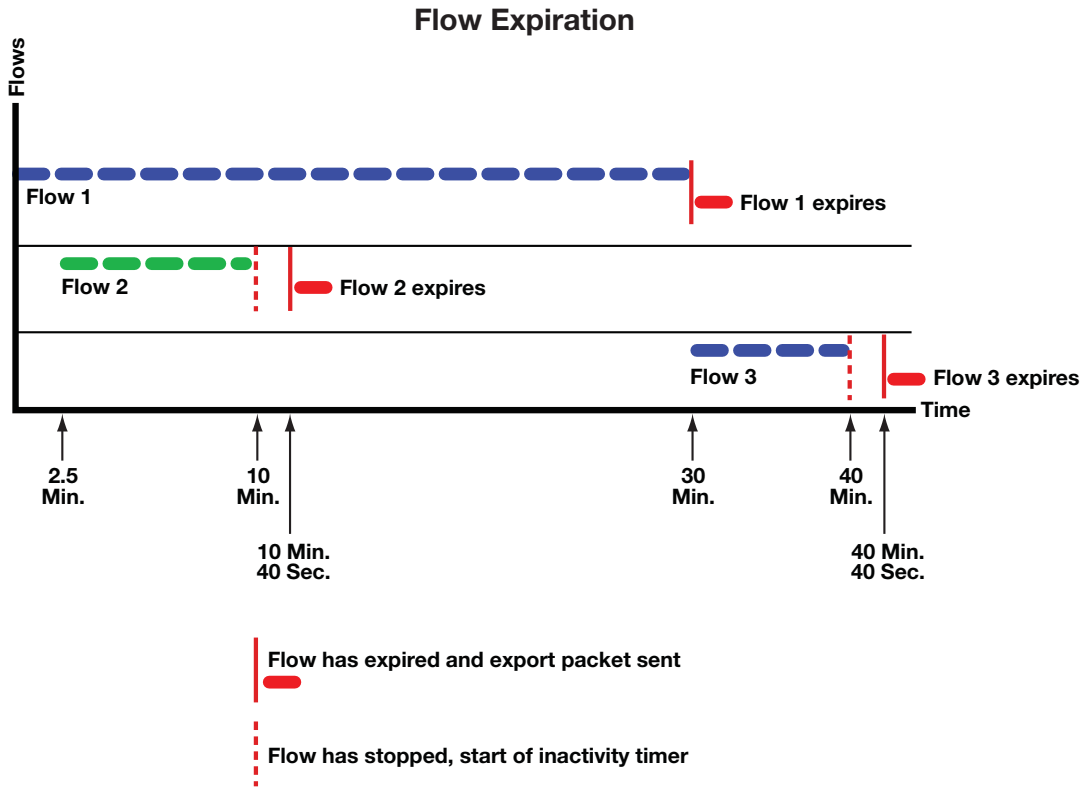
Rules for expiring NetFlow cache entries include:

- 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 18-2 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 18-2 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 N-Series

The N-Series modules support NetFlow. NetFlow is disabled by default on all devices at device startup.

Enterasys N-Series Implementation

The Enterasys N-Series flow-based architecture provides a powerful mechanism for collecting network flow statistics, with reporting capacity that scales with the addition of each N-Series module. For each flow, packet and byte count statistics are collected by the N-Series forwarding hardware. The flow report generation logic is distributed, permitting each module to report flows on its own ports.

The Enterasys N-Series implementation enables the collection of NetFlow data on both switched and routed frames, allowing N-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 N-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 routed, the N-Series firmware reports the source and destination ifIndexes as the physical ports, not 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 Enterasys N-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 N-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 Enterasys 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 N-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 18-7.

NetFlow Version 9 records generated by N-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 Enterasys Version 9 implementation:

- Does not support aggregation caches.
- Provides 15 IPv4 and 15 IPv6 predefined templates. The N-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 18-5 on page 18-13 for a listing of the header fields supported by the NetFlow Version 9 templates. See Table 18-6 on page 18-13 for a listing of the base data record fields supported by all NetFlow Version 9 templates. See Table 18-7 on page 18-14 for a listing of the additional template specific data record fields supported by the NetFlow Version 9 templates. See Table 18-8 on page 18-14 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 N-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 N-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 N-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 N-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 N-Series Systems

Table 18-1 provides a listing of the default NetFlow configuration settings for the N-Series systems.

Table 18-1 Default NetFlow Configuration Settings for N-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 18-1 Default NetFlow Configuration Settings for N-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 18-1 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 18-1 Configuring NetFlow on N-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 N-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 18-2 lists terms and definitions used in this NetFlow configuration discussion.

Table 18-2 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 18-2 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 Enterasys 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 18-3 provides a listing and description for the NetFlow Version 5 header fields. Table 18-4 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 18-3 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 18-4 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 18-4 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 N-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 Enterasys 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 N-Series platform implementation of the NetFlow Version 9 templates are detailed in the following tables:

- Table 18-5 on page 18-13 provides a listing and description of the supported NetFlow Version 9 header fields
- Table 18-6 on page 18-13 provides a listing and description of the supported NetFlow Version 9 base data record fields
- Table 18-7 on page 18-14 provides a listing of the supported additional template specific data record fields
- Table 18-8 on page 18-14 provides the template ID and a general description of each N-Series Version 9 template

Table 18-5 on page 18-13 details the NetFlow Version 9 template header fields supported by all Version 9 templates.

Table 18-5 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 18-6 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 18-6 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 18-7 details the additional NetFlow Version 9 data record fields specific to a given Version 9 template.

Table 18-7 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 18-8 provides a description of each IPv4 and IPv6 NetFlow Version 9 template per template ID.

Table 18-8 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 18-8 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 18-8 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.

19

Virtual Routing and Forwarding (VRF) Configuration

This document provides the following information about configuring Virtual Routing and Forwarding (VRF) on the Enterasys N-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

N-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. Enterasys recommends that you provide the VRF with a meaningful name such as “Marketing” or “Internet-Access”.

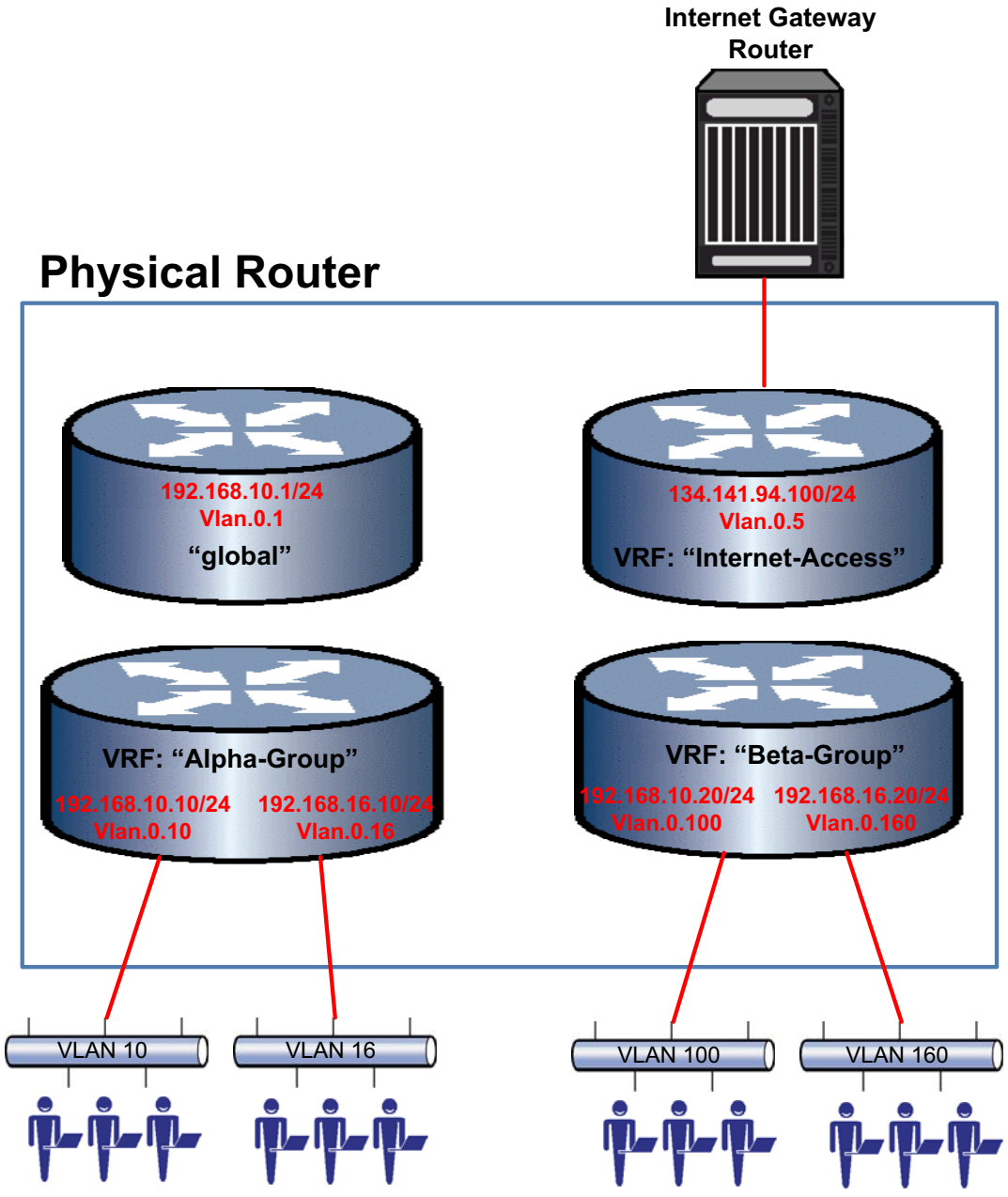
You can optionally specify an SNMPv3 context of up to 31 characters. If not specified, the SNMPv3 context defaults to the name of the VRF instance. If the VRF instance name exceeds 28 characters, the SNMPv3 context must be specified when creating the VRF. Refer to the **set router vrf create** command 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 19-1 on page 19-3 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 19-1 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 19-5 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 currently not supported between two non-global VRF routers. Static routes are supported between the **global** router and any non-global VRF router.

Refer to Figure 19-1 on page 19-3 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:

```
N 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:

```
N 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:

```
N Chassis(rw-vr1-config)->route-map policy 101 permit 20
N Chassis(rw-vr1-config-route-map-pbr)->match ip address 1
N 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 19-2 on page 19-6, 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.

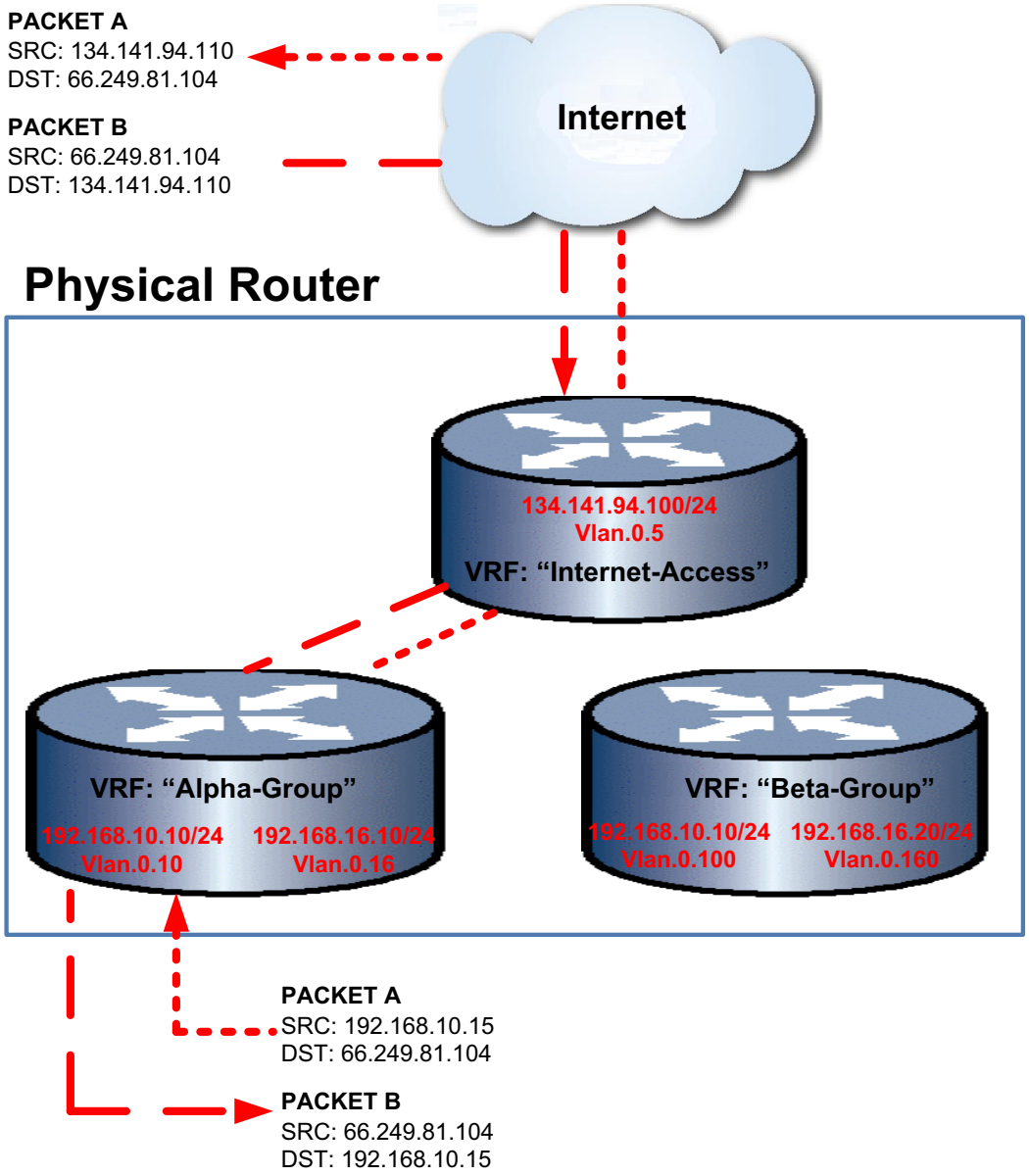
Given the configuration in Figure 19-2, there is a conflict between VRFs Alpha-Group and Beta-Group for any packet sourced outside of the system that needs to be routed to the correct VRF through the shared-access-VRF Internet-Access.

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 19-2 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.

```
N Chassis(su)->router Alpha-Group
N Chassis(su-*ha-Group)->configure
N Chassis(su-*ha-Group-config)->interface vlan 10
N Chassis(su-*ha-Group-config-intf-vlan.0.10)->ip address 192.168.10.1/24
N Chassis(su-*ha-Group-config-intf-vlan.0.10)->ip nat inside
N Chassis(su-*ha-Group-config-intf-vlan.0.10)->exit
N Chassis(su-*ha-Group-config)->exit
N Chassis(su-*ha-Group)->exit
N Chassis(su)->router Internet-Access
N Chassis(su-*t-Access)->configure
N Chassis(su-*t-Access-config)->interface vlan 5
N Chassis(su-*t-Access-config-intf-vlan.0.5)->ip address 134.141.94.100/24
N Chassis(su-*t-Access-config-intf-vlan.0.5)->ip nat outside
N Chassis(su-*t-Access-config-intf-vlan.0.5)->exit
N 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.

```
N Chassis(su)->router Alpha-Group
N Chassis(su-*ha-Group)->configure
N Chassis(su-*ha-Group-config)->interface vlan 10
N Chassis(su-*ha-Group-config-intf-vlan.0.10)->ip address 192.168.10.1/24
N Chassis(su-*ha-Group-config-intf-vlan.0.10)->ip nat inside
N Chassis(su-*ha-Group-config-intf-vlan.0.10)->exit
N Chassis(su-*ha-Group-config)->exit
N Chassis(su-*ha-Group)->exit
N Chassis(su)->router Internet-Access
N Chassis(su-*t-Access)->configure
N Chassis(su-*t-Access-config)->interface vlan 5
N Chassis(su-*t-Access-config-intf-vlan.0.5)->ip address 134.141.94.100/24
N Chassis(su-*t-Access-config-intf-vlan.0.5)->ip nat outside
N Chassis(su-*t-Access-config-intf-vlan.0.5)->exit
N Chassis(su-*t-Access-config)->ip access-list standard dynamic-nat
N Chassis(su-*t-Access-cfg-std-acl-dyna*-nat)->permit host 192.168.10.15
N Chassis(su-*t-Access-cfg-std-acl-dyna*-nat)->exit
N Chassis(su-*t-Access-config)->ip nat pool internet-out 134.141.94.121
134.141.94.129
N 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

N-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.

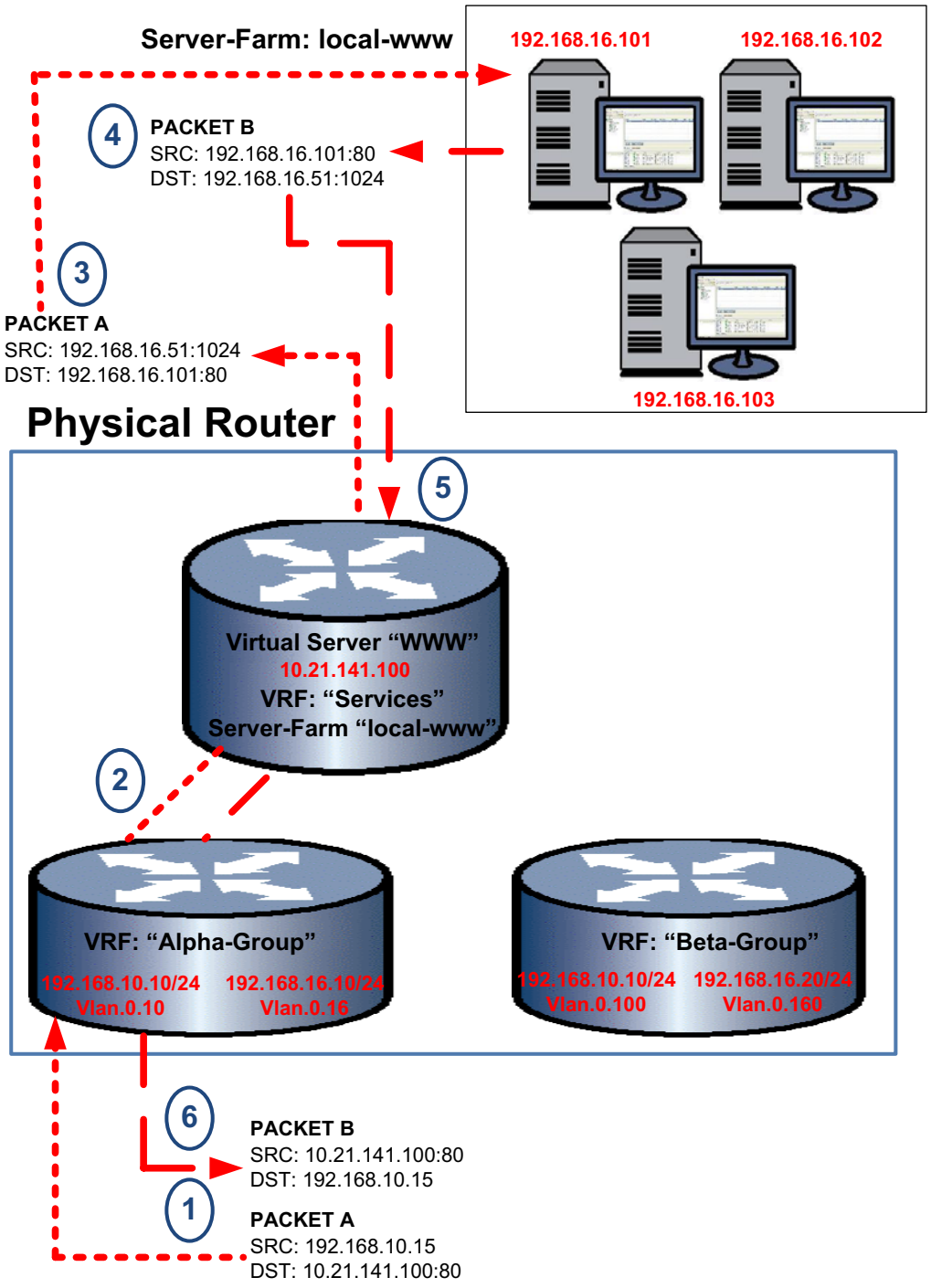
Figure 19-3 on page 19-11 presents an example of an SLB all-VRFs configuration. The packet processing and flow for this example is as follows:

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 19-5 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.

```
N Chassis(su)->router Services
N Chassis(su-Services)->configure
N Chassis(su-Services-config)->ip nat pool local-net 192.168.16.51 192.168.16.55
N Chassis(su-Services-config)->ip slb serverfarm local-www
N Chassis(su-Services-config-slb-sfarm)->real 192.168.16.101
N Chassis(su-Services-config-slb-real)->inservice
N Chassis(su-Services-config-slb-real)->exit
N Chassis(su-Services-config-slb-sfarm)->real 192.168.16.102
N Chassis(su-Services-config-slb-real)->inservice
N Chassis(su-Services-config-slb-real)->exit
N Chassis(su-Services-config-slb-sfarm)->real 192.168.16.103
N Chassis(su-Services-config-slb-real)->inservice
N Chassis(su-Services-config-slb-real)->exit
```

```
N Chassis(su-Services-config-slb-sfarm)->exit
N Chassis(su-Services-config)->ip slb vserver WWW
N Chassis(su-Services-config-slb-vserver)->virtual 10.21.141.100 tcp www all-vrfs
N Chassis(su-Services-config-slb-vserver)->serverfarm local-www
N Chassis(su-Services-config-slb-vserver)->source nat pool local-net
N Chassis(su-Services-config-slb-vserver)->inservice
N Chassis(su-Services-config-slb-vserver)->exit
N Chassis(su-Services-config)->
```

Figure 19-3 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

```
N Chassis(su)->router Alpha-Group
N Chassis(su-*ha-Group)->configure
N Chassis(su-*ha-Group-config)->ip forward-protocol udp
N Chassis(su-*ha-Group-config)->interface vlan.0.10
N Chassis(su-*ha-Group-config-intf-vlan.0.10)->ip helper-address 134.141.95.105
vrf Internet-Access
N Chassis(su-*ha-Group-config-intf-vlan.0.10)->exit
N Chassis(su-*ha-Group-config)->ip dhcp relay information option vpn
N Chassis(su-*ha-Group-config)->
```

Configuring VRF

This section provides details for the configuration of VRF on the N-Series products.

Table 19-1 lists VRF parameter default values.

Table 19-1 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 19-1 describes how to configure VRF.

Procedure 19-1 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 19-1 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.	**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 19-2 lists terms and definitions used in this VRF configuration discussion.

Table 19-2 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 19-2 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.

20

IP Routing Configuration

This document describes IPv4 and IPv6 routing configuration on Enterasys N-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 19, **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:

```
N Chassis(rw)->configure
N Chassis(rw--config)->show router
Router Services are currently running on module 1.
VRF Context    : global
RD             : not set
N 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:

```
N Chassis(rw)->router global
N Chassis(rw-router)->configure
N Chassis(rw-router-config)->show router
Router Services are currently running on module 1.
VRF Context    : global
RD             : not set
N Chassis(rw-router-config)->
```

Table 20-1 describes how to enter router configuration mode.

Table 20-1 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. The following example displays a sample output of the **show limits** command:

```
N 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


N 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:

```
N 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
.
.
.
N 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 12-9 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. IPv6 forwarding is currently not configurable on the N-Series platform.

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:

```
N Chassis(rw-config-intf-vlan.0.99)->ip address 99.0.0.1/24
N 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
N Chassis(rw-config-intf-vlan.0.99)->ip address 99.0.0.2/16 secondary
N 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
N 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 20-5 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

```
N Chassis(rw)->configure
N Chassis(rw-config)->interface vlan 1
N Chassis(rw-config-intf-vlan.0.1)->ip address 10.21.130.59 255.255.128.0
N Chassis(rw-config-intf-vlan.0.1)->no shutdown
N Chassis(rw-config-intf-vlan.0.1)->exit
N Chassis(rw-config)->
```

See the current firmware release notes for the number of routing interfaces supported on an N-Series routing module. Each interface can be configured for the RIP 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.

N-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 N-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 20-6 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 20-4 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 N-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:

```
N Chassis(su)->configure
N Chassis(su-config)->ipv6 general-prefix Doc-Prefix 2001:11ac:fd34::/48
N Chassis(su-config)->show ipv6 general-prefix
  ipv6 general-prefix Doc-Prefix 2001:11ac:fd34::/48
N Chassis(su-config)->interface vlan 51
N Chassis(su-config-intf-vlan.0.51)->ipv6 address Doc-Prefix ::50:0:0:abcd:33/64
N 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
...
N 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:

```
N Chassis(su-config)->interface vlan 50
```

```
N Chassis(su-config-intf-vlan.0.50)->ipv6 address
ba10:1100:aa11:c171:0:0:1111:00/48
N Chassis(su-config-intf-vlan.0.50)->
```

This example sets the IPv6 link local address for interface VLAN 50 to fe80:1234:5678::300:

```
N Chassis(su-config)->interface vlan 50
NChassis(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
N 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:

```
N Chassis(su-config)->interface vlan 50
N Chassis(su-config-intf-vlan.0.50)->ipv6 address 2001:febd:1234:0/64 eui-64
N 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]
...
N Chassis(su-config-intf-vlan.0.50)->
```

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.

The **clear ip address** command will not clear the interface the IP address is assigned to. Use the **clear ip interface** command to clear the IP interface the IP address is assigned to.

The following example clears the IP address **125.100.10.1** assigned to VLAN 5 in the example above and then deletes IP interface VLAN **5**:

```
N Chassis(rw)->clear ip address 125.100.10.1
N 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:

```
N Chassis(rw)->configure
N Chassis(rw-config)->interface vlan.0.1
N Chassis(rw-config-intf-vlan.0.1)->ip address 125.50.10.1/16
N Chassis(rw-config-intf-vlan.0.1)->no ip forwarding
N Chassis(rw-config-intf-vlan.0.1)->no ip proxy-arp
N Chassis(rw-config-intf-vlan.0.1)->no shutdown
N Chassis(rw-config-intf-vlan.0.1)->exit
N Chassis(rw-config)->interface vlan.0.5
N Chassis(rw-config-intf-vlan.0.5)->ip address 125.100.10.1/16
N Chassis(rw-config-intf-vlan.0.5)->no ip forwarding
N Chassis(rw-config-intf-vlan.0.5)->no ip proxy-arp
N Chassis(rw-config-intf-vlan.0.5)->no shutdown
N Chassis(rw-config-intf-vlan.0.5)->exit
N Chassis(rw-config)->
```

Note: The **ip forwarding** command is used to enable or disable IPv4 forwarding. IPv4 forwarding is enabled by default on all IPv4 interfaces. IPv6 forwarding is disabled by default on all IPv6 interfaces. The configuration of IPv6 forwarding is not currently supported on the N-Series platform.

The above example is replicated below using the **set ip address** command in system command mode:

```
N Chassis(rw)->set ip address 125.50.10.1 mask 255.255.0.0 interface vlan.0.1
N Chassis(rw)->set ip address 125.100.10.1 mask 255.255.0.0 interface vlan.0.5
N 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:

```
N Chassis(rw-config)->interface vlan 1
N Chassis(rw-config-intf-vlan.0.1)->ipv6 address
ba10:1100:aa11:c171:0:0:1111:1/48
N Chassis(rw-config-intf-vlan.0.1)->no shutdown
N Chassis(rw-config-intf-vlan.0.1)->exit
N Chassis(rw-config)->interface vlan.0.5
N Chassis(rw-config-intf-vlan.0.5)->ipv6 address
ba10:1100:aa11:c171:0:0:111:5/48
N Chassis(rw-config-intf-vlan.0.5)->no shutdown
N Chassis(rw-config-intf-vlan.0.5)->exit
N 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.

```
N 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.

```
N 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
N Chassis(rw-config)->
```

This example shows how to display IPv6 configuration information for VLAN 51:

```
N 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
N Chassis(rw)->
```

Procedure 20-1 describes how to configure the routing interface.

Procedure 20-1 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, IPv4 address for this interface, in interface configuration command mode.	**ip address** {*ip-address* | *ip-address*/*prefixLength*} *ip-mask* [**primary** | **secondary**]
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.	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. An IP static route can be configured as a non-forwarding management route for IPv6.

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 20-14 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.

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. The N-Series device only supports IPv6 static routes in a non-forwarding host context. See "Traffic Non-Forwarding IP Static Routes" on page 20-14 for details on configuring non-forwarding IP static routes.

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**.

```
N 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:

  ```
  N 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:

  ```
  N 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:

```
N 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.

```
N 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:

```
N 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**:

```
N 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:

```
N 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 20-12.

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:

```
N Chassis(rw-config)->interface vlan.0.4000
N Chassis(rw-config-intf-vlan.0.4000)->no ip forwarding
N Chassis(rw-config-intf-vlan.0.4000)->exit
N Chassis(rw-config)->ip route 10.0.0.0/8 10.21.128.1 interface vlan.0.4000
N Chassis(rw-config)->ip route 125.20.0.0/16 125.20.10.1 interface vlan.0.4000
N 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:

```
N 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:

```
N Chassis(rw)->set ip route 192.122.173.50 vlan.0.50
```

Procedure 20-2 describes how to configure a non-forwarding IP traffic route.

Procedure 20-2 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**
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.

The ICMPv6 neighbor solicitation message is sent by a node to determine the link-layer address of a neighbor or to verify that a neighbor is still reachable via a cached link-layer address (Neighbor Unreachability Detection). Neighbor solicitations are also used for Duplicate Address Detection (DAD).

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:

```
N Chassis(su-config)->interface vlan 3050
N Chassis(su-config-intf-vlan.0.3050)->ipv6 address autoconfig
N 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:

```
N Chassis(su)->configure
N Chassis(su-config)->ipv6 neighbor 2001:11ac:fd34:3333:0:0:0:3 1111.1111.1111 interface vlan.0.2501
N 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
N Chassis(su-config)->
```

Configuring IPv6 Neighbor Discovery

Procedure 20-4 describes how to configure a static IPv6 neighbor discovery cache entry.

Procedure 20-3 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**] [**host**] [**interface** *interface*] [**verbose**] [**statistics**]
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 N-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 to configure a permanent static ARP entry.

Use the **set arp** command to configure a permanent static 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 static 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 an ARP request, but it may also be an ARP reply. ARP announcements are 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 N-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 disabled by default for the modification of pre-existing ARP table entries and 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 (the default)
- Configure the device to change the ARP table if the gratuitous ARP announcement is a reply
- Configure the device to change the ARP table 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. This command is disabled if **ip gratuitous-arp ignore** is configured (default setting). Otherwise it 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.

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

```
N Chassis(rw)->set arp 10.21.128.1 00:00:5e:00:01:01 temp
N Chassis(rw)->configure
N Chassis(rw-config)->arp timeout 2800
N Chassis(rw-config)->arp stale-entry-timeout 900
N 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
N Chassis(rw-config)->
```

The following example enables gratuitous ARP and ARP learning for ARP replies on VLAN 1:

```
N Chassis(rw)->configure
N Chassis(rw-config)->interface vlan 1
N Chassis(rw-config-intf-vlan.0.1)->ip gratuitous-arp reply
N Chassis(rw-config-intf-vlan.0.1)->ip gratuitous-arp-learning reply
N Chassis(rw-config-intf-vlan.0.1)->exit
N Chassis(rw-config)->
```

The following example enables proxy ARP for both default and local routes:

```
N Chassis(rw)->configure
N Chassis(rw-config)->ip prox
```

```
N Chassis(rw-config)->interface vlan 1
N Chassis(rw-config-intf-vlan.0.1)->ip proxy-arp default-route local
N Chassis(rw-config-intf-vlan.0.1)->exit
N Chassis(rw-config)->
```

Procedure 20-4 describes how to configure the ARP table.

Procedure 20-4 Configuring the ARP Table

Step	Task	Command(s)
1.	In configuration command mode, 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**}
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**}

Procedure 20-4 Configuring the ARP Table (continued)

Step	Task	Command(s)
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, set a MAC address on an interface.	**ip mac-address** *address*
9.	Optionally, in interface configuration command mode, remove the multicast restriction on ARP packets.	**ip multicast-arp-learning**

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 received on this interface.

Directed Broadcast Configuration Example

The following example enables directed broadcasts on VLAN 5:

```
N Chassis(rw)->configure
N Chassis(rw-config)->interface vlan 5
N Chassis(rw-config-intf-vlan.0.5)->ip directed-broadcast
N Chassis(rw-config-intf-vlan.0.5)->exit
N 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 N-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):

```
N 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:

```
N Chassis(rw-config)->ip forward-protocol udp domain
```

Procedure 20-5 describes how to configure IP broadcast.

Procedure 20-5 Configuring IP Broadcast

Step	Task	Command(s)
1.	In interface configuration command mode, enable IP directed broadcasts on an interface.	**ip directed-broadcast**

Procedure 20-5 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 interface configuration command mode, optionally insert the VPN option (82) into the relay agent DHCP packet.	**ip dhcp relay information option vpn**

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).

When forwarding the local UDP broadcasts from a VRF to a destination address on a different VRF, 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 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 server by the DHCP client.

When forwarding the local UDP broadcasts from a VRF to a destination address on a different VRF, 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 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 server by the DHCP client.

DHCP/BOOTP Relay Configuration Example

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:

```
N Chassis(rw)->configure
N Chassis(rw-config)->ip forward-protocol udp
N Chassis(rw-config)->interface vlan.0.1
N Chassis(rw-config-intf-vlan.0.1)->ip helper-address 192.168.1.255
N Chassis(rw-config-intf-vlan.0.1)->exit
N Chassis(rw-config)->interface vlan.0.2
N Chassis(rw-config-intf-vlan.0.2)->ip helper-address 192.24.1.255
```

Router Management and Information Display

Table 20-2 lists routing parameters and their default values.

Table 20-2 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
equal cost multipath algorithm	Specifies the algorithm used for selecting the next path used by the equal cost multipath feature.	hash threshold

Table 20-2 Default IP Routing Parameters (continued)

Parameter	Description	Default Value
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. IPv6 forwarding is currently not configurable on the N-Series platform.	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 20-3 describes how to manage IP configuration.

Table 20-3 Managing the Router

Task	Command
To clear this router configuration:	**clear router all**

Table 20-3 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 20-4 describes how to display IP configuration information and statistics.

Table 20-4 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 :

- 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 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 20-5 describes how to configure IP debug. All IP debug commands are accessed in configuration command mode.

Table 20-5 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 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 20-6 lists terms and definitions used in this IP routing configuration discussion.

Table 20-6 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.

Table 20-6 IP Routing Terms and Definitions (continued)

Term	Definition
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.
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.

21

Routing Information Protocol (RIP) Configuration

This document describes the RIP feature and its configuration on Enterasys N-Series devices.

Using RIP in Your Network

The N-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 N-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 N-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.

Automatic route summarization can be disabled. By default, RIP version 2 supports automatic route summarization, which summarizes subprefixes to the classful network boundary when crossing network boundaries. Disabling automatic route summarization enables CIDR, allowing RIP to advertise all subnets and host routing information on the N-Series device. To verify which routes are summarized for an interface, use the **show ip protocols** command.

Use the **no auto-summary** command in configuration command mode to disable automatic route summarization on this device. Use the **auto-summary** command to reenable automatic route summarization on this device.

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

```
N Chassis(rw-config)->router rip
N Chassis(rw-config-rip)->network 10.10.20.0 0.0.0.255
N Chassis(rw-config-rip)->network 10.10.50.0 0.0.0.255
N Chassis(rw-config-rip)->passive-interface vlan 10
N Chassis(rw-config-rip)->passive interface vlan 20
N Chassis(rw-config-rip)->timers basic 25 150 100
N Chassis(rw-config-rip)->redistribute ospf 16546
N Chassis(rw-config-rip)->exit
N 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

```
N Chassis(rw-config)->key chain md5key
N Chassis(rw-config-keychain)->key 3
N Chassis(rw-config-keychain-key)->key-string password
N Chassis>Router(config-keychain-key)->accept-lifetime 02:30:00 jul 30 2009
infinite
N Chassis(rw-config-keychain-key)->send-lifetime 02:30:00 jul 30 2009 infinite
N 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
!
N Chassis(rw-config-keychain-key)->exit
N Chassis(rw-config-keychain)->exit
N Chassis(rw-config)->interface vlan 5
```

```
N Chassis(rw-config-intf-vlan.0.5)->ip rip authentication mode md5
N Chassis(rw-config-intf-vlan.0.5)->ip rip authentication keychain md5key
N Chassis(rw-config-intf-vlan.0.5)->exit
N 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:

```
N Chassis(rw-config)->interface vlan 1
N 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 N-Series products.

Table 21-1 lists RIP parameters and their default values.

Table 21-1 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 21-1 describes how to configure RIP.

Procedure 21-1 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*

Procedure 21-1 Configuring RIP (continued)

Step	Task	Command(s)
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*
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 configuration command mode, disable automatic route summarization for this device.	**no auto-summary**
14.	Optionally, in RIP configuration command mode, specify an interface that will be prevented from transmitting RIP update packets.	**passive-interface vlan** *vlan-id*
15.	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 21-2 lists terms and definitions used in this RIP configuration discussion.

Table 21-2 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.
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.

22

Open Shortest Path First (OSPFv2) Configuration

This chapter provides the following information about configuring and monitoring OSPFv2 on Enterasys N-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 N-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, and static routes.

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 on a network basis
- Optionally configure areas including virtual-links, stub, and NSSA
- Optionally enable graceful-restart

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 N-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 22-9 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 22-1 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 22-9 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 22-1 on page 22-5:

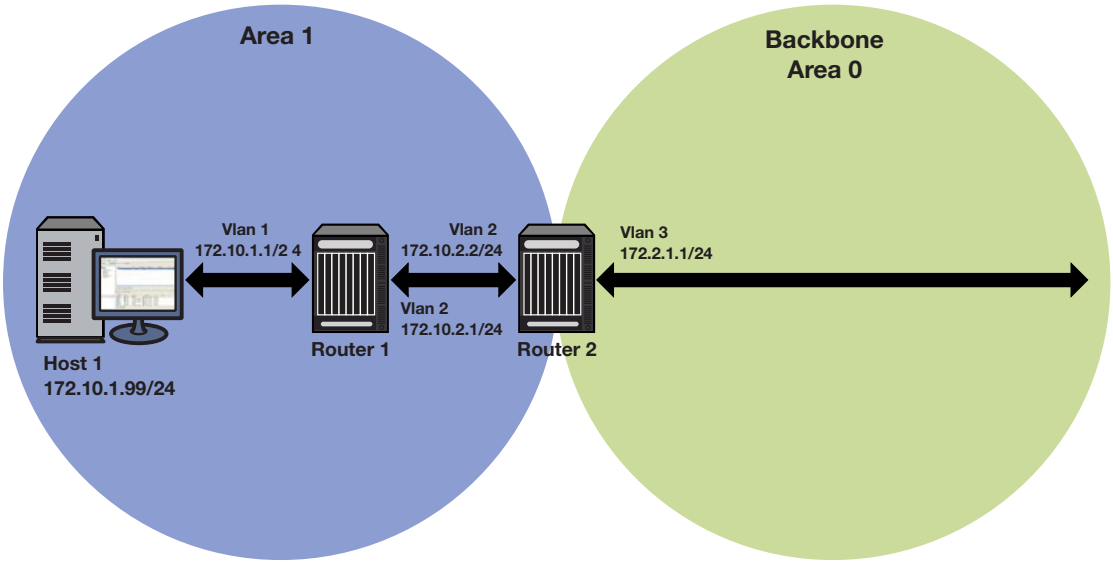
Router 1 CLI Input

```
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 22-1 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 22-2 on page 22-6:

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-vlan.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-vlan.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)->
```

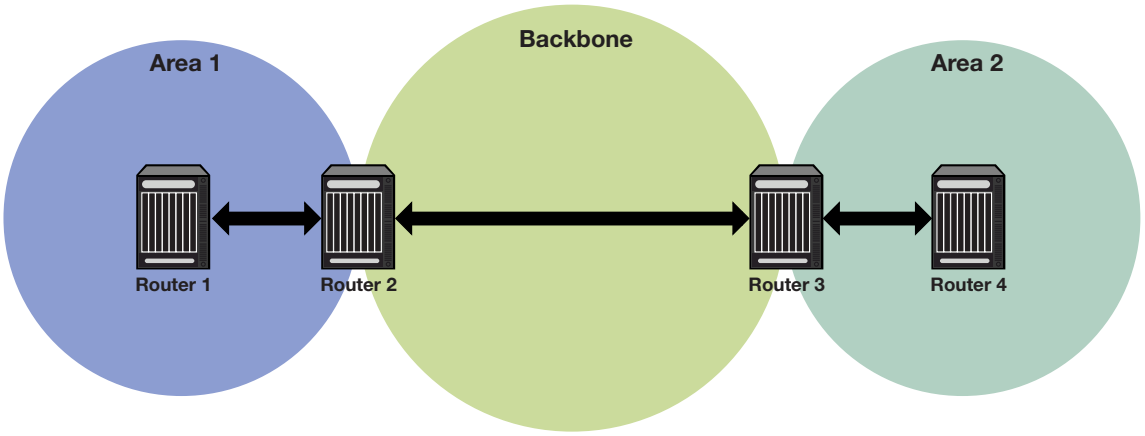
Router 3

```
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 22-2 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 22-7 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 22-6 for a router priority configuration example.

Figure 22-3 on page 22-8 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 22-3 on page 22-8:

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)->
```

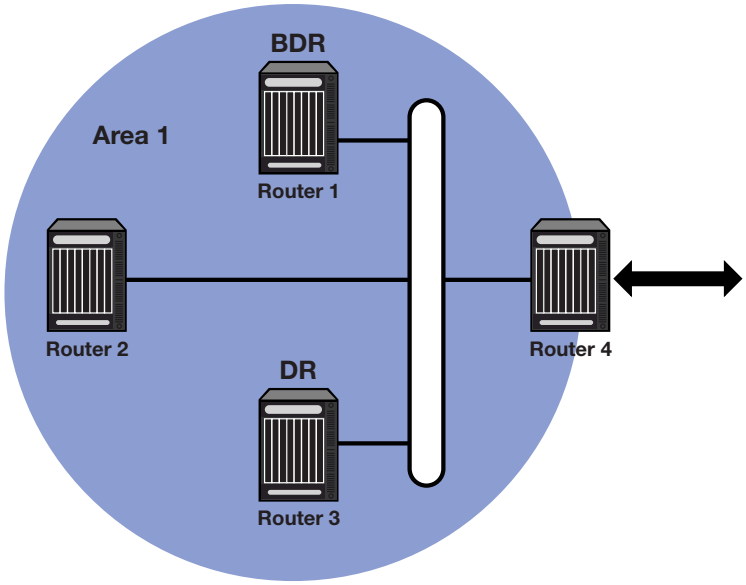
Router 3

```
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 22-3 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 N-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
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 22-3). 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 Djikstra 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 22-4 on page 22-10:

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)->
```

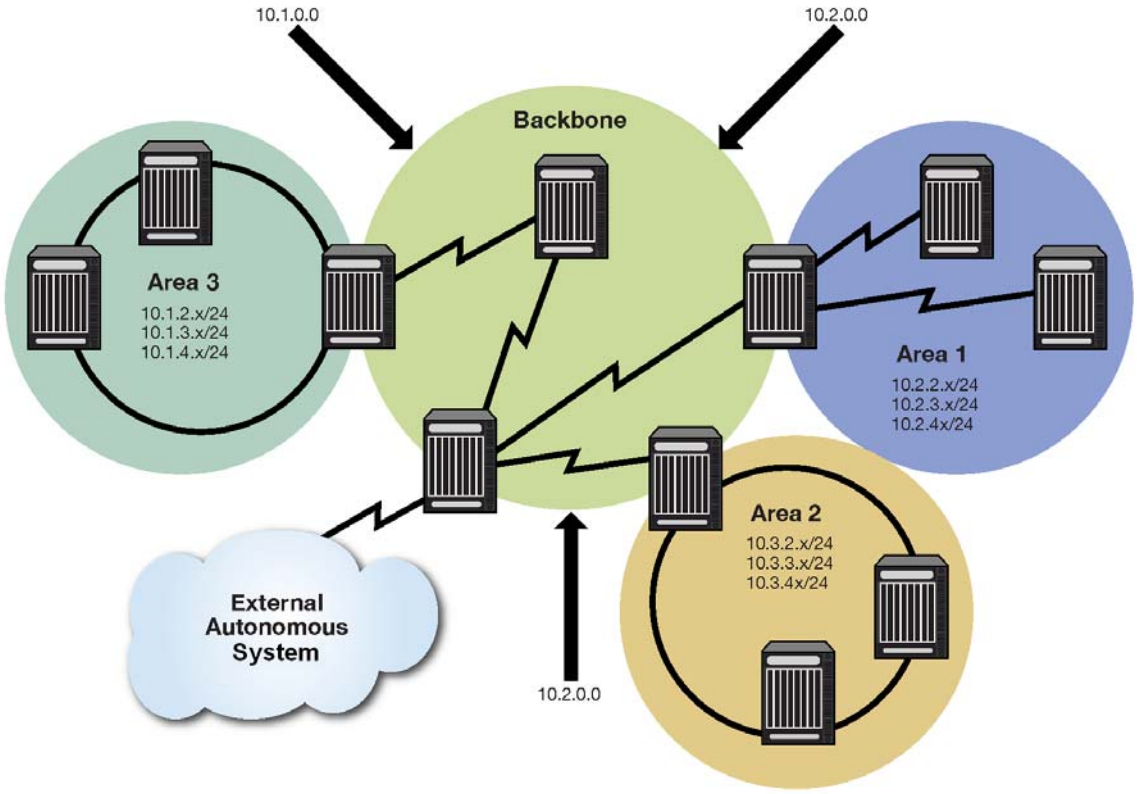
Area 2

```
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 22-4 OSPF Summarization Topology



Configuring Area Authentication

OSPF authentication can be configured at the area level, as well as at the interface level. OSPF authentication configured at the interface level overrides any authentication configuration at the area level. Area level OSPF authentication can be configured as either simple or message-digest

(MD5). See "Configuring OSPF with Authentication at the Interface" on page 22-19 for an OSPF authentication discussion.

Use the **area authentication** command in OSPF configuration command mode to configure area level OSPF authentication.

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 22-5 on page 22-12:

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
```

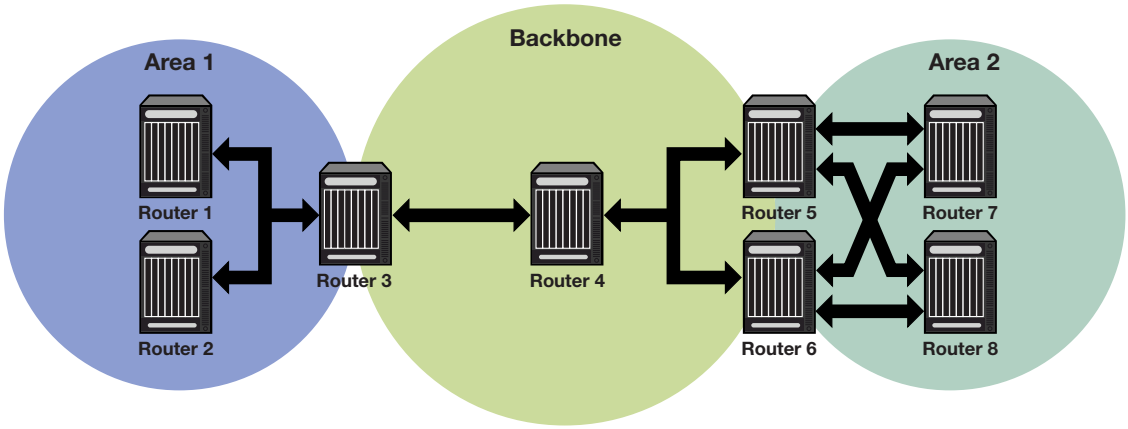
Router 7

```
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 22-5 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 22-6 on page 22-14:

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)

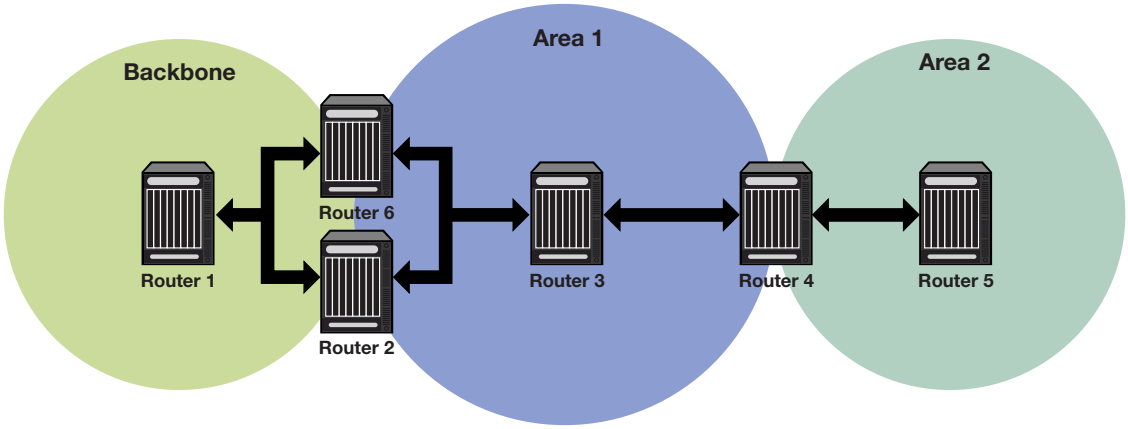
```
Router 2(rw)->configure
Router 2(rw-config)->interface vlan 1
Router 2(rw-config-intf-vlan.0.1)->ip address 11.1.1.2 255.255.255.252
Router 2(rw-config-intf-vlan.0.1)->no shutdown
Router 2(rw-config-intf-vlan.0.1)->exit
Router 2(rw-config)->interface vlan 2
Router 2(rw-config-intf-vlan.0.2)->ip address 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-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)->area 1 nssa transrole always
Router 4(rw-config-ospf-1)->redistribute static metric-type 1
Router 4(rw-config-ospf-1)->exit
```

Figure 22-6 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 22-11 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 22-7 on page 22-16 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 22-7 on page 22-16:

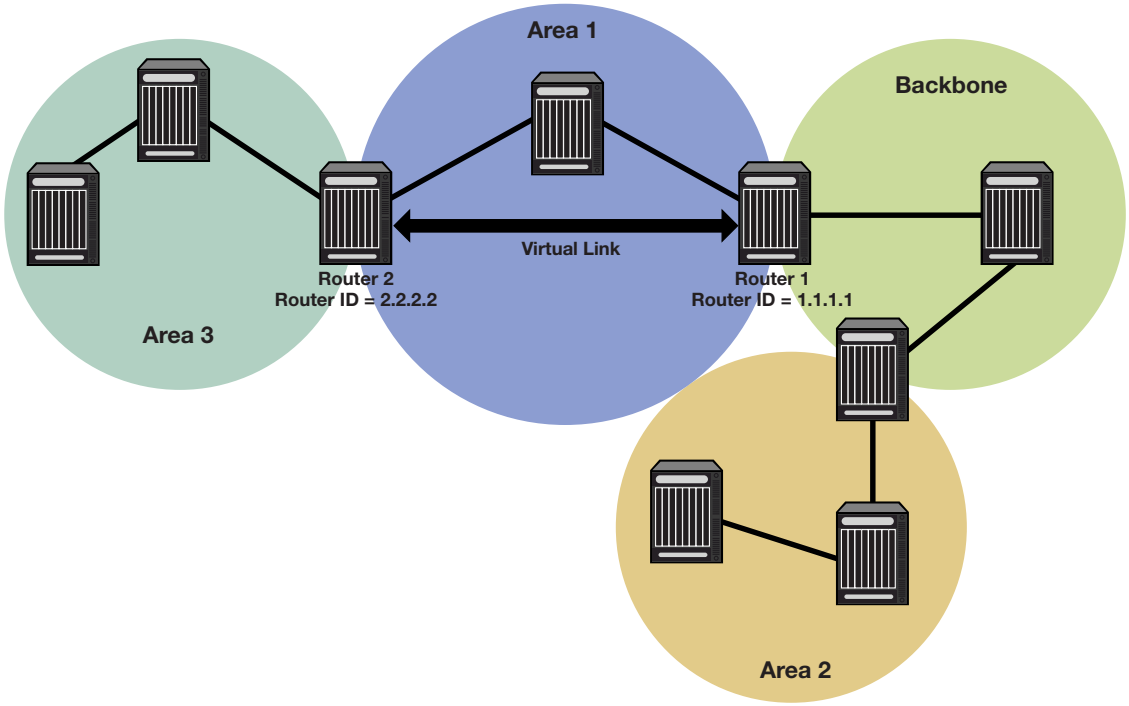
Router 1

```
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 22-7 Virtual Link Topology



Configuring Area Virtual-Link Authentication

An area virtual-link can be configured for either simple or MD5 authentication. See "Configuring Area Authentication" on page 22-10 for a discussion of these authentication types.

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 22-20 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 22-12 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 *Enterasys Matrix N-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.

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 N-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 22-8 Physical and Logical Single Router HA Failover Configuration

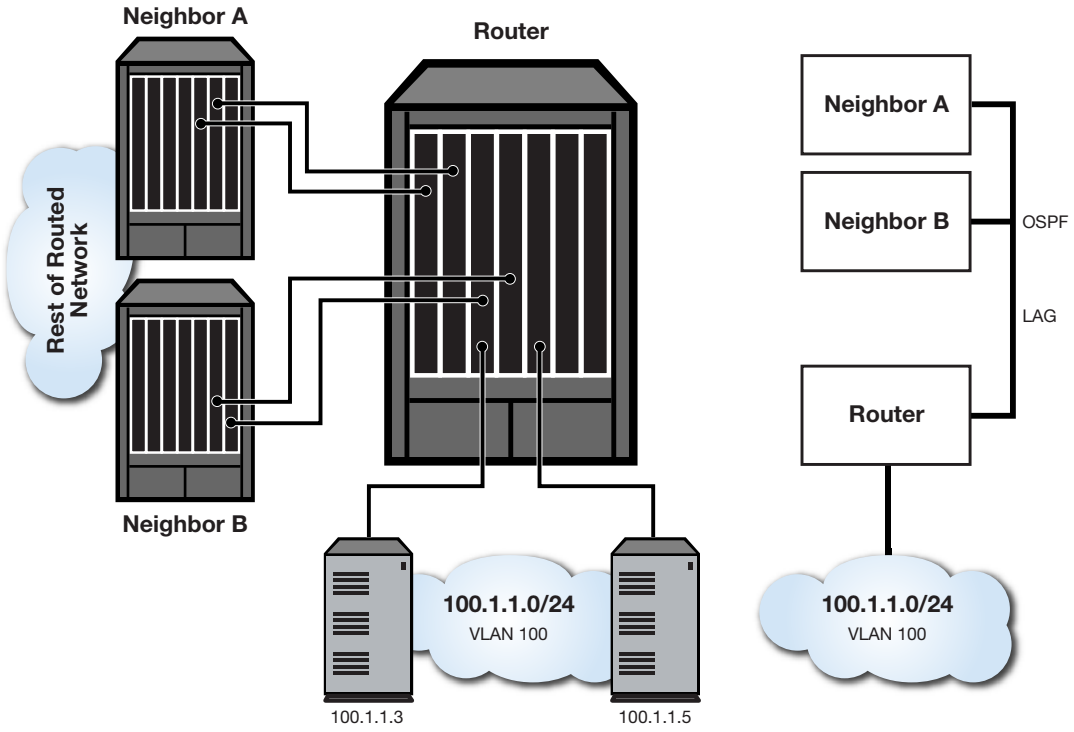


Figure 22-8 depicts the physical and logical configurations of the single router high availability failover mechanism. The neighbor to router lines display direct neighbor connections to the router enabled for OSPF graceful restart and members of LAGs common to the failing router. The server to router 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.

Use the **ip ospf cost** command in interface configuration command mode to specify the outbound cost of this interface.

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. See "Configuring Area Authentication" on page 22-10 for authentication configuration at the area 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.

OSPF authentication can be configured at the area level (see "Configuring Area Authentication" on page 22-10). OSPF authentication configured at the interface level overrides any authentication configuration at the area level.

N-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 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 N-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 22-14.

Configuring OSPF

This section provides details for the configuration of OSPF on N-Series platforms.

Important Notice

OSPF is an advanced routing feature that must be enabled with a license key. If you have purchased an advanced license key, and have enabled routing on the device, you must activate your license as described in the *Enterasys Matrix N-Series Configuration Guide* in order to enable the OSPF command set. If you wish to purchase an advanced routing license, contact Enterasys Networks Sales.

Default Settings

Table 22-1 lists OSPF parameters and their default values.

Table 22-1 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
transmit delay	Specifies the number of seconds it takes to transmit a link state update packet over this interface.	1 second

Table 22-1 Default OSPF Parameters (continued)

Parameter	Description	Default Value
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

Procedure 22-1 describes how to configure basic OSPF parameters. All commands in this procedure are entered in OSPF router configuration command mode, except where indicated.

Procedure 22-1 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 22-2 describes how to configure basic OSPF parameters.

Procedure 22-2 Configuring OSPF General Optional Parameters

Step	Task	Command(s)
1.	Optionally, change the OSPF router ID for this device.	**router-id** *ip-address*
2.	Optionally, configure the OSPF router neighbors for this router.	**neighbor** *ip-address* [**priority** *priority*]
3.	Optionally, change the SPF LSA thresholds for this router.	**spf lsa-thresholds** *num-start num-restart num-intra-full num-ia-ext-full*
4.	Optionally, change the SPF pause frequency to specify the number of units SPF calculation runs before pausing.	**spf pause-frequency** *units*
5.	Optionally, change the delay, in milliseconds, between the receipt of an update and the beginning of the SPF execution.	**timers spf** *spf-delay*
6.	Optionally, change the administrative distance for OSPF routes.	**distance** [**ospf** {**external** | **intra-area**}] *weight*
7.	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**]
8.	Optionally, enable authentication for a specified OSPF area.	**area** *area-id* **authentication** {**simple** | **message-digest**}
9.	Optionally, configure an area as a stub area.	**area** *area-id* **stub** [**no-summary**]
10.	Optionally, set the cost for the default route that is sent into a stub area by an ABR.	**area** *area-id* **default-cost** *cost*
11.	Optionally, configure an area as a not so stubby area.	**area** {*area-id* | *ip-address*} **nssa** [**no-summary**] [**transstabilityint** *seconds*] [**transrole always**]
12.	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*
13.	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*
14.	Optionally, enable passive OSPF on the specified interface.	**passive-interface** {*vlan-id* | *interface-name*}

Procedure 22-2 Configuring OSPF General Optional Parameters (continued)

Step	Task	Command(s)
15.	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*]
16.	Optionally, assign an OSPF route filter route-map to the OSPF distribute-list.	**distribute-list route-map** *name* **in**
17.	Optionally, enable the graceful-restart feature on this router.	**graceful-restart enable**
18.	Optionally, change the graceful-restart restart interval for this router.	**graceful-restart restart-interval** *interval*
19.	Optionally, in system command mode, reset the specified OSPF process ID or the OSPF process.	**clear ip ospf process** [*process-id*]
20.	Optionally, in global configuration command mode, enable OSPF protocol debugging output for the specified subsystem.	**debug ip ospf** {*subsystem*}
21.	Optionally, enable this OSPF router for RFC 1583 compatibility.	**rfc1583compatible**

Procedure 22-3 describes how to configure optional OSPF interface parameters. All commands in this procedure are entered in interface configuration command mode.

Procedure 22-3 Configuring OSPF Optional Interface Parameters

Step	Task	Command(s)
1.	Optionally, change the cost of sending an OSPF packet on this router interface.	**ip ospf cost** *cost*
2.	Optionally, change the OSPF priority value for this router interface.	**ip ospf priority** *number*
3.	Optionally, change the OSPF poll-interval value for this non-broadcast neighbor.	**ip ospf poll-interval** *seconds*
4.	Optionally, change the amount of time between retransmissions of LSAs for adjacencies that belong to this interface.	**ip ospf retransmit-interval** *seconds*
5.	Optionally, change the amount of time required to transmit a link state update packet on this interface.	**ip ospf transmit-delay** *seconds*
6.	Optionally, enable the ignore MTU advertisement feature for the neighbor of this interface.	**ip ospf ignore-mtu**
7.	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*
8.	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*}

Procedure 22-3 Configuring OSPF Optional Interface Parameters (continued)

Step	Task	Command(s)
9.	Optionally, assign a password on this interface to be used by neighboring routers using OSPF's simple password authentication.	**ip ospf authentication-key** *password*
10.	Optionally, enable OSPF MD5 authentication on this interface.	**ip ospf message-digest-key** *keyid* **md5** *key*
11.	Optionally, disable the graceful restart helper feature on this interface.	**ip ospf helper-disable**
12.	Optionally, specify the network type that this interface is connected to.	**ip ospf network** {**non-broadcast** | **broadcast** | **point-to-point** | **point-to-multipoint**}

Table 22-2 describes how to display OSPF configuration and statistics.

Table 22-2 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*]
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 OSPF virtual-links configuration information.	**show ip ospf virtual-links**

23

Network Address Translation (NAT) Configuration

This document provides the following information about configuring Network Address Translation (NAT) on the Enterasys N-Series platform.

Using Network Address Translation in Your Network

Network Address Translation (NAT) and 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 is a method by which IP addresses are mapped 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 is a method by which many private network addresses, along with each private address' associated TCP/UDP port, are translated 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.

In addition, the following features are also supported:

- Static NAT using singular IP addresses
- Dynamic NAT using NAT address pools
- FTPALG, DNS ALG, NAPT for ICMP Pings, and ICMP error fixups

Enterasys 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 4 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 configure 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 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.

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.

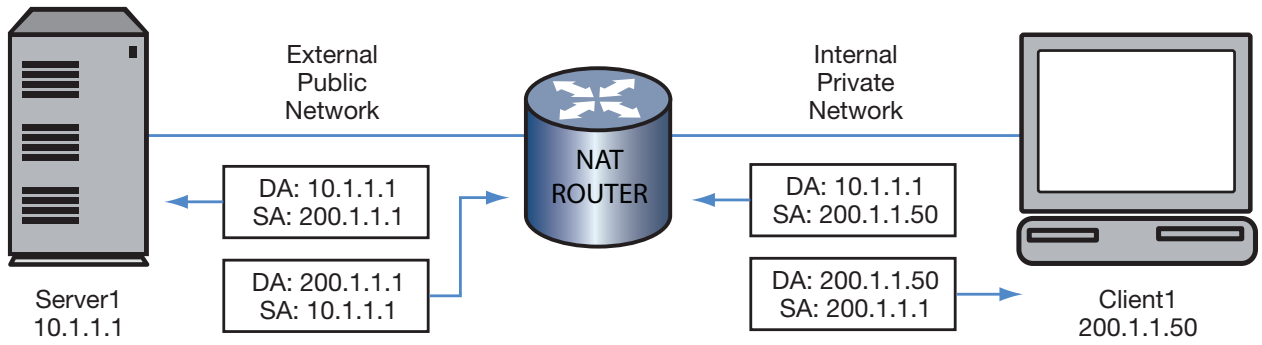
NAT Static Address Translation

Figure 23-1 on page 23-3 shows an example of a basic static NAT address translation. Client1 has a source address of 200.1.1.50 and a destination address of 10.1.1.1 (Server1). A static NAT translation is configured that maps Client1 to a publicly addressable outside address 200.1.1.1.

A packet arrives at the NAT router from Client1 with a source address of 200.1.1.50, but leaves the NAT router with a source address of 200.1.1.1. The IP packet's destination address is not changed, only the source IP address is. Server1 receives a packet from 200.1.1.1 and has no knowledge of the private address 200.1.1.50.

When Server1 responds to Client1, the packet arrives at the NAT router with Client1's translated address of 200.1.1.1 as the destination address, but leaves the NAT router with Client1's actual address of 200.1.1.50 as the destination address. Server1's response is delivered to IP address 200.1.1.50.

Figure 23-1 Basic NAT Static Address Translation



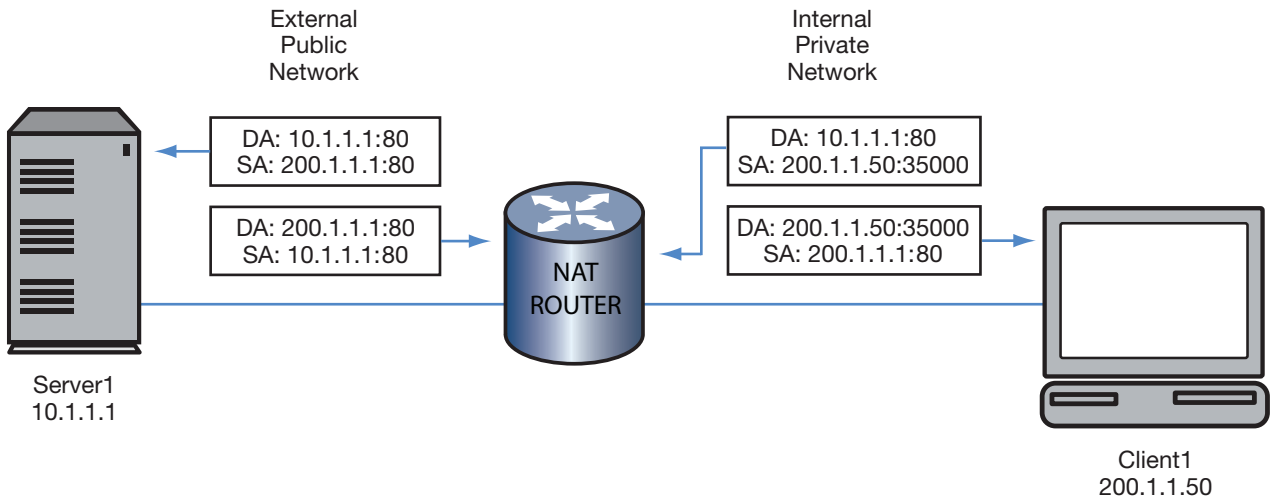
NAPT Static Address Translation

Figure 23-2 on page 23-4 shows an example of a basic static NAPT translation. Client1 is a device on a public network that wants to connect to the web service at 10.1.1.1 TCP port 80. The web service is in fact hosted by a Server1 with IP address 10.1.1.1 on the private network. A static NAT translation is configured that maps a public network IP address and TCP port (200.1.1.1 port 80) to a private network IP address and TCP port (10.1.1.1 port 80).

A packet arrives at the NAT router from Client1 with a source address of 200.1.1.50:35000, but leaves the NAT router with a source address of 200.1.1.1:80. In both cases the destination is for Server1's IP address of 10.1.1.1:80. From Server1's point of view, Client1's IP address is 200.1.1.1:80. Server1 doesn't know anything about its actual IP address of 200.1.1.50:35000.

When Server1 responds to Client1, its packet arrives at the NAT router with Client1's translated address of 200.1.1.1:80 as the destination address, but leaves the NAT router with Client1's actual address of 200.1.1.50:35000 as the destination address. Server1's response is delivered to IP address 200.1.1.50:35000.

Figure 23-2 Basic NAPT Static Address Translation



Dynamic Address Translations

Dynamic NAT is configured using a standard access-list, a NAT address pool, and a dynamic list rule.

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 list rule is used to configure dynamic NAT. This is an association of an access-list and a NAT pool. The access list specifies the source IP addresses that match the list rule and the pool specifies the NAT pool to assign global IP addresses from. If a list rule 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.

You can also specify the egress VLAN interface for which this list rule will be applied. Otherwise, the list rule applies to all interfaces.

NAT Dynamic Address Translation

Figure 23-3 on page 23-5 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 (200.1.1.50). A NAT pool must be configured with at least a single address range of publicly available IP addresses and assigned to this list rule. In this case the public IP address range is the single address of 200.1.1.1. This is a NAT (not NAPT) dynamic translation so we do not assign the overload option.

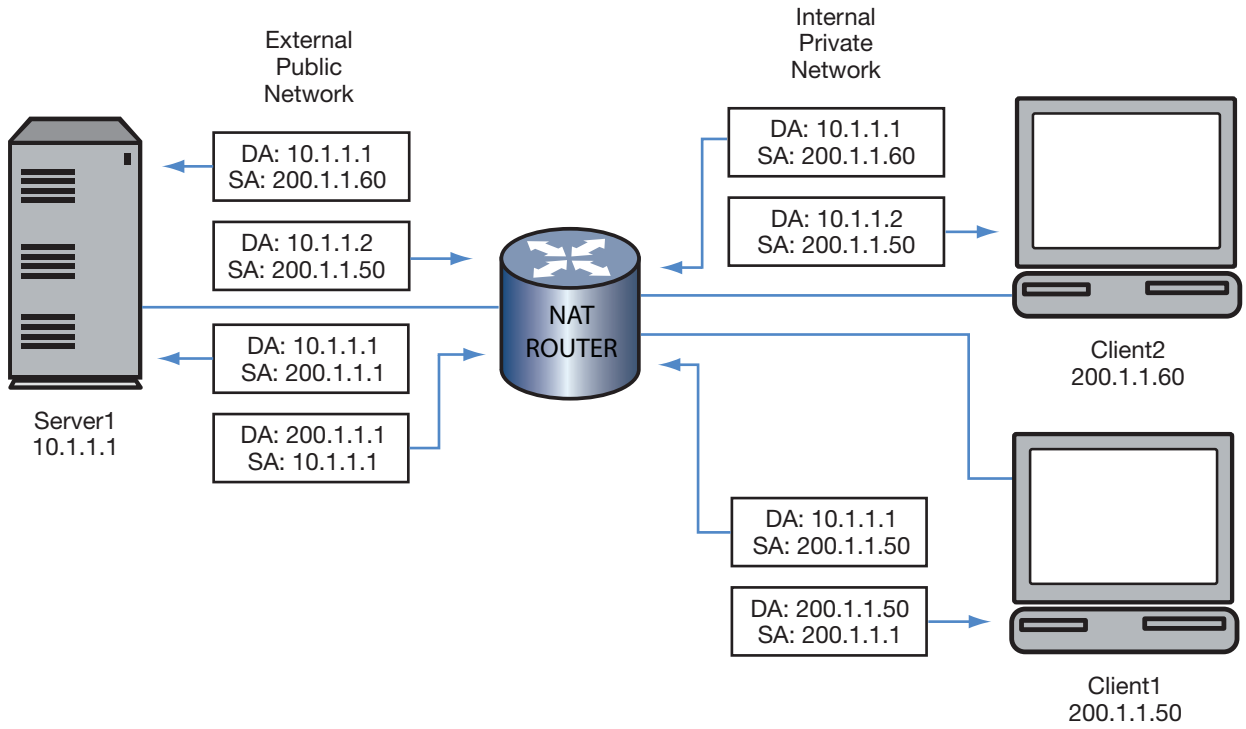
Client1 Walkthrough:

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 list rule. A dynamic binding is created and a global IP address is assigned to the binding 200.1.1.1. The packet is sent to Server1 with the destination IP unchanged and the source IP address changed to 200.1.1.1.

Server1 sends an IP packet back to 200.1.1.1. This packet matches the previously created dynamic binding. The packet is sent on to Client1 with the destination IP address changed from 200.1.1.1 to 200.1.1.50. The source IP address remains unchanged.

Figure 23-3 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 23-4 on page 23-6 shows an example of a basic dynamic NAPT address translation. The example shows network client Client1. The access-list assigned to this dynamic translation must contain permits for the Client1 IP address (200.1.1.50). A NAT pool can be configured with a single IP address for its range of publicly available IP addresses and assigned to this list rule. A single public IP address will be sufficient should multiple clients be configured because NAPT will use the available L4 port range of this address when assigning addresses for dynamic translation. In this case the public IP address range is for the single address 200.1.1.1. This is a NAPT dynamic translation so we must assign the overload option.

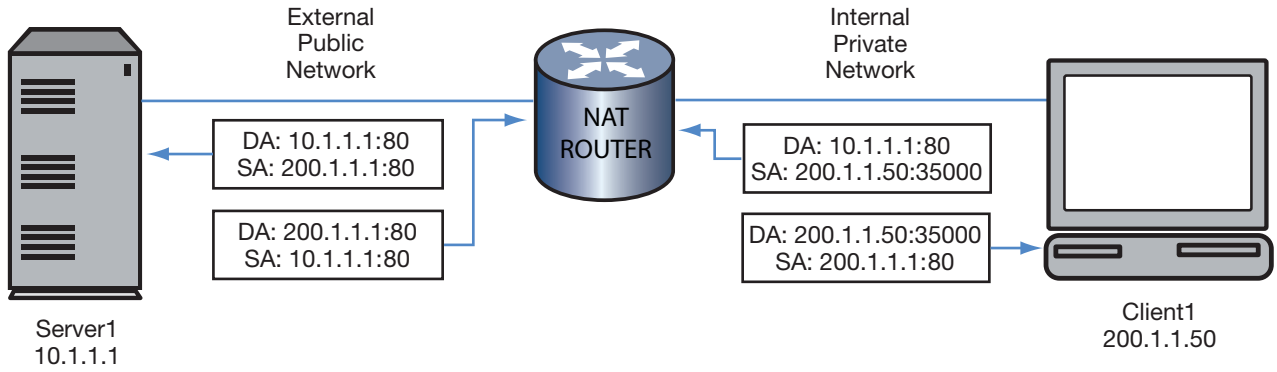
Client1 sends a TCP packet (source port 35000) to Server1 port 80, via the NAT router. The packet arrived on a VLAN configured as NAT inside and Server1 is accessible through a VLAN configured as NAT outside.

An access-list matching Client1's source IP address is configured to a NAT list rule. A dynamic binding is created and a global IP address is assigned to the binding 200.1.1.1. Since the list rule is overloaded the NAT pool is checked to see if Client1's original source port (35000) is in use for the global address 200.1.1.1. If this port is already in use by some other binding then 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 200.1.1.1 and the TCP source port changed to 80.

When Server1 responds to Client1, its packet arrives at the NAT router with Client1's translated address of 200.1.1.1:80 as the destination address, but leaves the NAT router with Client1's actual address of 200.1.1.50:35000 as the destination address. Server1's response is delivered to IP address 200.1.1.50:35000.

Figure 23-4 Basic NAPT Dynamic Inside Address Translation



NAT Translation Protocol Rules

Translation protocol rules are provided as a dynamic means of setting NAT binding idle time out and "one-shot" setting, 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 send 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 nat translation protocol** in global configuration command mode to create 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

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 the translation of the IP address embedded in the data portion of 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 when these 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 fixing up the DNS packet with the appropriate NAT translations.

NAT inspection of DNS packets is disabled by default.

Use the **ip 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. If the interface is part of the external public network, it should be enabled as an outside interface.

Configuring NAT

This section provides details for the configuration of NAT on the N-Series products.

Table 23-1 lists NAT parameters and their default values.

Table 23-1 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.	240 seconds

Table 23-2 lists NAT resource limits.

Table 23-2 NAT Resource Limits

Resource	N-Series
Global Bindings	32768
IP Addresses	1000
Pools	10
Port Mapped Addresses	10
Static Rules	500

Configuring Traditional NAT Static Inside Address Translation

Procedure 23-1 describes how to configure traditional NAT for a static configuration.

Procedure 23-1 Traditional NAT Static Configuration

Step	Task	Command(s)
1.	Enable NAT, in interface configuration mode, on all interfaces on which translation takes place for both the internal and external networks.	**ip nat** {**inside** | **outside**}

Procedure 23-1 Traditional NAT Static Configuration

Step	Task	Command(s)
2.	Enable, in global configuration mode, any static NAT translations of 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*
3.	Enable, in global configuration mode, any static NAPT translations of 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 23-2 describes how to configure traditional NAT for a dynamic configuration.

Procedure 23-2 Traditional NAT Dynamic Configuration

Step	Task	Command(s)
1.	Enable, in interface configuration mode, NAT on all interfaces on which translation takes place for both the internal and external networks.	**ip nat** {**inside** | **outside**}
2.	Define, in global configuration mode, an access-list of permits for all inside addresses to be used by this dynamic translation.	**ip access-list** *list-number* {**deny** | **permit**} *source*
3.	Define, in global configuration mode, a NAT address pool for all 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*}
4.	Enable, in global configuration mode, dynamic translation of inside source addresses. Specify the overload option for NAPT translations.	**ip nat inside source list** *access-list* **pool** *pool-name* [**overload** | **interface vlan** *vlan-id* [**overload**]]

Managing a Traditional NAT Configuration

Table 23-3 describes how to manage traditional NAT configurations.

Table 23-3 Managing a Traditional NAT Configuration

Task	Command(s)
Optionally, in global configuration mode, specify a non-default NAT FTP control port.	**ip nat ftp-control-port** *port-number*
Optionally, in global configuration mode, set the maximum number of translation entries.	**ip nat translation max-entries** *number*
Optionally, in global configuration mode, set NAT translation timeout values.	**ip nat translation** {**timeout** | **udp-timeout** | **tcp-timeout** | **icmp-timeout** | **dns-timeout** | **ftp-timeout**} *seconds*
Optionally, in global configuration mode, create a NAT translation protocol rule.	**ip nat translation protocol** *protocol* **timeout** [*seconds*] [**one-shot**]

Table 23-3 Managing a Traditional NAT Configuration

Task	Command(s)
Optionally, in global configuration mode, enable logging to log a message each NAT binding is created or deleted.	**ip nat log translations**
Optionally, in global configuration mode, enable NAT inspection and fixup of DNS packets forwarded by the NAT process.	**ip nat inspect dns**
Optionally, in global configuration mode, clear NAT bindings.	**clear ip nat bindings** {**all** | **pool** *pool* | **id** *id* | **match** {*protocol* | ***** | **icmp** {*sip* | *****} {*dip* | *****} | **tcp** {*sip* | ***** *port* | **} {*dip* | ***** *port* | *****} | **udp** {*sip* | *****} {*dip* | *****}} [**detail**]
Optionally, in global configuration mode, clear NAT statistics.	**clear ip nat statistics**

Displaying NAT Statistics

Table 23-4 describes how to display NAT statistics.

Table 23-4 Displaying NAT Statistics

Task	Command(s)
Display NAT bindings.	**show ip nat bindings** [**id** *binding-id*] [**pool** *pool* [**detail**]] [**match** *protocol* {*sip dip* [**detail**] | *****}] [**summary**]
Display NAT information.	**show ip nat info**
Display NAT lists matching rules.	**show ip nat lists** [*list-name*] [**detail**]
Display NAT pools.	**show ip nat pools** [*name*] [*detail*]
Display NAT static matching rules.	**show ip nat statics** [**detail**]
Display NAT statistics.	**show ip nat statistics**

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. These parameters should only be modified to assure availability to functionalities that share these resources such as TWCB and LSNAT. It is recommended that you consult with Enterasys Customer Support before modifying these parameter values.

We will also assume that the FTP control port will use the default value.

NAT Static Configuration Example

This example steps you through a NAT static configuration for both NAT and NAPT translation methods. See Figure 23-5 on page 23-11 for a depiction of the NAT static configuration example setup.

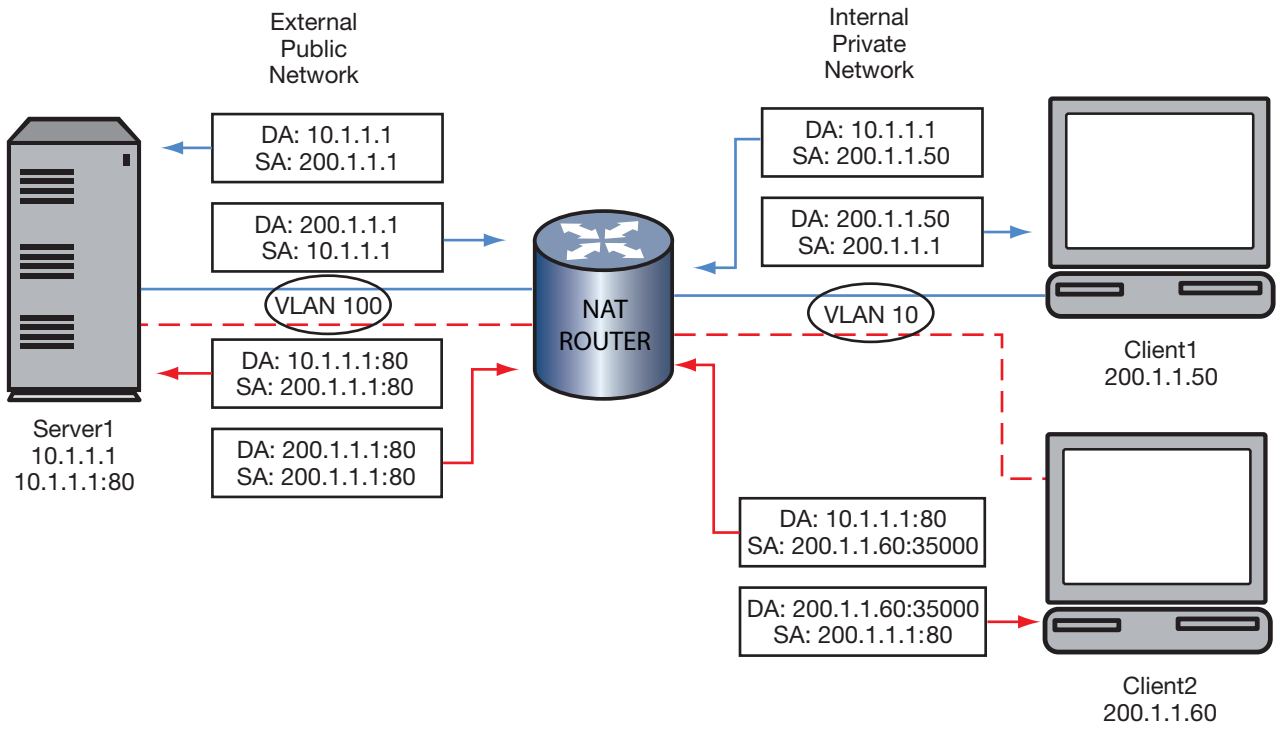
Our static NAT configuration example configures two clients: Client1 with NAT translation and Client2 with NAPT translation. Both clients are on the internal private network VLAN 10 interface

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 23-5 NAT Static Configuration Example



Enable NAT Inside and Outside Interfaces

Enable NAT inside interface:

```
N Chassis(rw)->configure
N Chassis(rw-config)->interface vlan 10
N Chassis(rw-config-intf-vlan.0.10)->ip nat inside
N Chassis(rw-config-intf-vlan.0.10)->exit
N Chassis(rw-config)->
```

Enable NAT outside interface:

```
N Chassis(rw-config)->interface vlan 100
N Chassis(rw-config-intf-vlan.0.100)->ip nat outside
N Chassis(rw-config-intf-vlan.0.100)->exit
N Chassis(rw-config)->
```

Enable Static Translation of Inside Source Addresses

Enable the NAT static translation of the inside source address:

```
N 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:

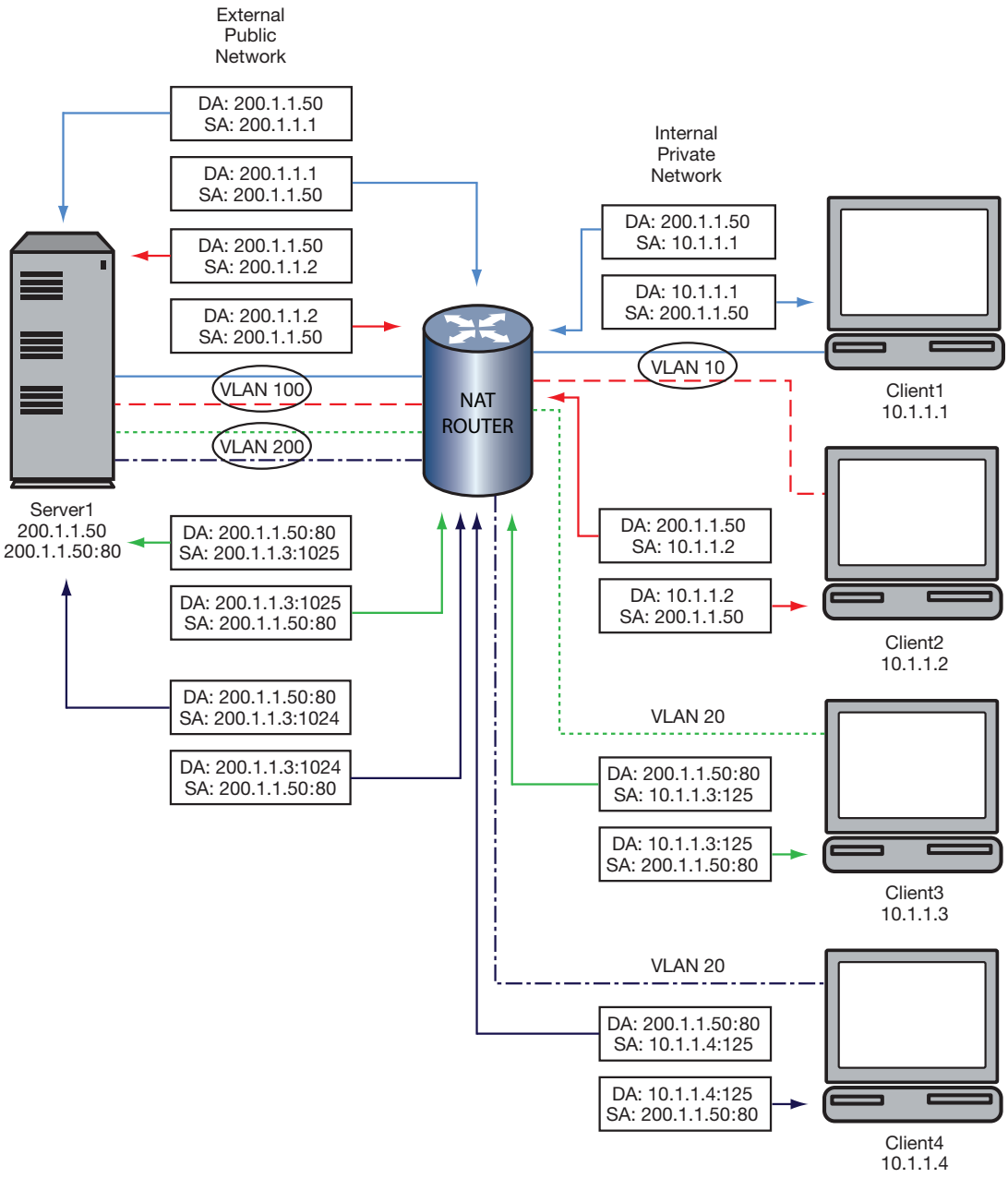
```
N Chassis(rw-config)->ip nat inside source static tcp 200.1.1.60:35000
200.1.1.2:80
```

NAT Dynamic Configuration Example

This example steps you through a NAT Dynamic Configuration for both NAT and NAPT translation methods. See Figure 23-6 on page 23-13 for a depiction of the example setup.

Our dynamic NAT configuration example configures four clients: Client1 and Client2 with NAT translation and Client3 and Client4 with NAPT translation. The two NAT clients are on the internal private network VLAN 10 interface and communicate with Server1 over the external public network VLAN 100 interface. The two NAPT clients are on the internal private network VLAN 20 and communicate with Server1 over the external public network VLAN 200 interface. NAT is enabled on VLAN 10 and VLAN 20 as inside interfaces. NAT is enabled on VLAN 100 and VLAN 200 as outside interfaces. These are the only VLANs over which translation occurs for the dynamic portion of this configuration example.

Figure 23-6 NAT Dynamic Configuration Example



To configure Client1 and Client2 for dynamic NAT translation on the NAT router, we define access-list **1** to permit the local IP addresses 10.1.1.1 and 10.1.1.2. We then configure the NAT translation NAT pool **natpool** with the global address range of 200.1.1.1 to 200.1.1.2. We then enable dynamic translation of inside addresses associating access-list **1** with the NAT pool **natpool**.

To configure Client3 and Client4 for dynamic NAPT translation on the NAT router, we define access-list **2** to permit the local IP addresses 10.1.1.3 and 10.1.1.4. We then configure NAT pool **dynamicpool** with a global range of 200.1.1.3 to 200.1.1.3. We then enable dynamic translation of inside addresses for overload associating access-list **2** with the NAT pool **naptpool**.

Enable NAT Inside and Outside Interfaces

Enable NAT inside interface:

```
N Chassis(rw)->configure
N Chassis(rw-config)->interface vlan 10
N Chassis(rw-config-intf-vlan.0.10)->ip nat inside
N Chassis(rw-config-intf-vlan.0.10)->exit
N Chassis(rw-config)->interface vlan 20
N Chassis(rw-config-intf-vlan.0.20)->ip nat inside
N Chassis(rw-config-intf-vlan.0.20)->exit
N Chassis(rw-config)->
```

Enable NAT outside interface:

```
N Chassis(rw-config)->interface vlan 100
N Chassis(rw-config-intf-vlan.0.100)->ip nat outside
N Chassis(rw-config-intf-vlan.0.100)->exit
N Chassis(rw-config)->interface vlan 200
N Chassis(rw-config-intf-vlan.0.200)->ip nat outside
N Chassis(rw-config-intf-vlan.0.200)->exit
N Chassis(rw-config)->
```

Define Inside Address Access-Lists

Define inside address access-list 1 for NAT clients:

```
N Chassis(rw-config)->access-list standard 1 permit host 10.1.1.1
N Chassis(rw-config)->access-list standard 1 permit host 10.1.1.2
N Chassis(rw-config)->
```

Define inside address access-list 2 for NAPT clients:

```
N Chassis(rw-config)->access-list standard 2 permit host 10.1.1.3
N Chassis(rw-config)->access-list standard 2 permit host 10.1.1.4
N Chassis(rw-config)->
```

Define the NAT Pools for Global Addresses

Define the NAT Pool for the NAT clients:

```
N Chassis(rw-config)->ip nat pool natpool 200.1.1.1 200.1.1.2 netmask
255.255.255.0
```

Define the NAT Pool for the NAPT clients:

```
N Chassis(rw-config)->ip nat pool naptpool 200.1.1.3 200.1.1.3 netmask
255.255.255.0
```

Enable Dynamic Translation of Inside Source Addresses

Enable the NAT dynamic translation of the inside source address:

```
N Chassis(rw-config)->ip nat inside source list 1 pool natpool
```

Enable the NAPT dynamic translation of the inside source address:

```
N Chassis(rw-config)->ip nat inside source list 2 pool naptpool overload
```

Terms and Definitions

Table 23-5 lists terms and definitions used in this NAT configuration discussion.

Table 23-5 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.
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.
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.

24

Load Sharing Network Address Translation (LSNAT) Configuration

This document provides the following information about configuring LSNAT on the Enterasys N-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, or FTP on port 21. 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

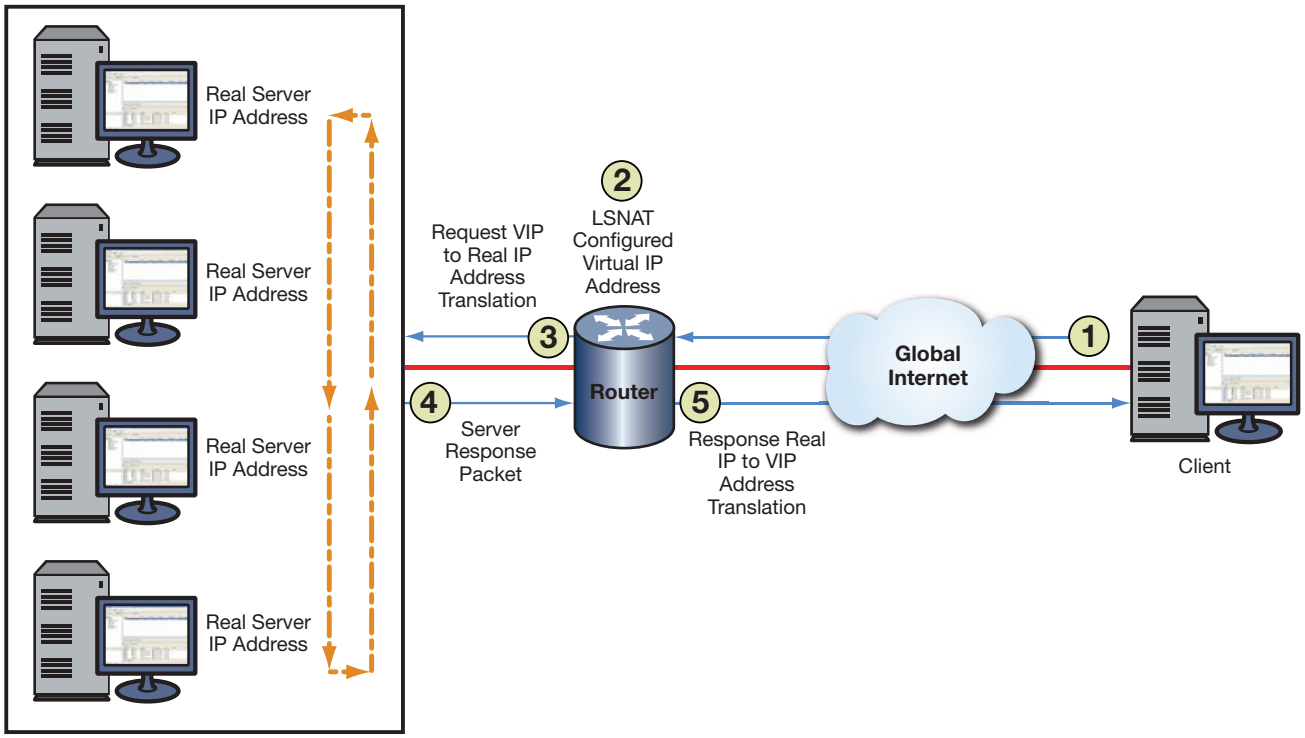
Figure 24-1 on page 2 provides the following example of an LSNAT deployment:

1. A request for service is sent by the client to a virtual server.
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. 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.

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 24-1 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. 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 N-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)
   - 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 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 optionally clearing load balancing connections or statistics

LSNAT Overview

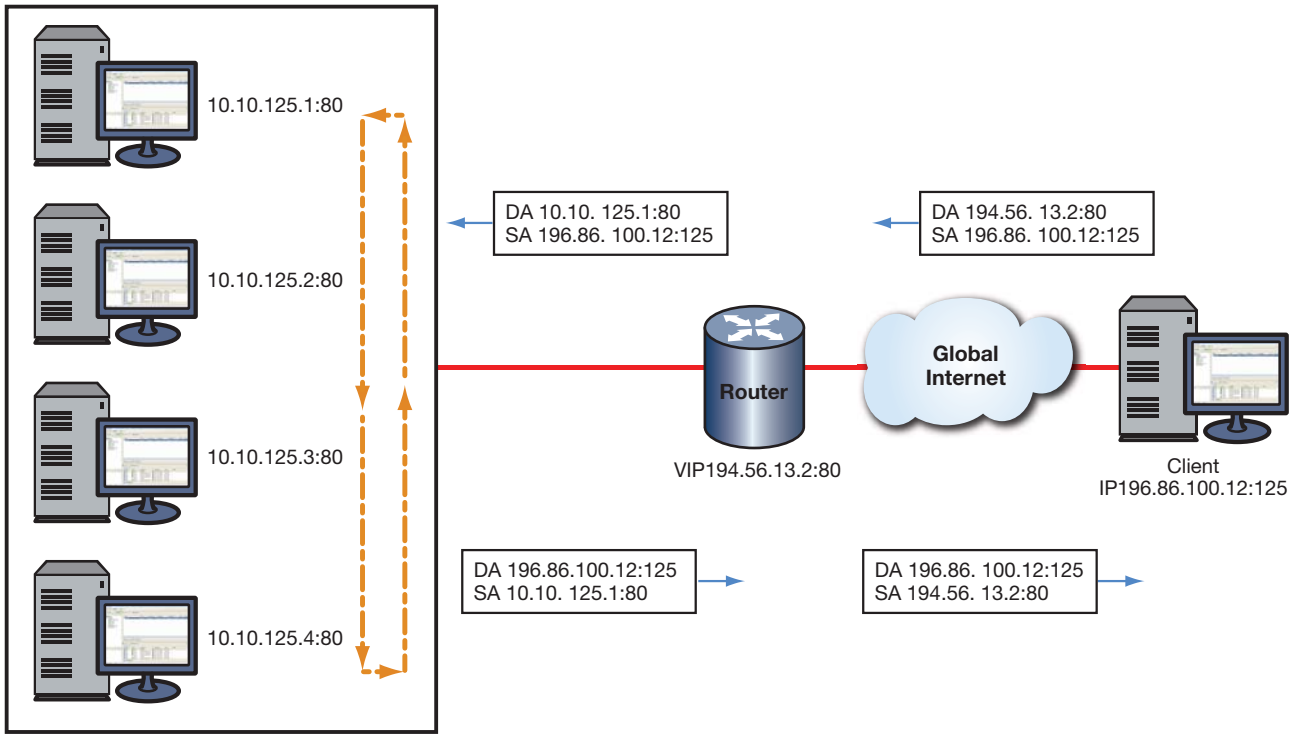
This section provides an overview of the LSNAT components.

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.

Figure 24-2 on page 24-4 presents an LSNAT packet flow. 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 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.

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 24-2 LSNAT Packet Flow



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 7, **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 7-5 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

```
N Chassis(su)->configure
N Chassis(su-config)->probe TCP-HTTP tcp
N Chassis(su-config-probe)->faildetect interval 5
N Chassis(su-config-probe)->passdetect interval 5
N Chassis(su-config-probe)->acv request “GET / HTTP/1.1\\r\\nHost:
2.0.0.5\\r\\n\\r\\n”
N Chassis(su-config-probe)->acv reply “HTTP/1.1 200 OK\\r\\n”
N Chassis(su-config-probe)->inservice
N 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
N Chassis(su-config-probe)->exit
N Chassis(su-config)->ip slb serverfarm myproductHTTP
N Chassis(su-config-slb-sfarm)->real 10.1.2.3 port 80
N Chassis(su-config-slb-real)->faildetect probe one TCP-HTTP
N Chassis(su-config-slb-real)->inservice
N 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, and configure the VIP. 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. 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, with real servers and virtual servers configured and “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, configured within global configuration mode with the **ip slb real-server access client** command, 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.

The second mechanism, configured in global configuration mode with the **ip slb real-server access unrestricted** command, allows all clients to directly access all services provided by real servers, except for those services configured for server load balancing.

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 23, **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-NATed. 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 NATed, this allows the real servers to be located anywhere in a network, since the packets from router to real-server will be source NATed with an IP address owned by the router itself.

Use the **source nat pool** command to specify a NAT pool to use for source NATing. The NAT pool is used in an overload mode.

The FTP Control Port

The FTP port assignment defaults to port 21. You can globally assign a non-standard FTP 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 N-Series products.

Important Notice

LSNAT is an advanced routing feature that must be enabled with a license key. If you have purchased an advanced license key, and have enabled routing on the device, you must activate your license, as described in "License Overview" on page 6-9, in order to enable the LSNAT command set. If you wish to purchase an advanced routing license, contact Enterasys Networks Sales.

Table 24-1 lists LSNAT parameters and their default values.

Table 24-1 Default LSNAT Parameters

Parameter	Description	Default Value
Port Number (FTP)	The port number for the FTP control port for all virtual servers.	21
Predictor	The load balancing algorithm for this server farm.	Round Robin

Table 24-1 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
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 24-2 lists LSNAT resource limits.

Table 24-2 LSNAT Resource Limits

Resource	N-Series
Bindings	32768
Reals	500
Server Farms	50
Sticky Entries	2000
VIP Addresses	1000
Virtual Servers	50

Configuring an LSNAT Server Farm

Procedure 24-1 describes how to configure an LSNAT server farm.

Procedure 24-1 LSNAT Server Farm Configuration

Step	Task	Command(s)
1.	In global router configuration command mode, specify a name for this server farm.	**ip slb serverfarm** *serverfarmname*

Procedure 24-1 LSNAT Server Farm Configuration

Step	Task	Command(s)
2.	In SLB server farm configuration command mode, specify the load balancing algorithm for this server farm.	**predictor** [**roundrobin** | **leastconns**]
3.	In SLB server farm configuration command mode, enable the this server farm. The default setting for server farms is inservice.	**inservice**

Configuring an LSNAT Real Server

Procedure 24-2 describes how to configure an LSNAT real server.

Procedure 24-2 Configuring an LSNAT Real Server

Step	Task	Command(s)
1.	In SLB server farm configuration command mode, configure the real server members for this server farm and enter real server configuration command mode.	**real** *ip-address* [**port** *number*]
2.	In SLB real 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 SLB real server configuration command mode, optionally specify whether the currently configured probes are active or inactive for this real server.	**faildetect type** {**none** | **probe**}
4.	In SLB real server configuration command mode, optionally reset fail detection configuration to the factory default settings for this real server.	**faildetect reset**
5.	In SLB real server configuration command mode, optionally limit the maximum number of active connections for this real server.	**maxconns** *maximum-number*
6.	In SLB real server configuration command mode, optionally configure a weight for this real server to be used by the round robin load balancing algorithm.	**weight** *weight-number*
7.	In SLB real server configuration command mode, enable each real server for service.	**inservice**

Configuring an LSNAT Virtual Server

Procedure 24-3 describes how to configure an LSNAT virtual server.

Procedure 24-3 Configuring an LSNAT Virtual Server

Step	Task	Command(s)
1.	In global router configuration command mode, specify a name for this virtual server.	**ip slb vserver** *vserver-name*
2.	In SLB virtual server configuration command mode, optionally specify a match source port to virtual server binding behavior.	**binding match source-port** {**any** | **exact**}
3.	In SLB virtual server configuration command mode, associate this virtual server with a server farm.	**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.	**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.	**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.	**source nat pool** *pool*
7.	In SLB virtual server configuration command mode, enable the virtual server for service	**inservice**
8.	In SLB virtual server configuration command mode, optionally configure this 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*
9.	In SLB virtual server configuration command mode, optionally restrict access to this virtual server to configured clients.	**client** [*ip-address network-mask*]
10.	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**
11.	In SLB virtual server configuration command mode, optionally configure the stickiness type.	**sticky type** [**sip** | **sip dip-dport**]
12.	In SLB virtual server configuration command mode optionally configure the sticky entry timeout value for this virtual server.	**sticky timeout** *timeperiod*

Procedure 24-3 Configuring an LSNAT Virtual Server (continued)

Step	Task	Command(s)
13.	In global configuration command mode, optionally allow specific clients to access the load balancing real servers in a particular LSNAT server farm without address translation.	**ip slb real-server access client** {*ip-address mask* | *ip-prefix/length*}
14.	In router command mode, optionally clear sticky entries or remove bindings.	**clear ip slb** {**sticky** | **bindings**} {**all** | **id** *id* | **match** {*sip* | *****} {*sport* | *****} {*dip* | *****} {*dport* | *****}}

Configuring Global Settings

Table 24-3 describes how to configure LSNAT global settings.

Table 24-3 Configuring LSNAT Global Settings

Task	Command(s)
In global configuration command mode, optionally specify a non-default FTP control port for all virtual servers. (Default = 21).	**ip slb ftpctrlport** *port-number*
In global configuration command mode, optionally specify a non-default TFTP control port for all virtual servers. (Default = 69).	**ip slb tftpctrlport** *port-number*
In global configuration command mode, optionally allow all clients to directly access all services provided by real servers, except for those services configured for server load balancing.	**ip slb real-server access unrestricted**
In global configuration command mode, allows specific client networks to access the real servers without address translation.	**ip slb real-server access client** *client-ip-address* {*ip-prefix* | *mask*}

Displaying LSNAT Configuration Information and Statistics

Table 24-4 describes how to display LSNAT configuration information and statistics.

Table 24-4 Displaying LSNAT Configurations and Statistics

Task	Command(s)
Display the specified or all server farm configurations	**show ip slb serverfarms** [**detail** | *serverfarmname*]
Display all real server configurations for this system or those for the specified server farm.	**show ip slb reals** [**detail** | **serverfarm** *serverfarmname* [**detail**]]
Display all or the specified virtual servers for this system.	**show ip slb vservers** [**detail** | *virtserver-name*]
Display server load balancing statistics.	**show ip slb statistics**

Table 24-4 Displaying LSNAT Configurations and Statistics (continued)

Task	Command(s)
Display SLB bindings.	**show ip slb bindings** {**match** [*ip-address* | ***] | **id** *id* | **summary**}
Display LSNAT configuration information.	**show ip slb info**
Display active server load balancing sticky mode connections.	**show ip slb sticky** {**match** *sip port dip port* | **id** *id* | **summary**}
Display sticky statistics.	**show ip slb statistics-sticky**

LSNAT Configuration Example

This section provides an enterprise LSNAT configuration example that includes five server farms. These server farms can be logically thought of as either product-based or enterprise internal server farms. The product-based server farms are accessible to the general public. The enterprise internal server farms are accessible only to enterprise employees. The myproduct HTTP and FTP server farms provide the product-based services. The myinternal HTTP, FTP, and SMTP server farms provide enterprise internal services.

Product-Based and Enterprise Internal Domains

The HTTP and FTP domains providing public access to the product-based server farms are:

- www.myproduct.com
- ftp.myproduct.com

The HTTP, FTP, and SMTP domains providing employee access to the enterprise internal server farms are:

- www.myinternal.com
- ftp.myinternal.com
- smtp.myinternal.com

Server Farms

For both the public product-based and enterprise internal server farms, the enterprise IT clients will have direct access to the servers without any address translation required. All other clients that have access rights to these server farms will be address translated.

Product-Based HTTP Server Farm

The product-based HTTP server farm, real server and virtual server configuration will:

- Handle HTTP requests from the general public using the www.myproduct.com domain.
- Load balance HTTP services across the three real servers associated with www.myproduct.com, using the weighted round robin selection process with a ratio of 3:2:2. The weighted round robin selection process takes into account the resource differences between the three servers.
- Configure and apply the TCP-HTTP probe that verifies port 80 and uses Application Content Verification TCP fail detection.

- Use the VIP 194.56.12.2 port 80.

Product-Based FTP Server Farm

The product-based FTP server farm, real server and virtual server configuration will:

- Handle FTP requests from the general public using the ftp.myproduct.com domain.
- Load balance FTP services using the least connections predictor across two real servers.
- Use both the default ICMP ping probe and the configured TCP-FTP probe for verification of port 21.
- Use the VIP 194.56.12.2 port 21.

Enterprise Internal HTTP Server Farm

The enterprise internal HTTP server farm, real server and virtual server configuration will:

- Handle HTTP requests from enterprise employees using the www.myinternal.com domain.
- Load balance HTTP services across two real servers, using the simple round robin selection process.
- Apply the TCP-HTTP probe that verifies port 80 and uses Application Content Verification TCP fail detection.
- Use the VIP 194.56.13.3 port 80.

Enterprise Internal FTP Server Farm

The enterprise internal FTP server farm, real server and virtual server configuration will:

- Handle FTP requests from enterprise employees using the ftp.myinternal.com domain.
- Load balance FTP services using the least connections predictor across two real servers.
- Use both the default ICMP ping probe and the configured TCP-FTP probe for verification of port 21.
- Use the VIP 194.56.13.3 port 21.

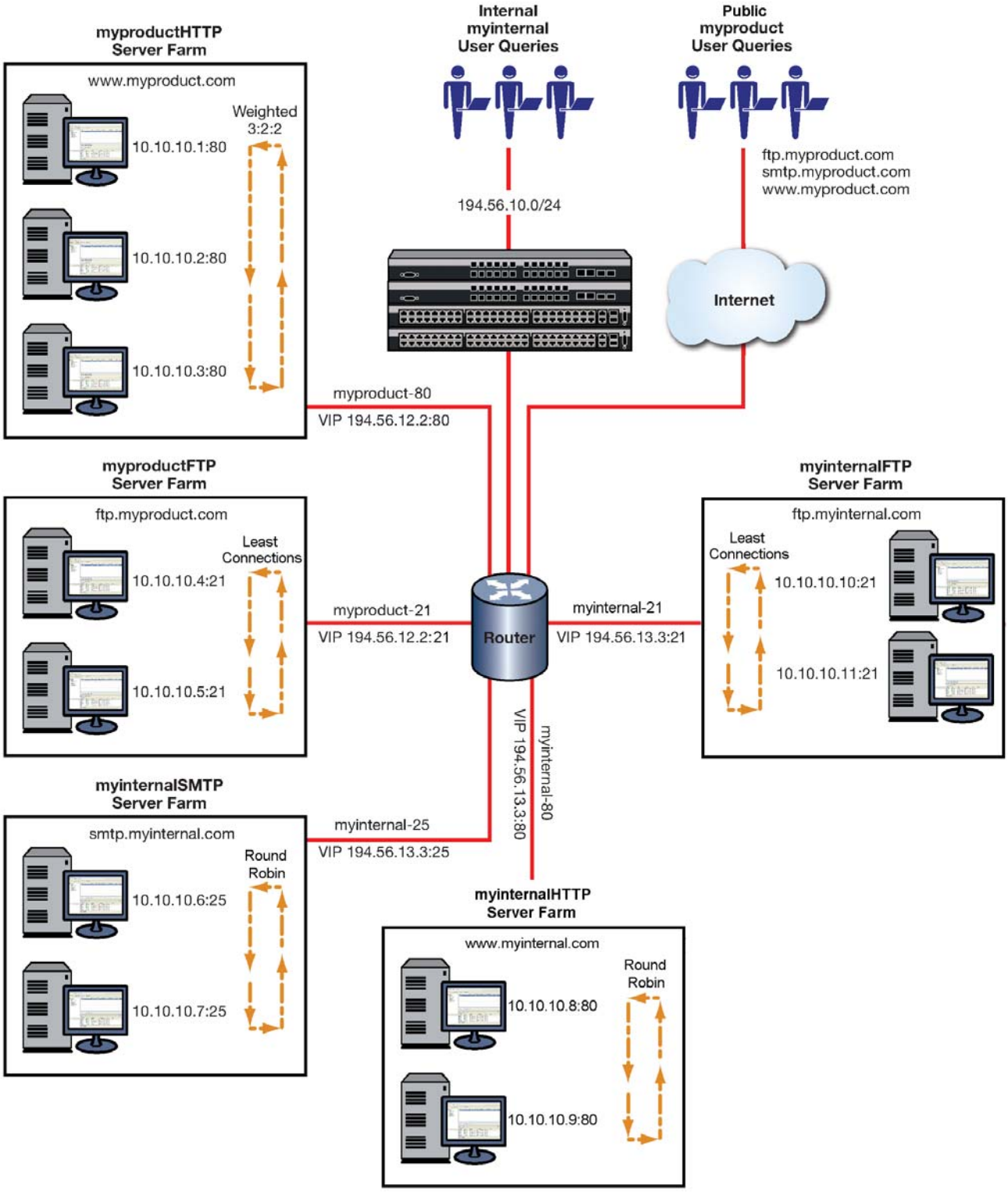
Enterprise Internal SMTP Server Farm

The enterprise internal SMTP server farm, real server and virtual server configuration will:

- Handle SMTP requests from the enterprise employees using the smtp.myproduct.com domain.
- Load balance SMTP services across two real servers, using the simple round robin selection process.
- Use both the default ICMP ping probe and the configured TCP-SMTP probe for verification of port 25.
- Use the VIP 194.56.13.3 port 25.

See Figure 24-3 on page 16 for a presentation of this LSNAT configuration.

Figure 24-3 LSNAT Configuration Example



Configuring the myproductHTTP Server Farm and Real Servers

Configure the myproductHTTP server farm by:

- Naming the server farm **myproductHTTP**
- Configuring round robin as the load balancing algorithm for this server farm (weight will be configured during real server configuration)

Configure the real servers on the myproductHTTP 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: **10.10.10.1:80**, **10.10.10.2:80**, and **10.10.10.3:80**
- Configuring weight for each real server
- 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.

myproductHTTP Server Farm and Real Server CLI Input

```
N Chassis(rw)->configure
N Chassis(su-config)->probe TCP-HTTP tcp
N Chassis(su-config-probe)->faildetect interval 5
N Chassis(su-config-probe)->passdetect interval 5
N Chassis(su-config-probe)->acv request “GET / HTTP/1.1\\r\\nHost:
2.0.0.5\\r\\n\\r\\n”
N Chassis(su-config-probe)->acv reply “HTTP/1.1 200 OK\\r\\n”
N Chassis(su-config-probe)->acv search-depth 50
N Chassis(su-config-probe)->inservice
N 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:           2
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: 50
N Chassis(su-config-probe)->exit
N Chassis(rw-config)->ip slb serverfarm myproductHTTP
N Chassis(rw-config-slb-sfarm)->predictor roundrobin
N Chassis(rw-config-slb-sfarm)->real 10.10.10.1 port 80
N Chassis(rw-config-slb-real)->faildetect probe two TCP-HTTP
N Chassis(rw-config-slb-real)->weight 3
N Chassis(rw-config-slb-real)->inservice
```

```
N Chassis(rw-config-slb-real)->exit
N Chassis(rw-config-slb-sfarm)->real 10.10.10.2 port 80
N Chassis(rw-config-slb-real)->faildetect probe two TCP-HTTP
N Chassis(rw-config-slb-real)->weight 2
N Chassis(rw-config-slb-real)->inservice
N Chassis(rw-config-slb-real)->exit
N Chassis(rw-config-slb-sfarm)->real 10.10.10.3 port 80
N Chassis(rw-config-slb-real)->faildetect probe two TCP-HTTP
N Chassis(rw-config-slb-real)->weight 2
N Chassis(rw-config-slb-real)->inservice
N Chassis(rw-config-slb-real)->exit
N Chassis(rw-config-slb-sfarm)->exit
N Chassis(rw-config)->
```

Configuring myproduct-80 Virtual Server

Configure the virtual server for the myproductHTTP server farm by:

- Naming the virtual server **myproduct-80**
- Associating the virtual server with the myproductHTTP server farm
- Assigning the virtual server IP address with the TCP protocol for www (port 80)
- Setting the idle timeout value of 360 seconds
- Placing the virtual server in service

myproduct-80 Virtual Server CLI Input

```
N Chassis(rw-config)->ip slb vserver myproduct-80
N Chassis(rw-config-slb-vserver)->serverfarm myproductHTTP
N Chassis(rw-config-slb-vserver)->virtual 194.56.12.2 tcp www
N Chassis(rw-config-slb-vserver)->idle timeout 360
N Chassis(rw-config-slb-vserver)->inservice
N Chassis(rw-config-slb-vserver)->exit
N Chassis(rw-config)->
```

Configuring the myproductFTP Server Farm and Real Servers

Configure the myproductFTP server farm by:

- Naming the server farm **myproductFTP**
- Configuring least connections as the load balancing algorithm for this server farm

Configure the real servers on the myproductFTP server farm by:

- Configuring the following real servers: **10.10.10.4:21** and **10.10.10.5:21**
- Configuring the FTP servers for both ping and TCP port service verification using the default ICMP probe in probe **one** and configuring the TCP-FTP probe, using default values, and configuring it for probe **two**
- Enabling each real server by placing each server in service

Notes: We will not modify the maximum number of active connections allowed on any real server for this configuration example.

myproductFTP Server Farm and Real Server CLI Input

```
N Chassis(su-config)->probe TCP-FTP tcp
N Chassis(su-config-probe)->inservice
N Chassis(su-config-probe)->exit
N Chassis(rw-config)->ip slb serverfarm myproductFTP
N Chassis(rw-config-slb-sfarm)->predictor leastconns
N Chassis(rw-config-slb-sfarm)->real 10.10.10.4 port 21
N Chassis(rw-config-slb-real)->faildetect probe two TCP-FTP
N Chassis(rw-config-slb-real)->inservice
N Chassis(rw-config-slb-real)->exit
N Chassis(rw-config-slb-sfarm)->real 10.10.10.5 port 21
N Chassis(rw-config-slb-real)->faildetect probe two TCP-FTP
N Chassis(rw-config-slb-real)->inservice
N Chassis(rw-config-slb-real)->exit
```

Configuring myproduct-21 Virtual Server

Configure the virtual server for the myproductFTP server farm by:

- Globally setting the FTP control port for all virtual servers to **21**
- Naming the virtual server **myproduct-21**
- Associating the virtual server with the **myproductFTP** server farm
- Assigning the virtual server IP address, TCP protocol, and FTP service
- Setting the idle timeout value of 360 seconds
- Placing the virtual server in service

myproductFTP Virtual Server CLI Input

```
N Chassis(rw-config)->ip slb ftpctrlport 21
N Chassis(rw-config)->ip slb vserver myproduct-21
N Chassis(rw-config-slb-vserver)->serverfarm myproductFTP
N Chassis(rw-config-slb-vserver)->virtual 194.56.12.2 tcp ftp
N Chassis(rw-config-slb-vserver)->idle timeout 360
N Chassis(rw-config-slb-vserver)->inservice
N Chassis(rw-config-slb-vserver)->exit
N Chassis(rw-config)->
```

Configuring the myinternalHTTP Server Farm and Real Servers

Configure the myinternalHTTP server farm by:

- Naming the server farm **myinternalHTTP**
- Configure simple round robin as the load balancing algorithm for this server farm

Configure the real servers on the myinternal server farm by:

- Configuring the following real servers: **10.10.10.8:80** and **10.10.10.9:80**
- Configuring the HTTP servers with probe TCP-HTTP first configured in "myproductHTTP Server Farm and Real Server CLI Input" on page 24-17 for probe two
- Configuring a faildetect command string, reply string, and read till index value for each HTTP server
- Enabling each real server by placing each server in service

myinternalHTTP Server Farm and Real Server CLI Input

```
N Chassis(rw-config)->ip slb serverfarm myinternalHTTP
N Chassis(rw-config-slb-sfarm)->predictor roundrobin
N Chassis(rw-config-slb-sfarm)->real 10.10.10.8 port 80
N Chassis(rw-config-slb-real)->faildetect probe two TCP-HTTP
N Chassis(rw-config-slb-real)->inservice
N Chassis(rw-config-slb-real)->exit
N Chassis(rw-config-slb-sfarm)->real 10.10.10.9 port 80
N Chassis(rw-config-slb-real)->faildetect probe two TCP-HTTP
N Chassis(rw-config-slb-real)->inservice
N Chassis(rw-config-slb-real)->exit
N Chassis(rw-config-slb-sfarm)->exit
N Chassis(rw-config)->
```

Configuring myinternal-80 Virtual Server

Configure the virtual server for the myinternalHTTP server farm by:

- Naming the virtual server **myinternal-80**
- Associating the virtual server with the myinternalHTTP server farm
- Assigning the virtual server IP address with the idle timeout value of 360 seconds
- Placing the virtual server in service

myinternal-80 Virtual Server CLI Input

```
N Chassis(rw-config)->ip slb vserver myinternal-80
N Chassis(rw-config-slb-vserver)->serverfarm myinternalHTTP
N Chassis(rw-config-slb-vserver)->virtual 194.56.13.3 tcp www
N Chassis(rw-config-slb-vserver)->idle timeout 360
N Chassis(rw-config-slb-vserver)->inservice
N Chassis(rw-config-slb-vserver)->
```

Configuring the myinternalFTP Server Farm Real Servers

Configure the myinternalFTP server farm by:

- Naming the server farm **myinternalFTP**
- Configuring least connections as the load balancing algorithm for this server farm

Configure the real servers on the myinternalFTP server farm by:

- Configuring the following real servers: **10.10.10.10:21** and **10.10.10.11:21**
- Configuring the FTP servers for both ping and TCP port service verification using the TCP-FTP probe first configured in "myproductFTP Server Farm and Real Server CLI Input" on page 24-19
- Enabling each real server by placing each server in service

myinternalFTP Server Farm and Real Servers CLI Input

```
N Chassis(rw-config)->ip slb serverfarm myinternalFTP
N Chassis(rw-config-slb-sfarm)->predictor leastconns
N Chassis(rw-config-slb-sfarm)->real 10.10.10.10 port 21
N Chassis(rw-config-slb-real)->faildetect probe two TCP-FTP
N Chassis(rw-config-slb-real)->inservice
N Chassis(rw-config-slb-real)->exit
N Chassis(rw-config-slb-sfarm)->real 10.10.10.11 port 21
N Chassis(rw-config-slb-real)->faildetect probe two TCP-FTP
N Chassis(rw-config-slb-real)->inservice
N Chassis(rw-config-slb-real)->exit
N Chassis(rw-config-slb-sfarm)->exit
N Chassis(rw-config)->
```

Configuring myinternal-21 Virtual Server

Configure the virtual server for the myinternalFTP server farm by:

- Naming the virtual server **myinternal-21**
- Associating the virtual server with the **myinternalFTP** server farm
- Assigning the virtual server IP address with the idle timeout value of 360 seconds
- Placing the virtual server in service

myinternal-21 Virtual Server CLI Input

```
N Chassis(rw-config)->ip slb vserver myinternal-21
N Chassis(rw-config-slb-vserver)->serverfarm myinternalFTP
N Chassis(rw-config-slb-vserver)->virtual 194.56.13.3 tcp 21
N Chassis(rw-config-slb-vserver)->idle timeout 360
N Chassis(rw-config-slb-vserver)->inservice
N Chassis(rw-config-slb-vserver)->exit
N Chassis(rw-config)->
```

Configuring the myinternalSMTP Server Farm and Real Servers

Configure the myinternalSMTP server farm by:

- Naming the server farm **myinternalSMTP**
- Configuring simple round robin as the load balancing algorithm for this server farm

Configure the real servers on the myinternalSMTP server farm by:

- Configuring the following real servers: **10.10.10.6:25** and **10.10.10.7:25**
- Configuring the SMTP servers for both ping and TCP port service verification using the default ICMP probe in probe **one** and configuring the TCP-SMTP probe, using default values, and configuring it for probe **two**
- Enabling each real server by placing each server in service

Notes: We will not modify the maximum number of active connections allowed on any real server for this configuration example.

myinternalSMTP Server Farm and Real Servers CLI Input

```
N Chassis(su-config)->probe TCP-SMTP tcp
N Chassis(su-config-probe)->inservice
N Chassis(su-config-probe)->exit
N Chassis(rw-config)->ip slb serverfarm myinternalSMTP
N Chassis(rw-config-slb-sfarm)->predictor roundrobin
N Chassis(rw-config-slb-sfarm)->real 10.10.10.6 port 25
N Chassis(rw-config-slb-real)->faildetect probe two TCP-SMTP
N Chassis(rw-config-slb-real)->inservice
N Chassis(rw-config-slb-real)->exit
N Chassis(rw-config-slb-sfarm)->real 10.10.10.7 port 25
N Chassis(rw-config-slb-real)->faildetect probe two TCP-SMTP
N Chassis(rw-config-slb-real)->inservice
N Chassis(rw-config-slb-real)->exit
```

Configuring myinternal-25 Virtual Server

Configure the virtual server for the myinternalSMTP server farm by:

- Naming the virtual server **myinternal-25**
- Associating the virtual server with the **myinternalSMTP** server farm
- Assigning the virtual server IP address with the idle timeout value of 360 seconds
- Placing the virtual server in service

myinternal-25 Virtual Server CLI Input

```
N Chassis(rw-config)->ip slb vserver myinternal-25
N Chassis(rw-config-slb-vserver)->serverfarm myinternalSMTP
N Chassis(rw-config-slb-vserver)->virtual 194.56.13.3 tcp 25
N Chassis(rw-config-slb-vserver)->idle timeout 360
N Chassis(rw-config-slb-vserver)->inservice
N Chassis(rw-config-slb-vserver)->exit
N Chassis(rw-config)->
```

This completes the LSNAT configuration example.

Terms and Definitions

Table 24-5 lists terms and definitions used in this LSNAT configuration discussion.

Table 24-5 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.
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.

Table 24-5 LSNAT Configuration Terms and Definitions (continued)

Term	Definition
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.

25

Transparent Web Cache Balancing (TWCB) Configuration

This document provides the following information about configuring Transparent Web Cache Balancing on the Enterasys N-Series platform.

Using Transparent Web Cache Balancing (TWCB) on Your Network

Transparent Web Caching is a means of transparently redirecting a client's 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 transparent 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 N-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.
- Apply the TWCB web-cache to an outbound interface, 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.

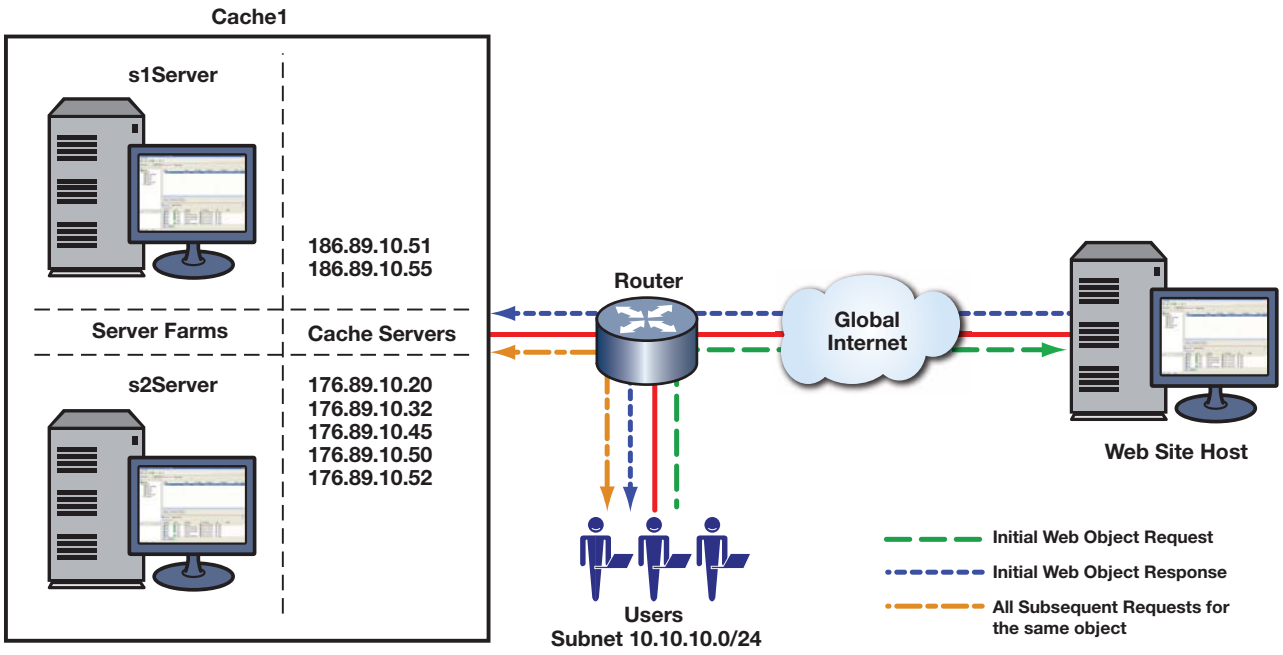
Figure 25-1 provides an overview of a TWCB configuration. In our overview, Cache1 is the name of the web-cache. It is made up of two server farms: s1Server and s2Server. The s1Server server farm is configured with 2 cache-servers from the 186.89.0.0 subnet. The s2Server server farm is configured with 5 cache-servers from the 176.89.0.0 subnet. Web objects for each end-user are cached on a cache server.

The N-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 a 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 a 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 sites 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 25-1 TWCB Configuration Overview



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 5 server farms that can be associated with the web-cache.

There are three aspects to configuring a server farm:

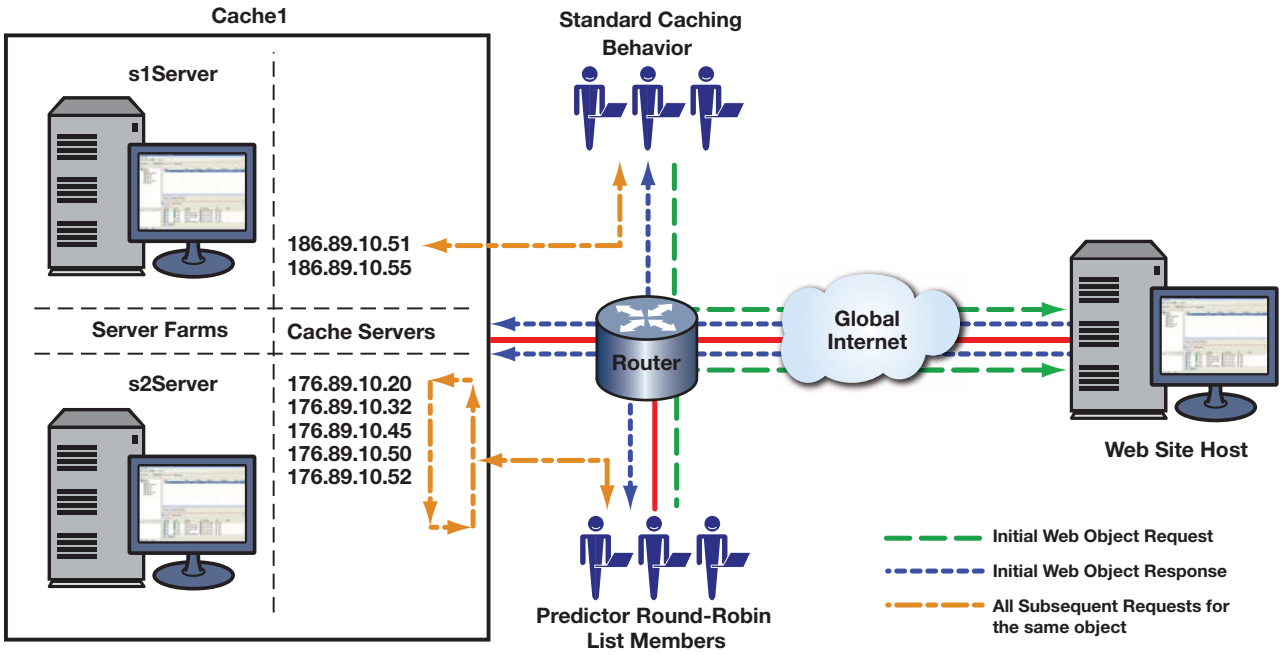
- Creating the server farm
- Associating one or more cache-servers with the server farm
- Optionally configuring some users to be members of a round-robin list on that server farm.

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 25-2 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 s1Server server farm 186.89.10.51 cache server. The s2Server 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 s2Server server farm.

Figure 25-2 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.

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 7, **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 7-5 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

```
N Chassis(su)->configure
N Chassis(su-config)->probe TCP-HTTP tcp
N Chassis(su-config-probe)->inservice
N Chassis(su-config-probe)->acv request “GET / HTTP/1.1\\r\\nHost:
2.0.0.5\\r\\n\\r\\n”
N Chassis(su-config-probe)->acv reply “HTTP/1.1 200 OK\\r\\n”
N 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
N Chassis(su-config-probe)->exit
N Chassis(su-config)->ip twcb wcserverfarm s1Server
N Chassis(config-twcb-wcsfarm)->cache 186.89.10.51
N Chassis(config-twcb-cache)->faildetect probe one TCP-HTTP
N Chassis(config-twcb-cache)->faildetect app-port 8080
N Chassis(config-twcb-cache)->inservice
N 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.

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.

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** command to display active TWCB bindings for this device.

Configuring TWCB

This section provides details for the configuration of TWCB on the N-Series products.

Table 25-1 lists TWCB parameters and their default values.

Table 25-1 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.
ping-int	Specifies the period between each test of the TWCB cache server up or down status.	5 seconds
ping-retries	Specifies the number of times the ping faildetect method will test the TWCB cache server up or down status.	4
app-int	Specifies the period between each test of the TWCB cache server up or down status.	15 seconds
app-retries	Specifies the number of times the application faildetect method will test the TWCB cache server up or down status.	4
maxconns	Specifies the maximum number of bindings allowed for this server.	0 (no limit)

Configuring the Server Farm

Procedure 25-1 describes how to configure a TWCB server farm.

Procedure 25-1 TWCB Server Farm Configuration

Step	Task	Command(s)
1.	Create the server farm.	**ip twcb wcserverfarm** *serverfarm-name*
2.	Associate a cache server with the server farm.	**cache** *ip-address*
3.	Optionally configure a predictor round-robin list.	**predictor** {**dest-ip-hash** | **roundrobin** *ip-address-begin ip-address-end*}
4.	Optionally configure a cache server round-robin weight.	**weight** *weight*

Configuring the Cache Server

Procedure 25-2 describes how to configure a TWCB cache server.

Procedure 25-2 TWCB Cache Server Configuration

Step	Task	Command(s)
1.	Create the cache server.	**cache** *ip-address*

Procedure 25-2 TWCB Cache Server Configuration (continued)

Step	Task	Command(s)
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.	Optionally change the maximum number of bindings allowed for this cache server.	**maxconns** *number*
7.	Place the cache server in service.	**inservice**

Configuring the Web-Cache

Procedure 25-3 describes how to configure a TWCB web-cache.

Procedure 25-3 TWCB Web-Cache Configuration

Step	Task	Command(s)
1.	Create a web-cache using the specified name.	**ip 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**
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.	**bypass-list range** *begin-ip-address end-ip-address*
7.	Optionally permit or deny redirection of HTTP requests for the list of clients to this web-cache.	**hosts** {**permit** | **deny**} **redirect range** *begin-ip-address end-ip-address*
8.	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.

Table 25-2 describes how to configure this interface for HTTP outbound redirection.

Table 25-2 HTTP Outbound Interface Configuration

Step	Task	Command(s)
1.	Redirect outbound HTTP traffic from this outbound interface to the cache servers.	**ip twcb** *webcache-name* **redirect out**

Displaying TWCB Statistics/Information

Table 25-3 describes how to display TWCB statistics/information.

Table 25-3 Displaying TWCB Statistics

Task	Command(s)
Display server farm configuration data.	**show ip twcb wcserverfarms** [*serverfarm-name* | **detail**]
Display web-cache configuration data.	**show ip twcb webcaches** [*webcache-name* | **detail**]
Display TWCB bindings.	**show ip twcb bindings** {**summary** | **id** *id* | **match** {*sip* | *****} {*dip* | *****} [**detail**]}
Display TWCB caches.	**show ip twcb caches** [*serverfarm-name*] [**detail**]
Display TWCB configuration information.	**show ip twcb info**
Display cache server statistical data.	**show ip twcb statistics**

TWCB Configuration Example

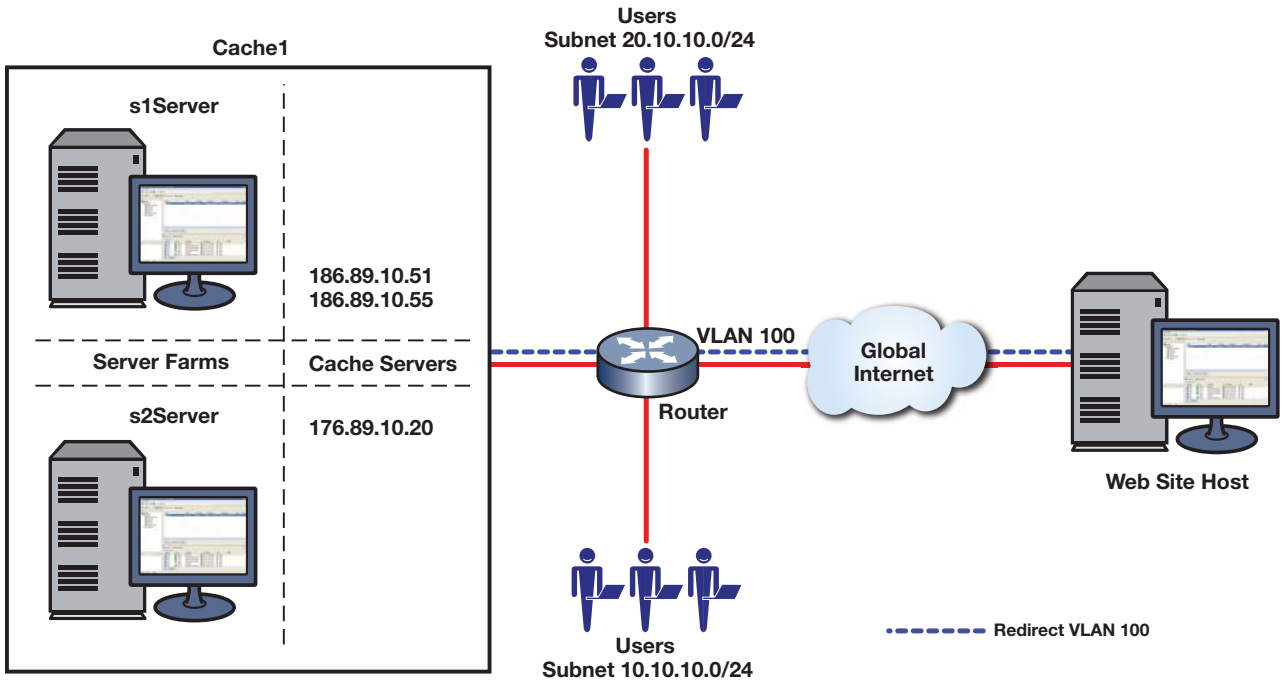
In this TWCB configuration example we will step through the configuration of two server farms named s1Server and s2Server. The s1Server server farm will have round-robin predictor destination IP address ranges associated with it from both the 50.10.10.0/24 subnet and the 40.10.10.0/24 subnet, for hosts with an expectation of heavy web-site access requirements. All other destination addresses not members of a predictor round-robin list or denied host redirect will use the s2Server server farm with a standard cache.

The s1Server will have cache servers 186.89.10.51 and 186.89.10.55 associated with it. The s2Server will have cache server 196.89.10.20 associated with it. s1Server 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 s2Server cache servers 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.

The web-cache will be configured as cache1. The HTTP port being used has been changed from the default of 80 to 8080. A bypass list has been configured to deny TWCB functionality for web requests to web host sites 50.10.10.30 to 50.10.10.43 because these sites require IP address authentication for user access. End-users 10.10.10.25 to 10.10.10.30 have been configured to deny TWCB functionality.

See Figure 25-3 for a depiction of the example setup.

Figure 25-3 TWCB Configuration Example Overview



Configure the s1Server Server Farm

Configure the ICMP probe:

```
N Chassis(rw)->configure
N Chassis(su-config)->probe s1-ICMP icmp
N Chassis(su-config-probe)->faildetect count 5 interval 4
N Chassis(su-config-probe)->exit
```

Create the server farm:

```
N Chassis(rw-config)->ip twcb wcserverfarm s1Server
N Chassis(rw-config-twcb-wcsfarm)->
```

Configure the end-users that will use this server farm by setting the round-robin predictor ranges:

```
N Chassis(rw-config-twcb-wcsfarm)->predictor roundrobin 50.10.10.01 50.10.10.15
N Chassis(rw-config-twcb-wcsfarm)->predictor roundrobin 40.10.10.45 40.10.10.60
N Chassis(rw-config-twcb-wcsfarm)->
```

Configure cache server 186.89.10.51:

```
N Chassis(rw-config-twcb-wcsfarm)->cache 186.89.10.51
N Chassis(rw-config-twcb-cache)->faildetect probe one s1-ICMP
N Chassis(rw-config-twcb-cache)->maxconns 800
N Chassis(rw-config-twcb-cache)->inservice
N Chassis(rw-config-twcb-cache)->exit
N Chassis(rw-config-twcb-wcsfarm)->
```

Configure cache server 186.89.10.55:

```
N Chassis(rw-config-twcb-wcsfarm)->cache 186.89.10.55
N Chassis(rw-config-twcb-cache)->faildetect probe one s1-ICMP
N Chassis(rw-config-twcb-cache)->maxconns 800
N Chassis(rw-config-twcb-cache)->inservice
N Chassis(rw-config-twcb-cache)->exit
N Chassis(rw-config-twcb-wcsfarm)->exit
N Chassis(rw-config)->
```

Configure the s2Server Server Farm

Configure the TCP probe:

```
N Chassis(rw)->configure
N Chassis(su-config)->probe s2-TCP tcp
N Chassis(su-config-probe)->faildetect count 5 interval 12
N Chassis(su-config-probe)->exit
```

Configure server farm s2Server:

```
N Chassis(rw-config)->ip twcb wcserverfarm s2Server
N Chassis(rw-config-twcb-wcsfarm)->
```

Configure cache server 176.89.10.20:

```
N Chassis(rw-config-twcb-wcsfarm)->cache 176.89.10.20
N Chassis(rw-config-twcb-cache)->faildetect probe two s2-TCP
N Chassis(rw-config-twcb-cache)->faildetect app-int 12
N Chassis(rw-config-twcb-cache)->faildetect app-retries 5
N Chassis(rw-config-twcb-cache)->maxconns 800
N Chassis(rw-config-twcb-cache)->inservice
N Chassis(rw-config-twcb-cache)->exit
N Chassis(rw-config-twcb-wcsfarm)->exit
N Chassis(rw-config)->
```

Configure the cache1 Web Cache

Configure the web-cache cache1:

```
N Chassis(rw-config)->ip twcb webcache cache1
N Chassis(rw-config-twcb-webcache)->http-port 8080
N Chassis(rw-config-twcb-webcache)->serverfarm s1Server
N Chassis(rw-config-twcb-webcache)->serverfarm s2Server
N Chassis(rw-config-twcb-webcache)->bypass-list range 50.10.10.30 50.10.10.43
N Chassis(rw-config-twcb-webcache)->hosts redirect deny redirect range
10.10.10.25 10.10.10.30
N Chassis(rw-config-twcb-webcache)->exit
N Chassis(rw-config)->
```

Configure the outbound interface that connects with the internet:

```
N Chassis(rw-config)->interface vlan 100
```

```
N Chassis(rw-config-intf-vlan.0.100)->ip twcb cache1 redirect out
N Chassis(rw-config-intf-vlan.0.100)->end
N Chassis(rw)->
```

This completes the TWCB configuration example.

26

Virtual Router Redundancy Protocol (VRRP) Configuration

This document describes the Virtual Router Redundancy Protocol (VRRP) feature and its configuration on Enterasys N-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.

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.

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 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 26-1 A Basic VRRP Topology

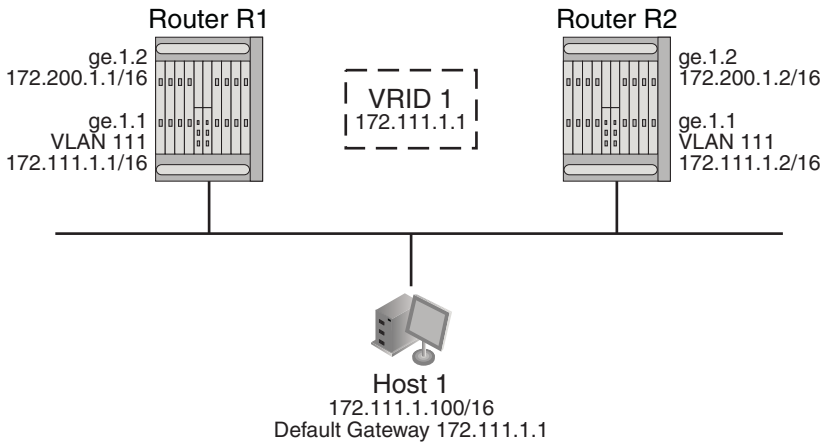


Figure 26-1 shows a basic VRRP topology with a single virtual router. Routers R1 and R2 are both configured with one virtual router (VRID 1). Router R1 serves as the master and Router R2 serves as the backup. The hosts are configured to use 172.111.1.1/16 as the default route.

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 7-5 for default ICMP probe details. Probes are configured in the tracked object manager. See Chapter 7, **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.

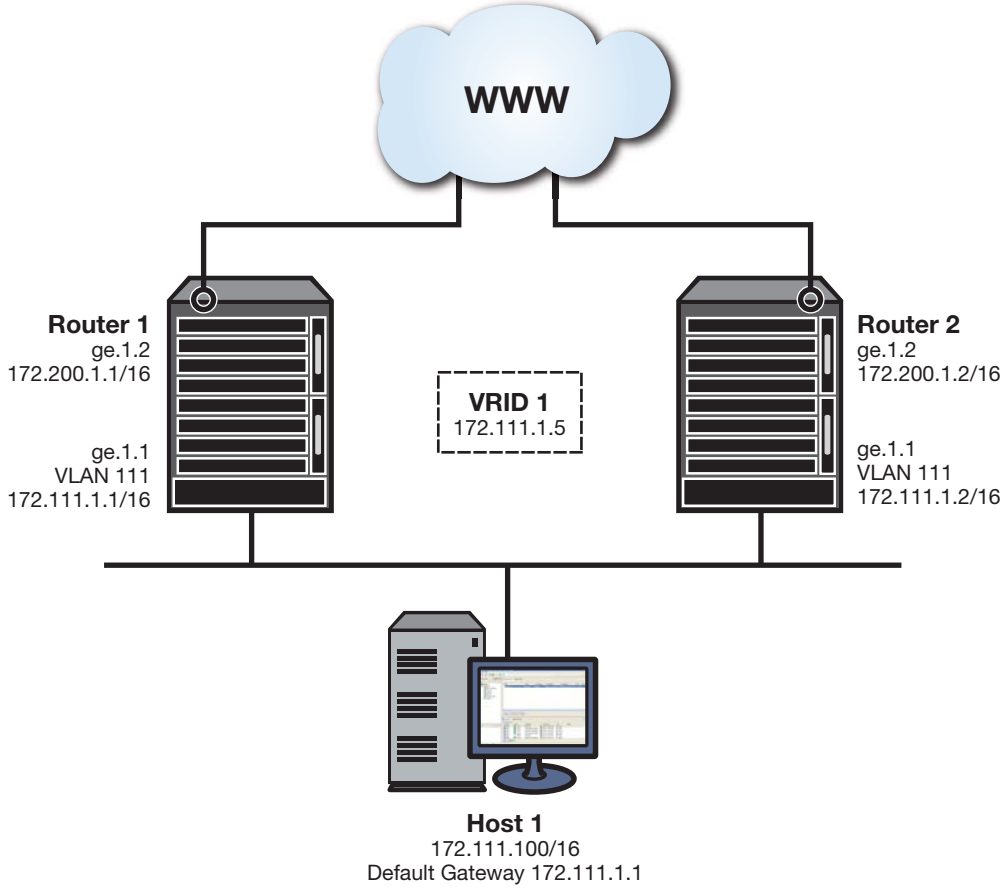
The default priority setting is enabled with a value of **10**. Setting the critical-IP address priority to enabled signals that the critical-IP will affect the operational priority for the VRID. Setting the priority to disabled signals the critical-IP interface state will have no effect on the operational priority for the VRID.

Up to 2048 critical-IP addresses can be configured on a device. Up to 10 per VRID. The DFE Gold module supports up to 4 critical-ip addresses per device.

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 26-2 presents a typical critical-IP address configuration.

Figure 26-2 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 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 N-Series products.

Table 26-1 lists VRRP parameters and their default values.

Table 26-1 Default VRRP Parameters

Parameter	Description	Default Value
priority	Specifies the router priority for the master election for this virtual router.	100
advertise-interval	Specifies the interval between the advertisement the master sends to other routers participating in the selection process.	1 second

Table 26-1 Default VRRP Parameters (continued)

Parameter	Description	Default Value
interface-up delay	Specifies the delay in seconds for the VRID state transition from interface down to interface up	0 seconds (no delay)
VRRP preemption	Specifies whether higher priority backup VRRP routers can preempt a lower priority master VRRP router and become master.	enabled
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

Procedure 26-1 describes how to configure VRRP.

Procedure 26-1 Configuring VRRP

Step	Task	Command(s)
1.	Create a virtual router instance. Supported VRRP Versions: • **v2-IPv4** - FRC2338 • **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*

Procedure 26-1 Configuring VRRP (continued)

Step	Task	Command(s)
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*

Table 26-2 describes how to display VRRP information and statistics.

Table 26-2 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

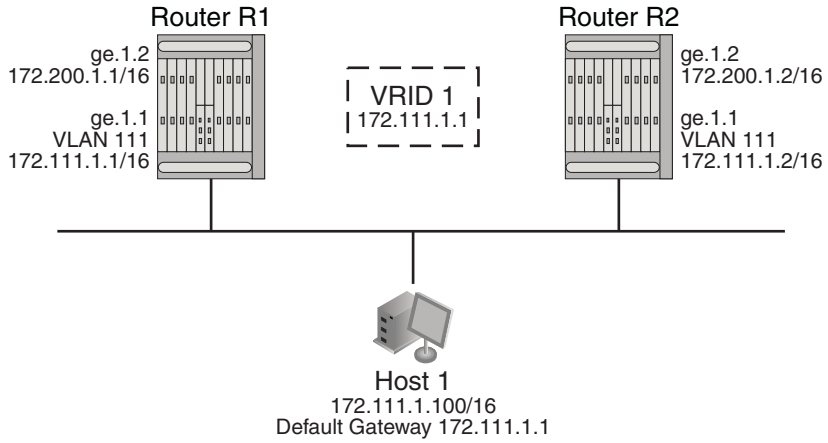
Basic VRRP Configuration Example

Figure 26-3 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 26-3 Basic Configuration Example



Router R1 Configuration of VRRP Instance 1

```
N Chassis(rw)->configure
N Chassis(rw-config)->interface vlan 111
N Chassis(rw-config-intf-vlan.0.111)->vrrp create 1 v3-IPv4
N Chassis(rw-config-intf-vlan.0.111)->vrrp address 1 172.111.1.1
N Chassis(rw-config-intf-vlan.0.111)->vrrp advertise-interval 1 centiseconds 150
N Chassis(rw-config-intf-vlan.0.111)->vrrp enable 1
N Chassis(rw-config-intf-vlan.0.111)->no shutdown
N Chassis(rw-config-intf-vlan.0.111)->exit
N 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
N Chassis(rw-config)->
```

Router R2 Configuration of VRRP Instance 1

```
N Chassis(rw)->configure
N Chassis(rw-config)->interface vlan 111
N Chassis(rw-config-intf-vlan.0.111)->vrrp create 1 v3-IPv4
N Chassis(rw-config-intf-vlan.0.111)->vrrp address 1 172.111.1.1
```

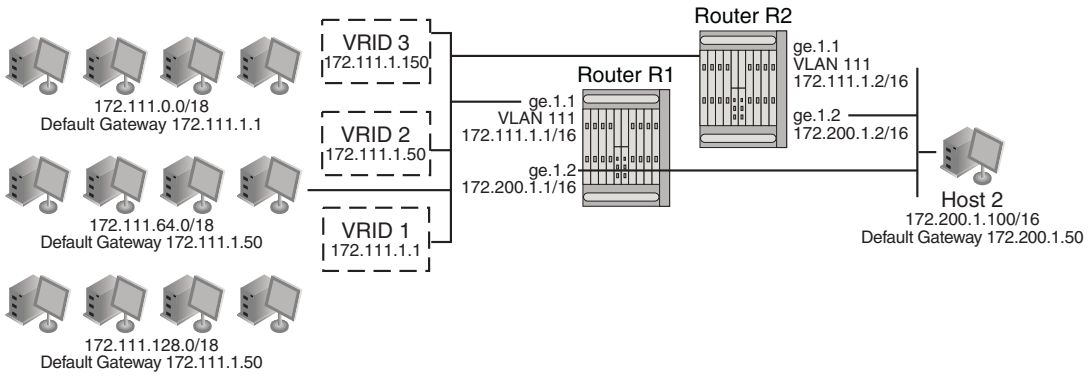
```
N Chassis(rw-config-intf-vlan.0.111)->vrrp advertise-interval 1 centiseconds 150
N Chassis(rw-config-intf-vlan.0.111)->vrrp enable 1
N Chassis(rw-config-intf-vlan.0.111)->no shutdown
N Chassis(rw-config-intf-vlan.0.111)->exit
N Chassis(rw-config)->show ip vrrp verbose
Interface: vlan.0.111
 VRID: 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
N 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 26-4 shows a multi-backup sample configuration.

Figure 26-4 Multi-Backup VRRP Configuration Example



Three VRRP instances are configured on VLAN 111 for both N-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

```
N Chassis(rw)->configure
N Chassis(rw-config)->interface vlan 111
N Chassis(rw-config-intf-vlan.0.111)->vrrp create 1 v2-IPv4
N Chassis(rw-config-intf-vlan.0.111)->vrrp address 1 172.111.1.1
N Chassis(rw-config-intf-vlan.0.111)->vrrp enable 1
N Chassis(rw-config-intf-vlan.0.111)->vrrp create 2 v2-IPv4
N Chassis(rw-config-intf-vlan.0.111)->vrrp address 2 172.111.1.50
N Chassis(rw-config-intf-vlan.0.111)->vrrp enable 2
N Chassis(rw-config-intf-vlan.0.111)->vrrp create 3 v2-IPv4
N Chassis(rw-config-intf-vlan.0.111)->vrrp address 3 172.111.1.150
N Chassis(rw-config-intf-vlan.0.111)->vrrp priority 3 200
N Chassis(rw-config-intf-vlan.0.111)->vrrp enable 3
N Chassis(rw-config-intf-vlan.0.111)->no shutdown
N Chassis(rw-config-intf-vlan.0.111)->exit
N Chassis(rw-config)->
```

Router R2

```
N Chassis(rw)->configure
N Chassis(rw-config)->interface vlan 111
N Chassis(rw-config-intf-vlan.0.111)->vrrp create 1 v2-IPv4
N Chassis(rw-config-intf-vlan.0.111)->vrrp address 1 172.111.1.1
N Chassis(rw-config-intf-vlan.0.111)->vrrp enable 1
N Chassis(rw-config-intf-vlan.0.111)->vrrp create 2 v2-IPv4
N Chassis(rw-config-intf-vlan.0.111)->vrrp address 2 172.111.1.50
N Chassis(rw-config-intf-vlan.0.111)->vrrp priority 2 200
N Chassis(rw-config-intf-vlan.0.111)->vrrp enable 2
N Chassis(rw-config-intf-vlan.0.111)->vrrp create 3 v2-IPv4
N Chassis(rw-config-intf-vlan.0.111)->vrrp address 3 172.111.1.150
N Chassis(rw-config-intf-vlan.0.111)->vrrp enable 3
N Chassis(rw-config-intf-vlan.0.111)->no shutdown
N Chassis(rw-config-intf-vlan.0.111)->exit
N Chassis(rw-config)->
```

Terms and Definitions

Table 26-3 lists terms and definitions used in this VRRP configuration discussion.

Table 26-3 VRRP Configuration Terms and Definitions

Term	Definition
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.
VRID	Virtual Router ID — a unique number associated with each virtual router.
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.
Backup	The set of VRRP routers available to assume forwarding responsibility for a virtual router should the current Master fail.
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.
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.
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.
Accept Mode	When enabled, it allows the master for this virtual router to accept IP packets for the configured associated IP address list.

27

Security Configuration

This document provides the following information about configuring security features on the Enterasys N-Series platforms.

Using Security Features in Your Network

The N-Series platform supports the following security features.

Note: The security feature Flow Setup Throttling (FST) is also supported by the N-Series platform (see Chapter 28, **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.

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 Enterasys N-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 N-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:

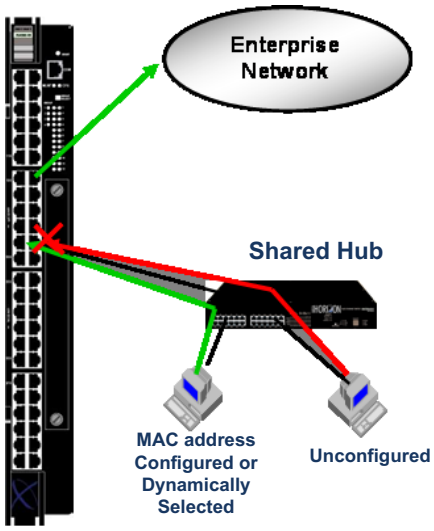
- Static MAC Locking - Locking one or more specified MAC addresses to a port.
- Dynamic MAC Locking - Locking one or more MAC addresses to a port based on first arrival of received frames after dynamic MAC locking is enabled. The configuration specifies the maximum number of end users that will be allowed. 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 27-1 displays two users on a shared hub connected to an N-Series switch port. Data from the MAC locked user is forwarded on to the enterprise network. Data from the unconfigured user is dropped.

Figure 27-1 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 N-Series SSHv2 implementation includes:

- Data privacy
- Communication integrity

An SSH server resides on the N-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 N-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.

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 N-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 N-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 N-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 N-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 N-Series device.

Host DoS

The Host DoS feature provides protection against all known DoS attack mitigation types.

Table 27-2 lists the configurable Host DoS mitigation types.

Table 27-1 Host DoS Mitigation Types

Threat	Description	Action
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.

Setting Logging Event Rates

The event-rate at which logging is displayed when logging is enabled can be set using the **event-rate** option of the **hostdos** command in global configuration command mode by specifying

the number of logs per specified time period. Supported time periods are seconds, minutes, hours, and days. The event rate default is for all logs to display.

Logging can be disabled using the **hostdos** command **nolog** option.

Configuring Security

Table 27-2 lists Security parameters and their default values.

Table 27-2 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 27-1 describes how to configure MAC locking on an N-Series device. All MAC locking commands can be entered in any command mode.

Procedure 27-1 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*]
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 27-3 describes how to manage MAC locking on an N-Series port. All MAC locking commands can be entered in any command mode.

Table 27-3 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**:

```
N Chassis(rw)->set maclock enable ge.1.1-5
N 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:

```
N Chassis(rw)->set maclock static ge.1.1 3
N Chassis(rw)->set maclock 00-10-a4-e5-08-4e ge.1.1 create
N Chassis(rw)->set maclock 08-00-20-7c-e0-db ge.1.1 create
N Chassis(rw)->set maclock 00-60-08-14-4b-15 ge.1.1 create
N Chassis(rw)->set maclock 00-01-f4-2c-ad-b4 ge.1.1 create
Set failed for ge.1.1.
N 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
N 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**:

```
N Chassis(rw)->set maclock firstarrival ge.1.2-5 15
N Chassis(rw)->set maclock trap ge.1.1-5 enable
N Chassis(rw)->
```

Configuring Secure Shell

Procedure 27-2 describes how to configure Secure Shell on an N-Series device. Secure Shell commands can be entered in any command mode.

Procedure 27-2 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.	Verify the SSH state.	**show ssh state**

SSH Configuration Example

The following commands enable and verify SSH:

```
N Chassis(rw)->set ssh enable
N Chassis(rw)->show ssh state
SSH Server state: Enabled
N Chassis(rw)->
```

The following command reinitializes the host key on the SSH server:

```
N Chassis(rw)->set ssh hostkey reinitialize
```

Configuring TACACS+

Procedure 27-3 describes how to configure TACACS+ on an N-Series device. TACACS+ commands can be entered in any command mode.

Procedure 27-3 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 27-4 describes how to manage TACACS+ on an N-Series device. All TACACS+ commands can be entered in any command mode.

Table 27-4 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:

```
N 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.

```
N Chassis(rw)->set tacacs server 1 10.10.10.20 49 mysecret1
N Chassis(rw)->set tacacs server 2 10.10.10.30 49 mysecret2
N 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
N Chassis(rw)->
```

The following command enables and verifies session authorization for the exec service:

```
N Chassis(rw)->set tacacs session authorization service exec
N 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'
N 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:

```
N Chassis(rw)->set tacacs session accounting enable
N Chassis(rw)->show tacacs session accounting
 TACACS+ session accounting state:     enabled
N Chassis(rw)->set tacacs command accounting enable
N Chassis(rw)->set tacacs command authorization enable
N Chassis(rw)->
```

The following command enables the TCP single connection feature for this device:

```
N Chassis(rw)->set tacacs singleconnect
N Chassis(rw)->
```

Configuring Host DoS

Procedure 27-4 describes how to configure Host DoS on an N-Series device. Host DoS configuration commands are entered in global configuration command mode.

Procedure 27-4 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	**hostdos** *mitigation-type*
3.	Optionally, set a logging event-rate for one or all DoS attack types, specifying the rate per specified time period.	**hostdos** *mitigation-type* **event-rate** *count* {**per-seconds** | **per-minutes** | **per-hours** | **per-days**}
4.	Optionally, disable logging for the specified DoS attack types.	**hostdos** *mitigation-type* **nolog**

Procedure 27-4 Host DoS Configuration (continued)

Step	Task	Command(s)
5.	Optionally, specify an ICMP maximum packet size for **icmpsize** mitigation.	**hostdos icmpsize maxlength** *length*

Table 27-4 describes how to display Host DoS configuration state and counters on an N-Series device.

Table 27-5 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

```
N Chassis(rw-config)->hostdos enable
N Chassis(rw-config)->hostDoS spoof rate 5 per-minute
N Chassis(rw-config)->hostdos xmastree nolog
N 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
N Chassis(rw-config)->
```

28

Flow Setup Throttling Configuration

This document provides the following information about configuring flow setup throttling on the Enterasys N-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 Enterasys 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 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 IP 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 N-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 28-1 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 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 N-Series product.

Table 28-1 lists FST parameters and their default values.

Table 28-1 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 28-1 describes how to configure FST.

Procedure 28-1 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. • **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*} [**userport** | **serverport** | **aggregateduser** | **interswitchlink** | **unspecified**]

Procedure 28-1 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. • 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**] [**userport** | **serverport** | **aggregateduser** | **interswitchlink** | **unspecified**]
3.	Set the ports to be used by the specified port classification. • See step 1 on page 28-5 for port classification definitions. If no port-string is specified, the specified port classification is applied to all ports.	**set flowlimit port class** {**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 28-1 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 28-2 describes how to manage link aggregation.

Table 28-2 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. • **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**} [**userport** | **serverport** | **aggregateduser** | **interswitchlink** | **unspecified**]

Table 28-2 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. • **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**] [**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 28-3 describes how to display link aggregation information and statistics.

Table 28-3 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** [**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 28-2. 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

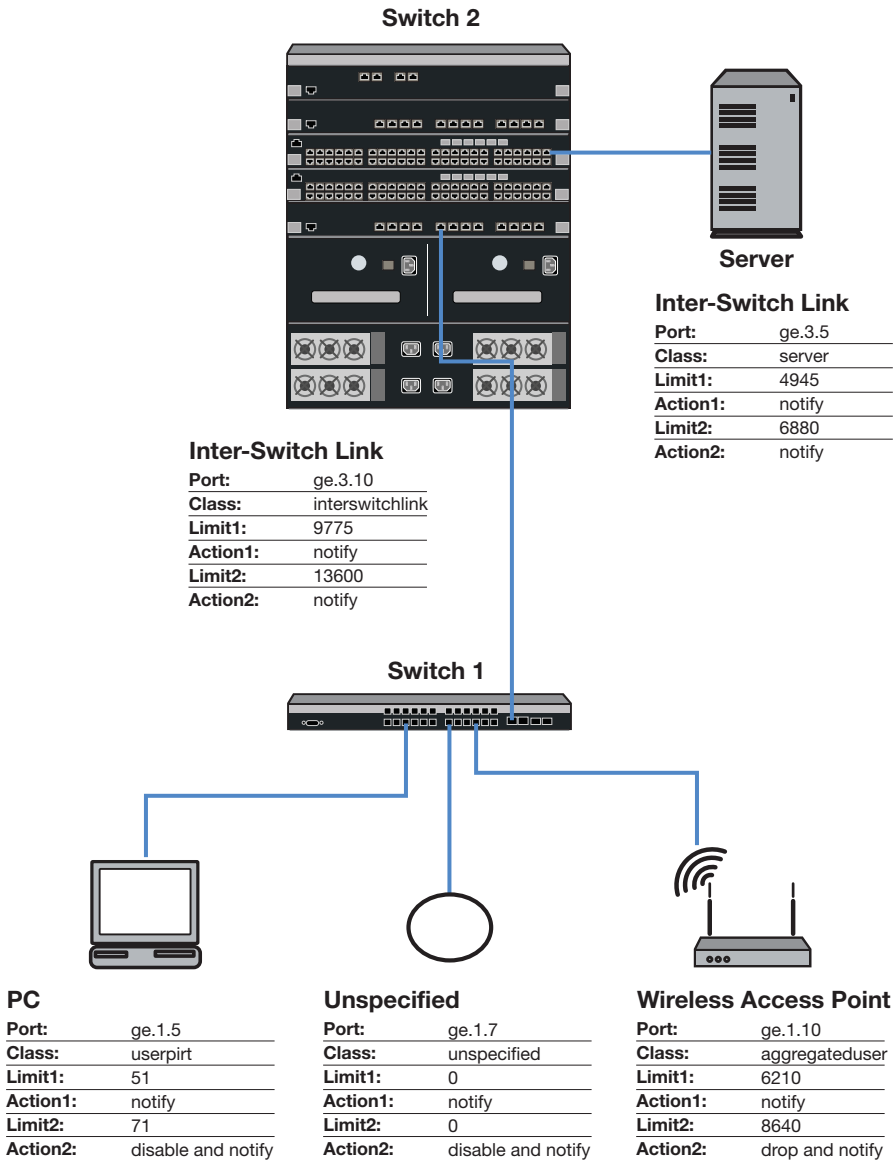
The configuration components used in this example are two N-Series chassis, a PC, a wireless access point, and a server.

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.

See Figure 28-1 on page 28-10 for an overview of this FST configuration example.

Figure 28-1 FST Configuration Example Overview



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 28-4 lists terms and definitions used in this link aggregation configuration discussion.

Table 28-4 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.
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.

Table 28-4 Flow Setup Throttling Terms and Definitions (continued)

Term	Definition
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. Configurable port types are: 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.

29

Access Control List Configuration

This document provides the following information about configuring IPv4 and IPv6 Access Control Lists (ACLs) on the Enterasys N-Series platforms.

Note: This chapter covers the configuration of both IPv4 and IPv6 ACLs. Unless specified, the discussion contents applies to both IPv4 and IPv6 ACLs. IPv4 and IPv6 ACLs are configured in separate configuration modes with separate command sets.

Using Access Control Lists (ACLs) in Your Network

ACLs allow the configuration of permit and denial of IPv4 and IPv6 packet forwarding based upon IP address, protocol, and port matching, depending upon the ACL type. The N-Series firmware supports configuration of both standard and extended ACLs. Standard ACLs allow the packet source IP address to be configured, while extended 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. 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.

Implementing ACLs

To implement an ACL on your network:

- Create the ACL
- Enter the rules and comments for this ACL that will determine which packets will be forwarded or not forwarded on the routing interface this ACL will be applied to
- Optionally manage your ACL 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 ACL to an host access interface

ACL Overview

This section describes ACL creation, rule entry, and application of the ACL to a routing interface required to implement an ACL, as well as, the features available for managing ACL rules and displaying ACLs.

Note: IPv6 support on the N-Series is currently limited to host access. IPv6 ACLs for purposes of host access are supported.

Creating an ACL

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 ACL can be copied to a non-existing ACL of the same IP type (IPv4 or IPv6). An existing ACL can be appended to the end of another existing ACL of the same IP type, but a standard ACL may not be appended to an extended ACL nor vice versa.

Upon creating the ACL, you are placed in the access-list configuration command mode where you can enter rules or comment entries for this ACL.

IPv4 ACL Creation Examples

The following example creates a standard IPv4 ACL with the access-list number **1** as its identifier:

```
N Chassis(rw-config)->ip access-list standard 1
N Chassis(rw-cfg-std-acl)->
```

The following example creates an extended IPv4 ACL with the access-list number **100** as its identifier:

```
N Chassis(rw-config)->ip access-list extended 100
N Chassis(rw-cfg-ext-acl)->
```

The following example creates a standard ACL with the name **ipv4acl1** as its identifier:

```
N Chassis(rw-config)->ip access-list standard ipv4acl1
N 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:

```
N Chassis(rw-config)->ipv6 access-list standard acl1
N Chassis(rw-cfg-ipv6-std-acl)->
```

The following example creates an extended IPv6 ACL with the access-list number **acl100** as its identifier:

```
N Chassis(rw-config)->ipv6 access-list extended 100
N Chassis(rw-cfg-ipv6-ext-acl)->
```

The following example creates a standard IPv6 ACL with the name **ipv6acl1** as its identifier:

```
N Chassis(rw-config)->ipv6 access-list standard ipv6acl1
N Chassis(rw-cfg-ipv6-std-acl)->
```

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 or an extended 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.

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 *Enterasys Matrix N-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.

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.

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.

Use the **permit** command to create a rule that forwards packets based upon the defined rule.

Use the **deny** command to create a rule that prevents the forwarding of packets 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:

```
N Chassis(rw-config)->ip access-list standard 1
N Chassis(rw-cfg-std-acl)->permit 10.0.0.1 0.0.255.255
N 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:

```
N Chassis(rw-config)->ip access-list extended 120
N Chassis(rw-cfg-ext-acl)->deny ip 20.0.0.1 0.0.255.255 any log
N Chassis(rw-cfg-ext-acl)->permit ip any any
N 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 --
N 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:

```
N Chassis(rw-config)->ipv6 access-list standard acl2
N Chassis(rw-cfg-ipv6-ext-acl)->permit 2001:1234:50:0:21f:45ff:fe3d:21be/64
N 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:

```
N Chassis(rw-config)->ipv6 access-list extended acl120
N Chassis(rw-cfg-ipv6-ext-acl)->permit ipv6 2001:1234:50:0:21f:45ff:fe3d:21aa/64
any
N Chassis(rw-cfg-ipv6-ext-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 ACL 120 and deletes and deletes entries 2 and 3:

```
N Chassis(rw-config)->ip access-list extended 120
N 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 --
N Chassis(rw-cfg-ext-acl)->delete from 2 to 3
N 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**:

```
N Chassis(rw-config)->ipv6 access-list standard acl2
N Chassis(rw-cfg-ipv6-std-acl)->delete from 10 to 12
N Chassis(rw-cfg-ipv6-std-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**:

```
N Chassis(rw-config)->ip access-list extended 121
N 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 --
N Chassis(rw-cfg-ext-acl)->move before 2 from 3 to 4
N 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**:

```
N Chassis(rw-config)->ipv6 access-list standard acl2
N Chassis(rw-cfg-ipv6-std-acl)->move before 5 from 10 to 12
N Chassis(rw-cfg-ipv6-std-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:

```
N Chassis(rw-config)->ip access-list extended 121
N 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 --
N Chassis(rw-cfg-ext-acl)->replace 1 deny ip 10.0.0.1 0.0.255.255 any
N 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 :

```
N Chassis(rw-config)->ipv6 access-list standard acl10
N Chassis(rw-cfg-ipv6-std-acl)->replace 1 permit any
N Chassis(rw-cfg-ipv6-std-acl)->
```

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:

```
N Chassis(rw-config)->ip access-list extended 121
N 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 --
N Chassis(rw-cfg-ext-acl)->insert before 2 deny ip 20.0.0.1 0.0.255.255 any
N 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**:

```
N Chassis(rw-config)->ipv6 access-list standard acl10
N Chassis(rw-cfg-ipv6-ext-acl)->insert before 10 permit host 2002:100::50
2001:100::100:25/64 traffic-class 6
N Chassis(rw-cfg-ipv6-ext-acl)->
```

Applying ACLs

Once you have defined an ACL, it can be applied per routing interface. An ACL can be applied to host access or an interface before it is created. The association of the name or number of the 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 to an interface and the **ipv6 access-group** command to apply an IPv6 access-list to an interface, in interface configuration command mode, specifying the access-list number or name 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 a routing interface.

The following example applies the extended ACL 121 to both the inbound and outbound direction on VLAN 2.

```
N Chassis(su-config)->interface vlan 2
N Chassis(su-config-intf-vlan.0.2)->ip access-group 121 in
N Chassis(su-config-intf-vlan.0.2)->ip access-group 121 out
N 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)
N 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:

```
N Chassis(su-config)->ipv6 access-list standard acl10
N Chassis(su-cfg-ipv6-std-acl)->permit fe80:0:0:0:21f:45ff:fe3d:21aa/64 log
N Chassis(su-cfg-ipv6-std-acl)->exit
N Chassis(su-config)->interface vlan 50
N Chassis(su-config-intf-vlan.0.50)->ipv6 access-group acl10 in
N Chassis(su-config-intf-vlan.0.50)-
```

Configuring ACLs

This section provides details for the configuration of ACLs on the N-Series products.

Important Notice

Extended ACL is an advanced routing feature that must be enabled with a license key. If you have purchased an advanced license key, and have enabled routing on the device, you must activate your license as described in the *Enterasys Matrix N-Series Configuration Guide* in order to enable the extended ACL command. If you wish to purchase an advanced routing license, contact Enterasys Networks Sales.

Procedure 29-1 describes how to create an IPv4 ACL and manage IPv4 ACLs at the ACL level.

Procedure 29-1 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*
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*

Procedure 29-1 Creating and Managing IPv4 and IPv6 ACLs (continued)

Step	Task	Command(s)
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 29-2 describes how to enter and manage standard ACL rules.

Procedure 29-2 Entering and Managing Standard IPv4 ACL Rules

Step	Task	Command(s)
1.	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**]
2.	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**]}
3.	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 29-3 describes how to enter and manage standard ACL rules.

Procedure 29-3 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 29-4 describes how to enter and manage extended IPv4 ACL rules.

Procedure 29-4 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 29-4 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 29-5 describes how to enter and manage extended IPv6 ACL rules.

Procedure 29-5 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 29-5 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 29-6 describes how to manage ACL rules.

Procedure 29-6 Managing IPv4 and IPv6 ACL Rules

Step	Task	Command(s)
1.	In IPv4 or IPv6 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 or IPv6 ACL configuration command mode, optionally, delete a preexisting ACL rule entry.	**delete** {*entry* | **from** *entry* **to** *entry*}
3.	In IPv4 or IPv6 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 or IPv6 ACL configuration command mode, optionally, enter a text comment as the next ACL entry.	**remark "***text***"**

Procedure 29-7 describes how to apply and display ACLs.

Procedure 29-7 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**}
2.	In configuration command mode, apply an 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**]]

Terms and Definitions

Table 29-1 lists terms and definitions used in this ACL configuration discussion.

Table 29-1 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 packets based upon one or more packet fields, such as source and destination IP address, and protocol.
entry	A member of an ACL that either permits or denies forwarding of the packet based upon one or more specified packet fields, or provides an ACL comment.
rule	An ACL entry that affects packet forwarding 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.

30

Route-Map Manager Configuration

This document describes the route-map manager feature and its configuration on Enterasys N-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

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.

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

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, 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 in 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 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 7, **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 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:

```
N Chassis(su-config)->probe ICMP-PBR icmp
N Chassis(su-config-probe)->faildetect count 5 interval 5
N Chassis(su-config-probe)->inservice
N Chassis(su-config-probe)->exit
N Chassis(su-config)->route-map probe 101.10.1.252 probe-name ICMP-PBR
N 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 session
...
N 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 *Enterasys Matrix N-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 N-Series products.

Table 30-1 lists route-map manager parameters and their default values.

Table 30-1 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

Table 30-1 Default Route-Map Manager Parameters (continued)

Parameter	Description	Default Value
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 30-1 describes how to configure a policy based route-map.

Procedure 30-1 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*]
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**}

Procedure 30-1 Configuring a Policy Based Route-Map (continued)

Step	Task	Command(s)
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 30-2 describes how to configure a redistribution route-map.

Procedure 30-2 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*]
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*

Procedure 30-2 Configuring a Redistribution Route-Map (continued)

Step	Task	Command(s)
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 30-2 describes how to configure an OSPF filter route-map.

Procedure 30-3 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*}

Procedure 30-3 Configuring a Filter Route-Map (continued)

Step	Task	Command(s)
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**

Table 30-2 describes how to display route-map manager information. Display commands can be entered in any command mode.

Table 30-2 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.

```
N Chassis(rw)->configure
N Chassis(rw-config)->ip access-list extended 101
N Chassis(rw-cfg-ext-acl)->permit ip 60.10.0.0 0.0.255.255 host 50.10.0.1
N Chassis(rw-cfg-ext-acl)->permit ip 60.10.0.0 0.0.255.255 host 50.10.0.2
N Chassis(rw-cfg-ext-acl)->deny ip any any
```

```
N 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 --
N Chassis(rw-cfg-ext-acl)->exit
N Chassis(rw-config)->route-map policy rmP1 permit 10
N Chassis(rw-config-route-map-pbr)->match ip address 101
N Chassis(rw-config-route-map-pbr)->set next-hop 30.10.0.10 30.10.0.20 30.10.0.30
N Chassis(rw-config-route-map-pbr)->exit
N 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
N Chassis(rw-config)->route-map probe 30.10.0.10 default
N Chassis(rw-config)->route-map probe 30.10.0.20 default
N Chassis(rw-config)->route-map probe 30.10.0.30 default
N Chassis(rw-config)->interface vlan 110
N Chassis(rw-config-intf-vlan.0.110)->ip policy priority only
N Chassis(rw-config-intf-vlan.0.110)->ip policy load-policy round-robin
N Chassis(rw-config-intf-vlan.0.110)->ip policy route-map rmP1
N 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
N Chassis(rw-config-intf-vlan.0.110)->exit
N 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

```
N Chassis(rw)->configure
N Chassis(rw-config)->ip access-list standard OSPF
N Chassis(rw-cfg-std-acl)->permit 40.0.0.0 0.0.0.255
N Chassis(rw-cfg-std-acl)->permit 40.0.10.0 0.0.0.255
N 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 --
N Chassis(rw-cfg-std-acl)->exit
N Chassis(rw-config)->route-map redistribution rmR1 permit 10
N Chassis(rw-config-route-map)->match ip address OSPF
N Chassis(rw-config-route-map)->set tag 65432
N Chassis(rw-config-route-map)->exit
N Chassis(rw-config)->show route-map rmR1
 route-map redistribution rmR1 permit 10
  match ip address OSPF
  set tag 65432
N Chassis(rw-config)->router ospf 1
N Chassis(rw-config-ospf-1)->redistribute rip route-map rmR1
N Chassis(rw-config-ospf-1)->exit
N Chassis(rw-config)->
```

Terms and Definitions

Table 30-3 lists terms and definitions used in this route-map manager configuration discussion.

Table 30-3 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.

31

Quality of Service (QoS) Configuration

This chapter describes the QoS feature as it is implemented on the Enterasys N-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 Enterasys 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

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 Enterasys recommends for configuring QoS are:

- Understand your network flows using NetFlow. See Chapter 18, **NetFlow Configuration** for NetFlow configuration details.
- Associate the flows on your network with a well defined role using Enterasys policy. See Chapter 14, **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.

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) 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:

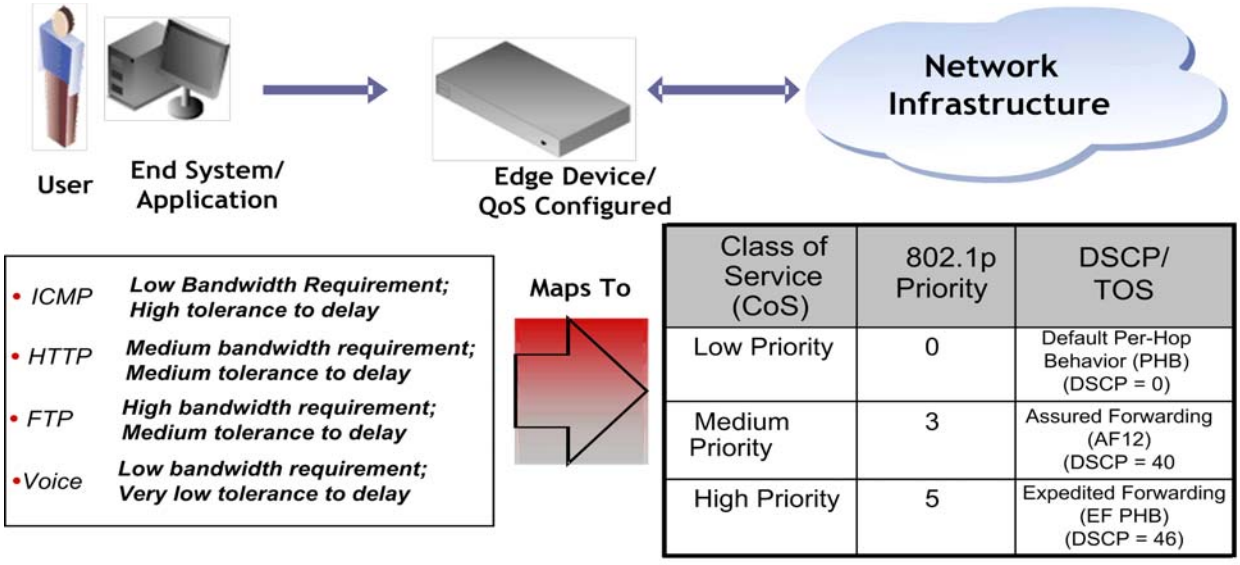
- An 8-bit field with values 0–255
- Supported in layer 3 only
- Also referred to as the Differentiated Services Code Point (DSCP) when limited to the lower 5 bits of the field

Figure 31-1 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 31-1 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.

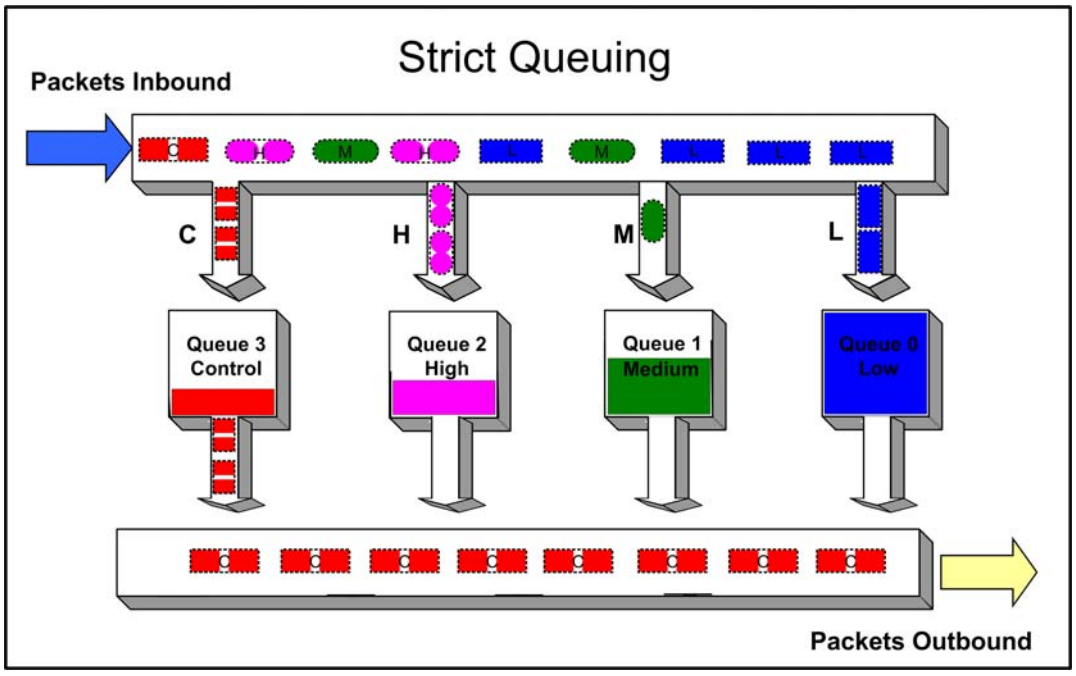
Preferential Queue Treatment for Packet Forwarding

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 31-2. 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 31-2 Strict Priority Queuing Packet Behavior

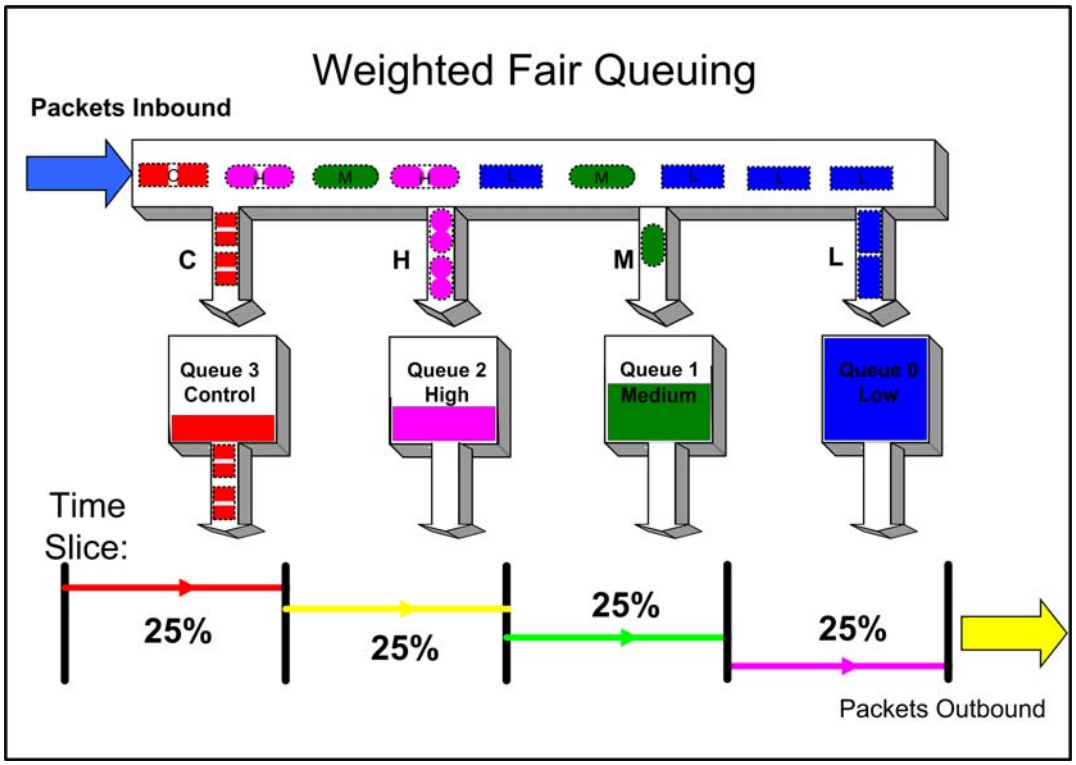


Weighted Fair Queuing

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 31-3 on page 31-5 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.

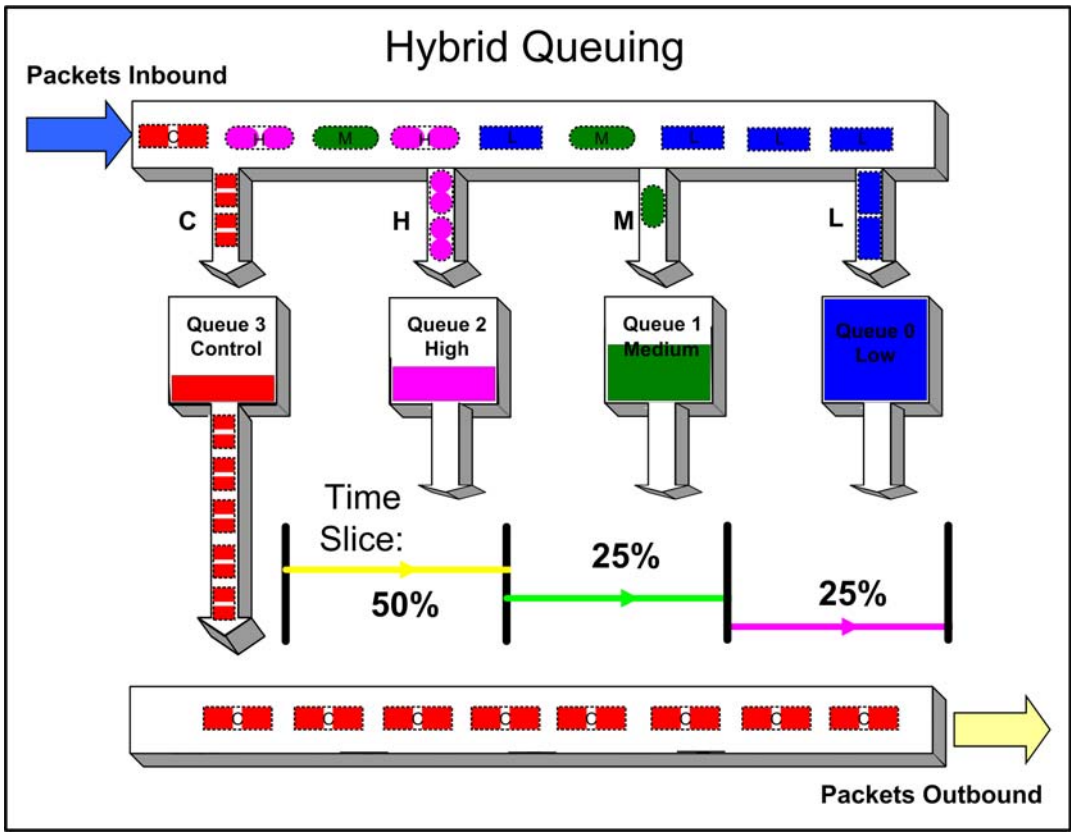
Figure 31-3 Weighted Fair Queuing Packet Behavior



Hybrid Queuing

Hybrid queuing combines the properties of both strict priority and weighted fair queuing. Figure 31-4 on page 31-6, 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 31-4 Hybrid Queuing Packet Behavior

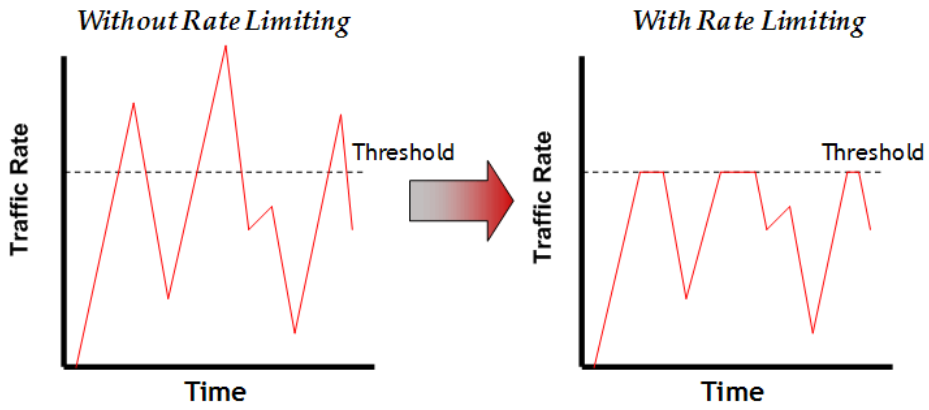


Rate Limiting

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 31-5 on page 31-7 illustrates how bursty traffic is clipped above the assigned threshold with rate limiting applied.

Figure 31-5 Rate Limiting Clipping Behavior

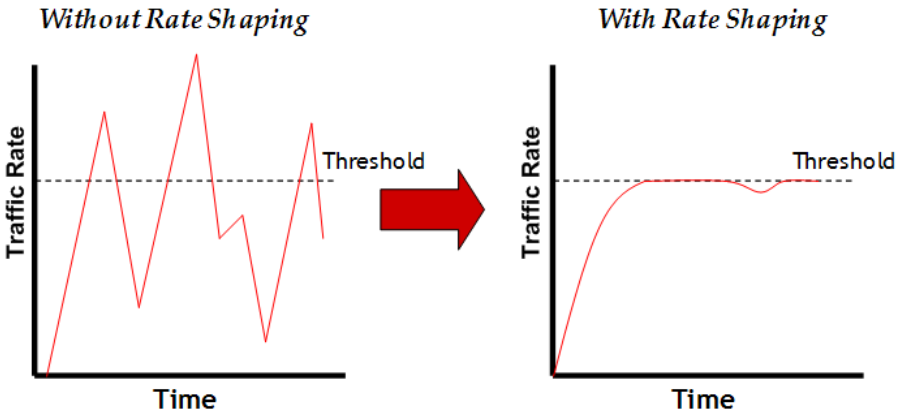


Rate Shaping

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 31-6 illustrates how bursty traffic is smoothed out when it bursts above the assigned threshold with rate shaping applied.

Figure 31-6 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 N-Series

This section discusses the six components for configuring QoS and displaying QoS status on an N-Series switch router:

CoS Port-Type: Based upon the transmit queue (TxQ), Inbound Rate Limiting (IRL), 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 and IRL 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 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 Enterasys NetSight Policy Manager as an alternative to CLI for configuring policy-based CoS on Enterasys Series devices.

A policy discussion is outside the scope of this document and will be limited to the relevant configuration example commands. See Chapter 14, **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 31-1 on page 31-19 identifies these mapping relationships.

See "Determining CoS Port-Type" on page 31-9 and "Configuring CoS Port Groups" on page 31-10 for a port group discussion.

Determining CoS Port-Type

Based on physical capability, all physical ports belong to one of three port-types. The importance of this port-type distinction lies in the resources available for transmit queue, inbound rate limiting, and flood control CoS features. The nomenclature distinguishes the types as port type 0 (16 queues, port type 1 (4 queues), and port type 2 (8 queues).

TxQ

Port type 0 supports sixteen transmit queues, while type 1 supports four. Port type 2 supports 8 queues. 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:

```
N 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
                                   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     N-Series 16Q    64/16     perc       ge.1.1-60          ge.1.1-60
                                 Kbps
                                 Mbps
                                 Gbps

1     N-Series 4Q     32/4      perc       None               None
                                 Kbps
                                 Mbps
                                 Gbps

2     N-Series 11Q     32/8     perc        None               None
                                 Kbps
                                 Mbps
                                 Gbps
```

IRL

Type 0 supports 32 Inbound Rate Limiters. Type 1 supports 8 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:

```
N 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
                                   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      N-Series 32     32 irl   perc       ge.1.1-60         ge.1.1-60
                                pps
                                Kbps
                                Mbps
                                Gbps

1      N-Series 8       8 irl   perc       None              None
                                pps
                                Kbps
                                Mbps
                                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:

```
N 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
                                   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      N-Series Flo     3 fld   perc       ge.1.1-60         ge.1.1-60
                                pps
                                Kbps
                                Mbps
                                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, and flood control CoS features and are identified as port group 0 and port type 0, 1, or 2 and are indexed as **0.0**, **0.1**, or **0.2** 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 0.2 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:

```
N 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  :N-Series 16Q
Port Group       :0
Port Type        :0
Assigned Ports   :none
Arbiter Mode     :Strict
Slices/queue     :Q [ 0]:   0  Q [ 1]:   0  Q [ 2]:   0  Q [ 3]:   0
                 :Q [ 4]:   0  Q [ 5]:   0  Q [ 6]:   0  Q [ 7]:   0
                 :Q [ 8]:   0  Q [ 9]:   0  Q [10]:   0  Q [11]:   0
                 :Q [12]:   0  Q [13]:   0  Q [14]:   0  Q [15]:  64
Percentage/queue :Q [ 0]:   0% Q [ 1]:   0% Q [ 2]:   0% Q [ 3]:   0%
                 :Q [ 4]:   0% Q [ 5]:   0% Q [ 6]:   0% Q [ 7]:   0%
                 :Q [ 8]:   0% Q [ 9]:   0% Q [10]:   0% Q [11]:   0%
                 :Q [12]:   0% Q [13]:   0% Q [14]:   0% Q [15]: 100%
--------------------------------------------------------------------------
Port Group Name  :N-Series 4Q
Port Group       :0
Port Type        :1
Assigned Ports   :ge.1.1-60
Arbiter Mode     :Strict
Slices/queue     :Q [ 0]:   0  Q [ 1]:   0  Q [ 2]:   0  Q [ 3]:  32
Percentage/queue :Q [ 0]:   0% Q [ 1]:   0% Q [ 2]:   0% Q [ 3]: 100%
--------------------------------------------------------------------------
Port Group Name  :N-Series 8Q
Port Group       :0
Port Type        :2
Assigned Ports   :none
Arbiter Mode     :Strict
Slices/queue     :Q [ 0]:   0  Q [ 1]:   0  Q [ 2]:   0  Q [ 3]:   0
                 :Q [ 4]:   0  Q [ 5]:   0  Q [ 6]:   0  Q [ 7]:  32
```

```
Percentage/queue :Q [ 0]:   0% Q [ 1]:   0% Q [ 2]:   0% Q [ 3]:   0%
                 :Q [ 4]:   0% Q [ 5]:   0% Q [ 6]:   0% Q [ 7]: 100%
-------------------------------------------------------------------
```

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 (arb-percentage and arb-slices are not supported on the DFE Gold module)

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 0.2 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 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:

```
N Chassis(rw)->show cos port-config irl

Inbound Rate Limiting Port Configuration Entries
-------------------------------------------------------------------
Port Group Name  :N-Series 32 IRL
Port Group       :0
Port Type        :0
Assigned Ports   :ge.1.1-60
-------------------------------------------------------------------
Port Group Name  :N-Series 8 IRL
Port Group       :0
Port Type        :1
Assigned Ports   :none
-------------------------------------------------------------------
Port Group Name  :N-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: 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 0.2 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 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:

```
N Chassis(rw)->show cos port-config flood-ctrl

Inbound Rate Limiting Port Configuration Entries
------------------------------------------------------------------------
Port Group Name  :N-Series 32 IRL
Port Group       :0
Port Type        :0
Assigned Ports   :ge.1.1-60
------------------------------------------------------------------------
Port Group Name  :N/A 8 IRL
Port Group       :0
Port Type        :1
Assigned Ports   :none
------------------------------------------------------------------------
Port Group Name  :N/A 24 IRL
Port Group       :0
Port Type        :2
Assigned Ports   :none
------------------------------------------------------------------------
```

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 31-7 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:

```
N Chassis(rw)->show cos port-resource txq

'?' after the rate value indicates an invalid rate value

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 31-6 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:

```
N 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 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:

```
N 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 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:

```
N 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:

```
N 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

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 31-16 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 31-17 for an IRL reference configuration 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:

```
N 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           *        2      *      *      *         Disabled
2          2           *        4      *      *      *         Disabled
3          3           *        6      *      *      *         Disabled
4          4           *        8      *      *      *         Disabled
5          5           *        10     *      *      *         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:

```
N 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
```

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 31-1 provides a CLI flow summary of each step in the configuration flow along with the show commands to verify the configuration.

Procedure 31-1 Class of Service CLI Configuration Command Summary

Step	Task	Command(s)
1.	Inspect the TxQs, IRL, 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 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*
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**

Procedure 31-1 Class of Service CLI Configuration Command Summary (continued)

Step	Task	Command(s)
4.	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**
5.	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*
6.	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*
7.	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*
8.	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*
9.	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*
10.	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*] [**flood-ctrl** *flood-ctrl*] **show cos settings**
11.	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 N-Series edge switches and a N-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: Enterasys 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.

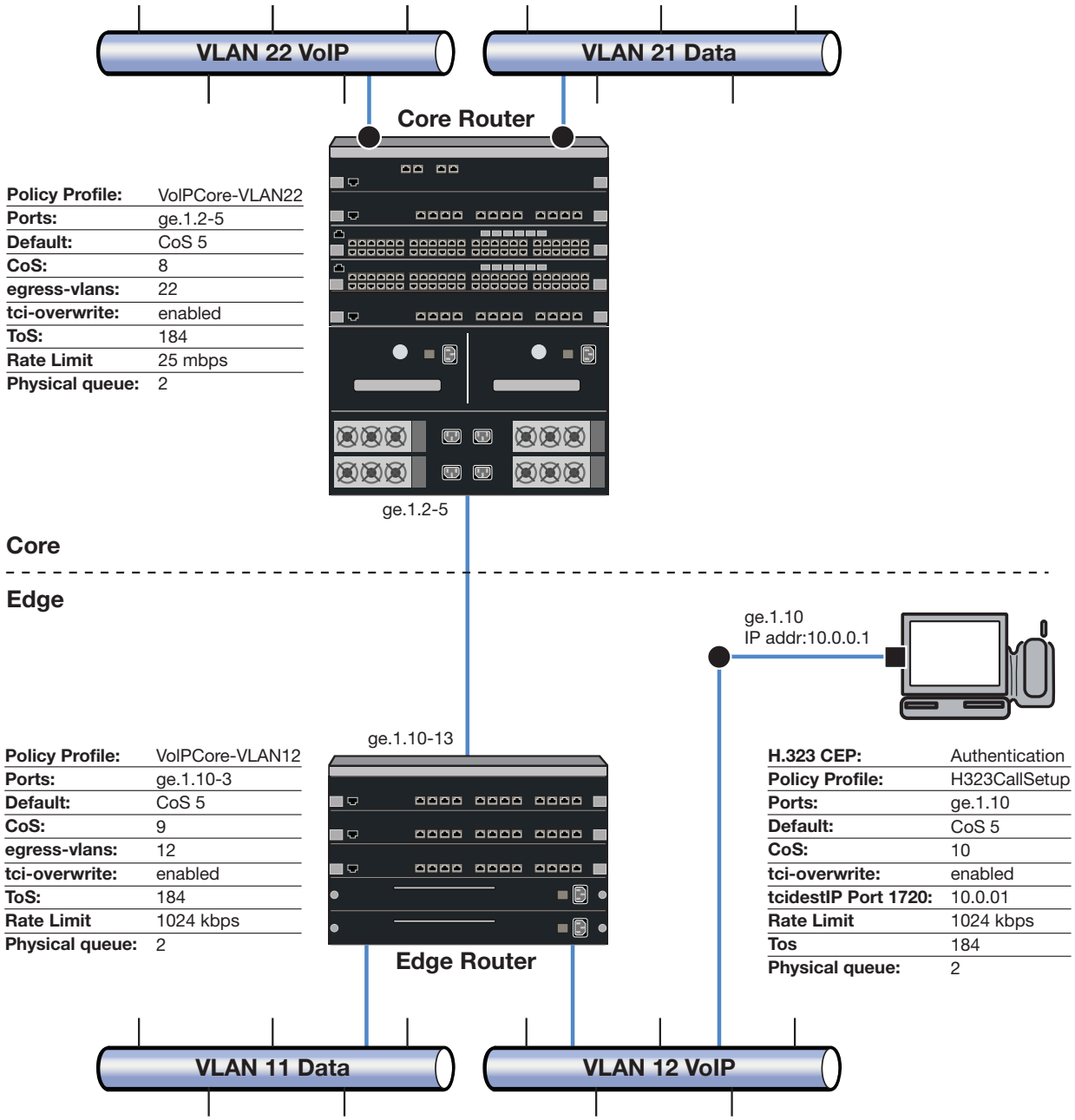
To simplify this discussion of the configuration process, this example is limited to the VoIP configuration context. Table 31-1 provides a set of sample values for priority, IRL, and transmit queue across a number of real world traffic types. This table can be used as an aid in thinking about how you 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 31-1 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 31-7 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 31-7 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 N-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 N-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 N-Series Router 1

```
N Chassis(rw)->set policy profile 1 name VoIPCore-VLAN22 cos 5 egress-vlans 22
tci-overwrite enable
N Chassis(rw)->set policy rule admin-profile vlantag 22 mask 12 port-string
ge.1.2-5 admin-pid 1
N Chassis(rw)->set policy rule 1 vlantag 22 mask 12 vlan 22 cos 8
N Chassis(rw)->set cos port-resource irl 1.1 0 unit mbps rate 25
N Chassis(rw)->set cos reference irl 1.1 8 rate-limit 0
N Chassis(rw)->set cos 8 priority 5 tos-value 184.0 txq-reference 8 irl-reference
0
N Chassis(rw)->set cos state enable
```

Summary of Command Line Input for N-Series Switch 1

```
N Chassis(rw)->set policy profile 1 name VoIPEdge-VLAN12 cos 5 egress-vlans 12
tci-overwrite enable
N Chassis(rw)->set policy rule admin-profile vlantag 12 mask 12 port-string
ge.1.10-13 admin-pid 1
N Chassis(rw)->set policy rule 1 vlantag 12 mask 12 vlan 12 cos 9
N Chassis(rw)->set cos port-resource irl 2.1 0 unit mbps rate 1
N Chassis(rw)->set cos reference irl 2.1 9 rate-limit 0
```

```
N Chassis(rw)->set cos 9 priority 5 tos-value 184.0 txq-reference 8 irl-reference
1
N Chassis(rw)->set policy profile 2 name H323CallSetup cos 5 tci-overwrite enable
N Chassis(rw)->set policy rule admin-profile port ge.1.10 mask 16 port-string
ge.1.10 admin-pid 2
N Chassis(rw)->set policy rule 1 tcpdestportIP 1720:10.0.0.1 cos 10 port-string
ge.1.10
N Chassis(rw)->set cos port-resource irl 3.1 2 unit pps rate 5
N Chassis(rw)->set cos reference irl 3.1 10 rate-limit 1
N Chassis(rw)->set cos 10 priority 5 tos-value 184.0 txq-reference 8 irl-reference
2
N Chassis(rw)->set cos state enable
```

Terms and Definitions

Table 31-2 lists terms and definitions used in this Quality of Service configuration discussion.

Table 31-2 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 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, 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.

32

RADIUS Snooping Configuration

This document provides the following information about configuring RADIUS-Snooping on the Enterasys N-Series platforms.

Using RADIUS-Snooping in Your Network

RADIUS-Snooping (RS) is one of the Enterasys MultiAuth suite of authentication methods. See Chapter 33, **Authentication Configuration** for a detailed discussion of the other authentication methods supported by the N-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 Enterasys 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 Enterasys MultiAuth framework, with the exception that RS can not force a reauthentication event; it can only timeout the session.

RADIUS-Snooping allows the Enterasys N-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.

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

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 33-5 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 32-1 RADIUS-Snooping Overview

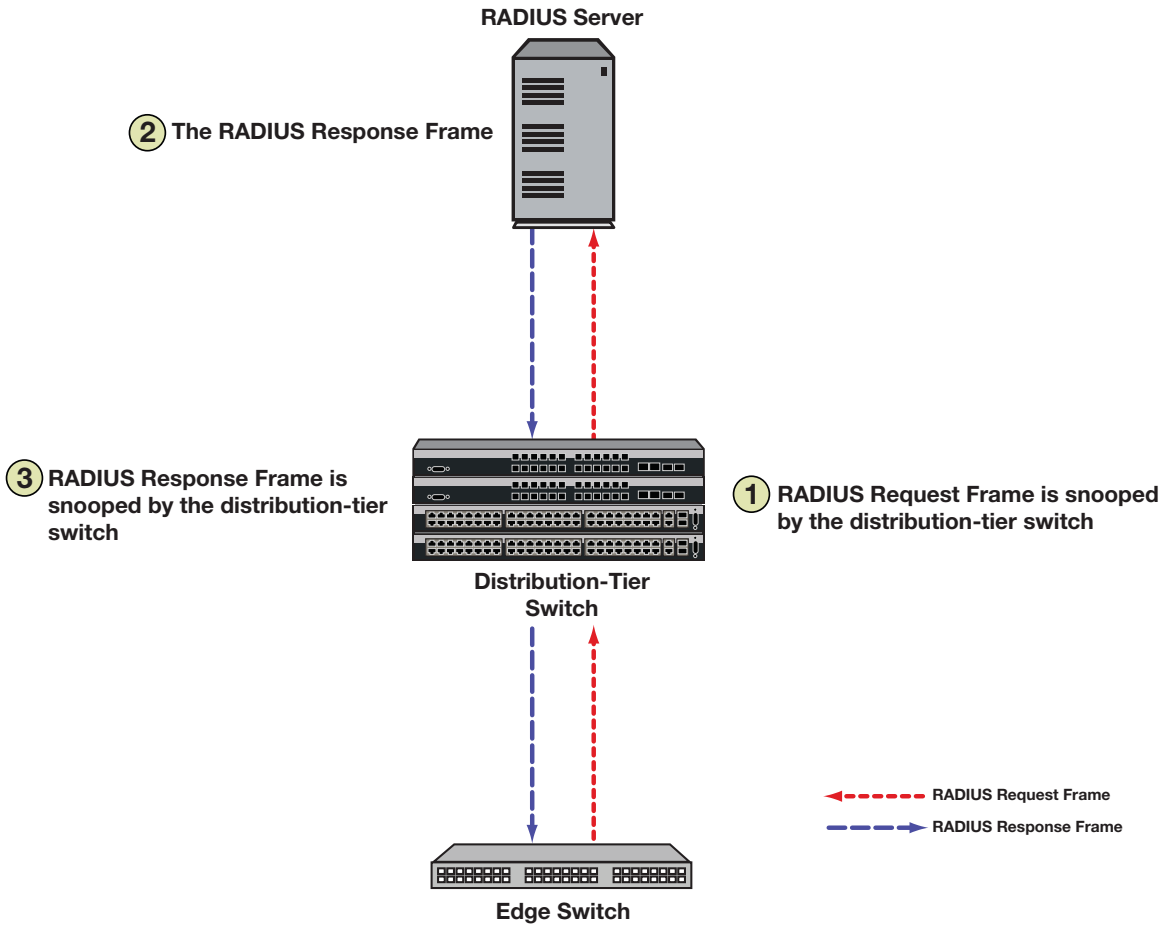


Figure 32-1 illustrates the RADIUS request frame and RADIUS response frame paths. As the RADIUS request frame from the RADIUS client edge device transits the distribution-tier switch, it is snooped. An RS session is created on the distribution-tier switch, if:

- 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.

Configuring RADIUS-Snooping

This section provides details for the configuration of RADIUS-Snooping on the N-Series products.

Table 32-1 lists RS parameters and their default values.

Table 32-1 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
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 32-1 describes how to configure RADIUS-Snooping on the distribution-tier switch.

Procedure 32-1 RADIUS-Snooping Configuration

Step	Task	Command(s)
1.	Globally enable MultiAuth for **multi** mode.	**set multiauth mode multi**

Procedure 32-1 RADIUS-Snooping Configuration

Step	Task	Command(s)
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.	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*]
5.	Configure RADIUS-Snooping flow table index entries.	**set radius-snooping flow** *index* {*client-IP-Address server-IP-Address* {*port* } [*secret*]
6.	Optionally modify the RADIUS-Snooping timeout setting.	**set radius-snooping timeout** *seconds*

Managing RADIUS-Snooping

Table 32-2 describes how to manage RADIUS-Snooping on the distribution-tier switch.

Table 32-2 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 32-3 describes how to display RADIUS-Snooping statistics.

Table 32-3 Displaying RADIUS-Snooping Statistics

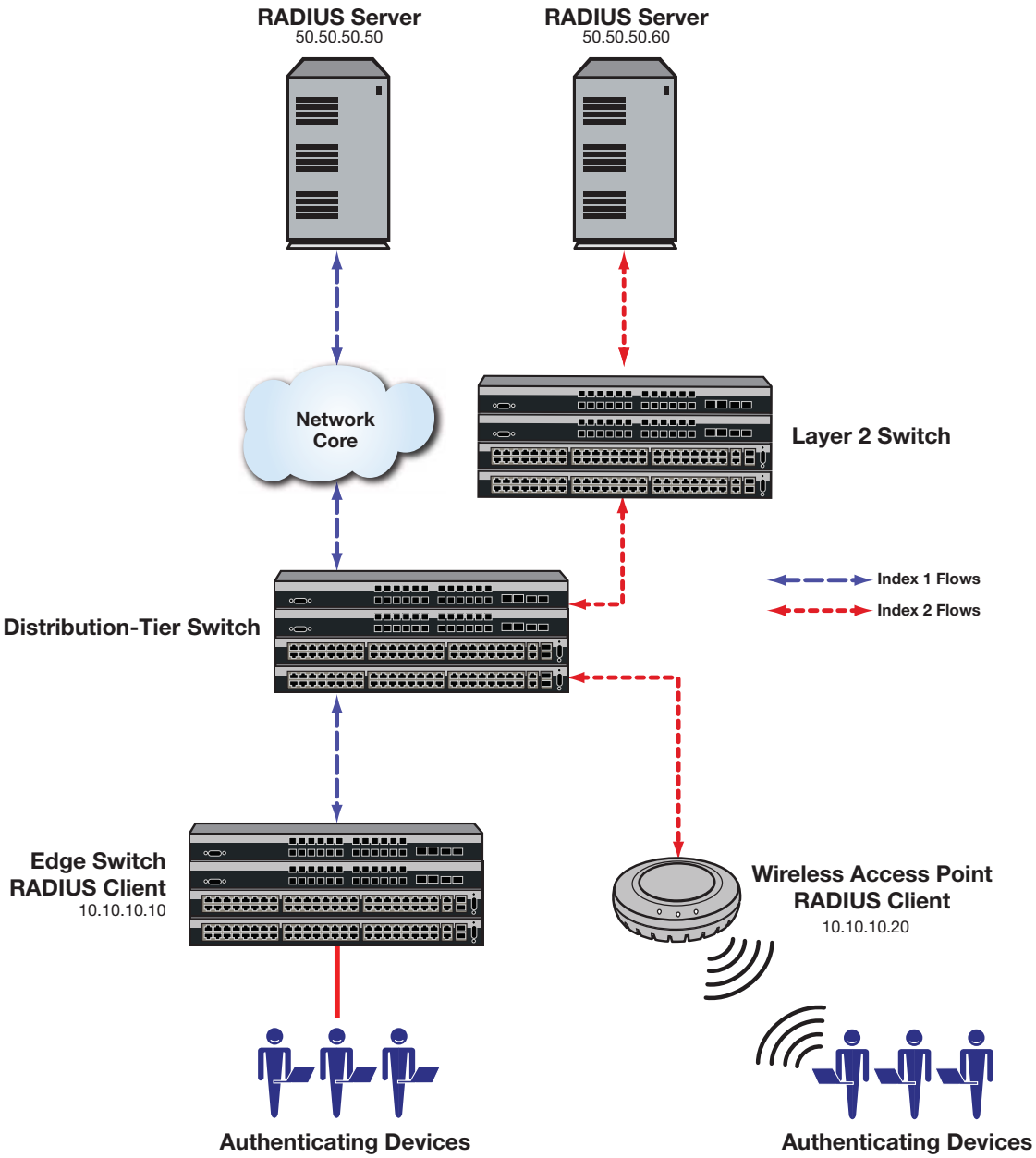
To display a general overview of the global RS status.	**show radius-snooping**
To display the RS status for the specified port.	**show radius-snooping port** *port-string*
To display information for all or the specified flow index entry.	**show radius-snooping flow** {*index* | **all**}
To display a 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 32-2 for an illustration of the example setup.

Figure 32-2 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

```
N Chassis(su)->set multiauth mode multi
```

Set the MultiAuth authentication mode for each port

```
N Chassis(su)->set multiauth port mode auth-opt ge.1.5-10,15-24
```

Enable RS on this system:

```
N Chassis(su)->set radius-snooping enable
```

Enable RS and set configuration for ports on this system

```
N Chassis(su)->set radius-snooping port enable drop enabled authallocated 256
ge.1.5-10
N Chassis(su)->set radius-snooping port enable drop enabled authallocated 256
ge.1.15-24
```

Configure RS flow table entries

```
N Chassis(su)->set radius-snooping flow 1 10.10.10.10 50.50.50.50 1812 mysecret
N Chassis(su)->set radius-snooping flow 2 10.10.10.20 50.50.50.60 1812 mysecret
```

Configure RS timeout for this system

```
N Chassis(su)->set radius-snooping timeout 15
```

Managing RADIUS-Snooping on the Distribution-tier Switch

Terminate an active session on port ge.1.15:

```
N Chassis(su)->set radius-snooping initialize port ge.1.15
```

Reset all RS configuration to its default value:

```
N Chassis(su)->clear radius-snooping all
```

Clear entry index 2 from the RS flow table:

```
N Chassis(su)->clear radius-snooping flow 2
```

This completes the RADIUS-Snooping configuration example.

Terms and Definitions

Table 32-4 lists terms and definitions used in this RADIUS-Snooping configuration discussion.

Table 32-4 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 Enterasys N-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.

33

Authentication Configuration

This document provides the following information about configuring user authentication on the Enterasys N-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. Enterasys 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 Enterasys switch products support the following five authentication methods:

- IEEE 802.1x
- MAC-based Authentication (MAC)
- Port Web Authentication (PWA)
- Convergence End Point (CEP)
- RADIUS Snooping

Enterasys switch products support the configuration of up to three 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 or granted on a limited basis. 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.

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

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 Enterasys 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.

MAC-Based Authentication (MAC)

MAC-based authentication (MAC) authenticates a device using the source MAC address of received packets. The authenticator sends the authentication server a source MAC address as the user name and a password that you configure on the switch. 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.

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.

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.
- **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.

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:

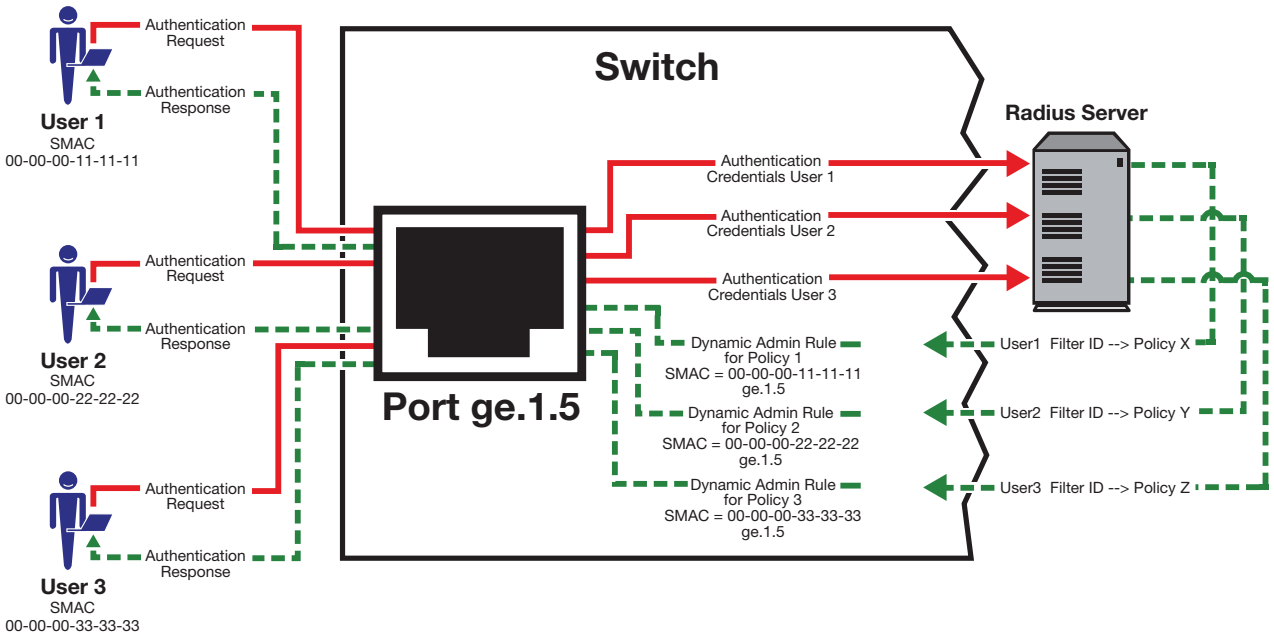
- A policy traffic profile, based on the user account's RADIUS Filter-ID configuration
- A base VLAN-ID, based on the RFC 3580 tunnel attributes configuration, also known as dynamic VLAN assignment

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 information in the *Enterasys Matrix N-Series CLI Reference* for the number of supported users per DFE 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 33-1 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 33-8 for Filter-ID information). The policy specified in the Filter-ID is then applied to the user. See section "RFC 3580" on page 33-8 for information on dynamic VLAN assignment and tunnel attribute configuration.

Figure 33-1 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.

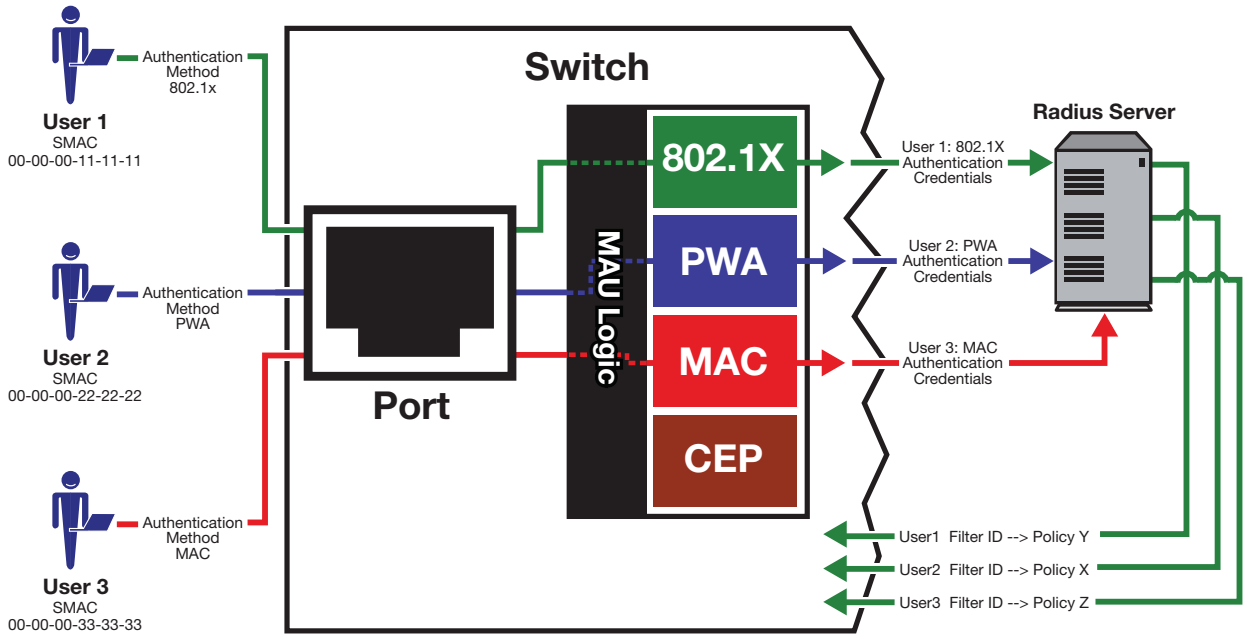
Strict mode is the appropriate mode when authenticating a single 802.1X user. All traffic on the port receives the same policy in strict mode. When authenticating PWA, CEP, or MAC, you must use MultiAuth authentication, whether authenticating a single or multiple supplicants.

MultiAuth authentication supports the simultaneous configuration of up to three 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. See “Setting MultiAuth Authentication Precedence” on page 33-20 for authentication method precedence details.

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.

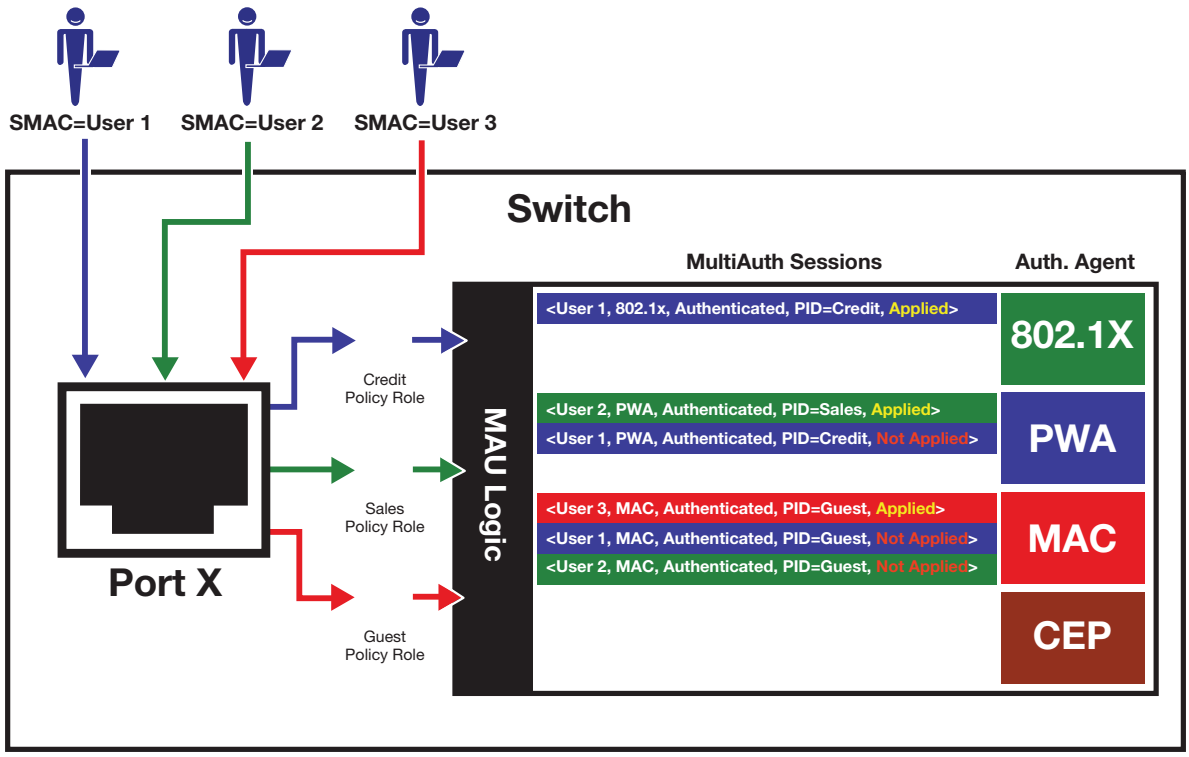
In Figure 33-2, multiple users are authenticated on a single port each with a different authentication method. In this case, each user on a single port successfully authenticates with a different authentication type. The authentication method is included in the authentication credentials sent to the 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 33-2 Authenticating Multiple Users With Different Methods on a Single Port



In Figure 33-3, 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 authentication methods are listed from top to bottom in order of precedence (the default order is displayed). 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 33-3 Selecting Authentication Method When Multiple Methods are Validated



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 Enterasys 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.

RFC 3580

Enterasys 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 Enterasys 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 Enterasys 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.

The N-Series supports RFC 3580 RADIUS VLAN Tunnel attributes starting in firmware release 5.31.xx.

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 33-8 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 33-1 lists Authentication parameters and their default values.

Table 33-1 Default Authentication Parameters

Parameter	Description	Default Value
cep port	Enables or disables CEP for the specified port.	Disabled.

Table 33-1 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: 802.1x, PWA, MAC, CEP.
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.
radius	Enable or disable RADIUS on this device.	Disabled.
radius accounting	Enables or disables RADIUS accounting for this device.	Disabled.
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.

Table 33-1 Default Authentication Parameters (continued)

Parameter	Description	Default Value
radius retries	Specifies the number of times a switch will try to establish with the authentication server.	3.
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 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 33-1 describes how to configure IEEE 802.1x on an authenticator switch port. Unspecified parameters use their default values.

Procedure 33-1 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. **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 *Enterasys Matrix N-Series CLI Reference* for a listing of parameter options that come with this command.	**set dot1x auth-config authcontrolled-portcontrol** {**auto** | **forced-auth** | **forced-unauth**}
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.	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 33-2 describes how to configure MAC-based authentication. Unspecified parameters use their default values.

Procedure 33-2 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.	Enable or disable MAC authentication globally on the device. By default, MAC authentication is globally disabled on the device.	**set macauthentication** {**enable** | **disable**}
5.	Set the MultiAuth mode.	**set multiauth mode multi**
6.	Display MAC authentication configuration or status of active sessions.	**show macauthentication** **show macauthentication session**
7.	If a session or port requires reinitialization, reinitialize a specific MAC session or port.	**set macauthentication macinitialize** *mac-address* **set macauthentication portinitialize** *port-string*
8.	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
- Enabling PWA on the port
- Globally enabling PWA on the switch
- Setting the authentication mode

Procedure 33-3 describes how to configure PWA authentication. Unspecified parameters use their default values.

Procedure 33-3 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.	Set the MultiAuth mode.	**set multiauth mode multi**
6.	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 33-2 describes how to optionally enable guest networking privileges.

Table 33-2 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 33-4 describes the creation of a CEP detection group.

Procedure 33-4 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 33-5 describes the steps to configure CEP.

Procedure 33-5 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** | **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.	Set the MultiAuth mode.	**set multiauth mode multi**
8.	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 33-6 describes setting the MultiAuth idle and session timeout for CEP.

Procedure 33-6 DNS and DHCP Spoofing Configuration

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 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 MAC, PWA, and CEP authentication.

Procedure 33-7 describes setting the MultiAuth authentication mode.

Procedure 33-7 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: 802.1x, PWA, MAC, and CEP. 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 PWA and MAC, the new precedence order will be PWA, MAC, 802.1x, and CEP.

Given the default order of precedence (802.1x, PWA, MAC, and CEP), 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.

Procedure 33-8 describes setting the order for MultiAuth authentication precedence.

Procedure 33-8 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** {[**dot1x**] [**mac**] [**pwa**] [**cep**] [**radius-snooping**]}
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 33-9 describes setting the MultiAuth authentication port and maximum user properties.

Procedure 33-9 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 33-10 describes setting the MultiAuth authentication timers.

Procedure 33-10 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*
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 33-11 describes setting the MultiAuth authentication traps.

Procedure 33-11 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**}}

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 33-3 describes displaying of MultiAuth authentication settings and statistics.

Table 33-3 MultiAuth Authentication Settings and Statistics Display

Task	Command(s)
Display system-wide MultiAuth authentication values.	**show multiauth**
Display MultiAuth authentication counters.	**show multiauth counters** [[**cep** | **dot1x** | **mac** | **pwa** | **radius-snooping**] [**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** {**dot1x** | **mac** | **pwa** | **cep** | **radius-snooping**}] [**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 33-12 describes setting VLAN authorization configuration.

Procedure 33-12 VLAN Authorization Configuration

Step	Task	Command(s)
1.	Enable or disable VLAN authorization both globally and per port.	**set vlanauthorization** {**enable** | **disable**}
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 33-13 describes setting dynamic policy profile assignment and invalid policy action configuration.

Procedure 33-13 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**

Procedure 33-13 Policy Profile Assignment and Invalid Action Configuration (continued)

Step	Task	Command(s)
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 N-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 33-14 describes authentication server configuration.

Procedure 33-14 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 the number of seconds the switch will wait before retrying authentication server establishment.	**set radius timeout** *timeout*
3.	Optionally set the number of retries that will occur before the switch declares an authentication server down.	**set radius retries** *retries*

Procedure 33-14 Authentication Server Configuration (continued)

Step	Task	Command(s)
4.	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**}
5.	Globally enable or disable RADIUS on the switch.	**set radius** {**enable | disable**}
6.	Reset the specified RADIUS setting to its default value.	**clear radius** {[**state**] [**retries**] [**timeout**] [**server** [*index* | **all**] [**realm** {*index* | **all**}]
7.	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 33-15 describes RADIUS accounting configuration.

Procedure 33-15 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**}

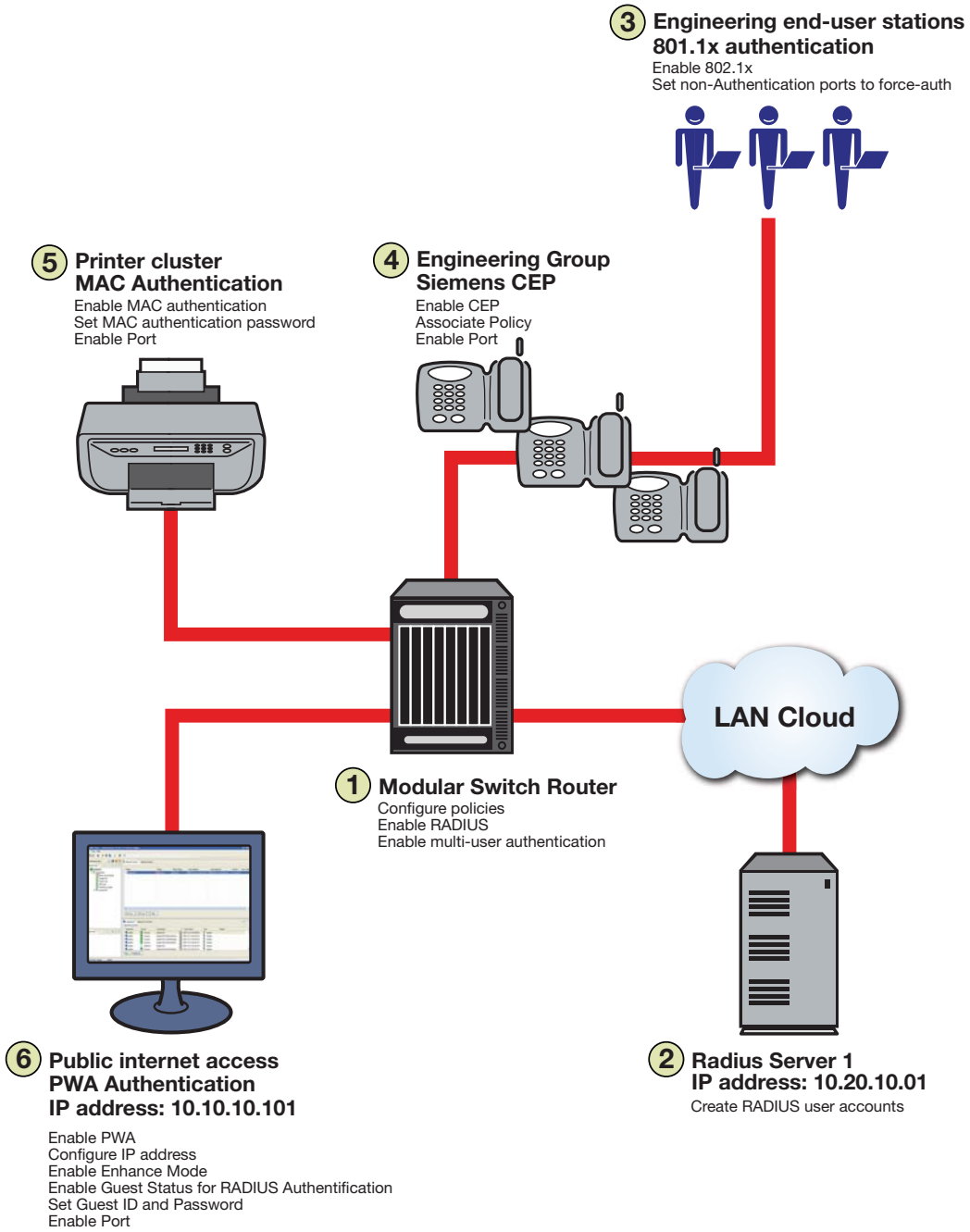
Procedure 33-15 RADIUS Accounting Configuration (continued)

Step	Task	Command(s)
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 the four supported authentication types being used in an engineering group: an end-user station, an IP phone, a printer cluster, and public internet access.

Figure 33-4 Authentication Configuration Example Overview



Our configuration example consists of the following steps as shown in Figure 33-4 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.

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 MAC-based, PWA, or CEP authentication 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.

```
N Chassis(rw)->set multiauth mode multi
N Chassis(rw)->set multiauth port mode force-auth ge.1.5-7
N Chassis(rw)->set multiauth port numusers 6 ge.1.5-7
N Chassis(rw)->set multiauth port mode force-auth ge.1.19-24
N Chassis(rw)->set multiauth port numusers 6 ge.1.19-24
```

- Enables MultiAuth authentication system and module traps for the N-Series configuration.

```
N Chassis(rw)->set multiauth trap system enabled
N 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.

```
N Chassis(rw)->set radius server 1 10.20.10.01
N 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 802.1x 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 33-8 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**.

```
N Chassis(rw)->set dot1x enable
N Chassis(rw)->set dot1x auth-config authcontrolled-portcontrol forced-auth
ge.1.5
N Chassis(rw)->set dot1x auth-config authcontrolled-portcontrol forced-auth
ge.1.19
N Chassis(rw)->set dot1x auth-config authcontrolled-portcontrol forced-auth
ge.2.24
```

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 31, **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.

```
N Chassis(rw)->set cep enable
N Chassis(rw)->set cep policy siemens 9
N Chassis(rw)->set cep port ge.1.16-18 siemens enable
```

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**

With the authentication server configured with a RADIUS account for each printer, and the printer policy preconfigured, enter the following CLI input:

```
N Chassis(rw)->set macauthentication enable
N Chassis(rw)->set macauthentication password enterasys
N Chassis(rw)->set macauthentication significant-bits 24
N Chassis(rw)->set macauthentication 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.

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:

```
N Chassis(rw)->set pwa enable
N Chassis(rw)->set pwa ipaddress 10.10.10.101
N Chassis(rw)->set banner \"Enterasys Networks Public Internet Access Station\"
N Chassis(rw)->set pwa enhancemode enable
N Chassis(rw)->set pwa gueststatus authradius
N Chassis(rw)->set pwa guestname guest
N Chassis(rw)->set pwa guestpassword password
N Chassis(rw)->set pwa portcontrol enable ge.1.6
```

This completes the Authentication configuration example.

Terms and Definitions

Table 33-4 lists terms and definitions used in this Authentication configuration discussion.

Table 33-4 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.
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.
MultiAuth Authentication	The ability to authenticate multiple authentication modes for a user and applying the authentication mode with the highest precedence.
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.
Port Web Authentication (PWA)	A means of authenticating a user by utilizing a web browser for the login process to authenticate to the network.
RADIUS Filter ID	An Enterasys 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.

Table 33-4 Quality of Service Configuration Terms and Definitions (continued)

Term	Definition
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.