Foundry FastIron X-Series Configuration Guide

FastIron Edge Switch X-Series

FastIron Workgroup Switch X-Series

FastIron SuperX Switch

2100 Gold Street
P.O. Box 649100
San Jose, CA 95164-9100
Tel 408.586.1700
Fax 408.586.1900

December 2005

Contents

Chapter 1
About This Guide

Introduction

This guide describes the following product families from Foundry Networks:

- FastIron Edge Switch X-Series (FESX) Layer 2/Layer 3 switch
- FastIron Workgroup Switch X-Series (FWSX) Layer 2 switch
- FastIron SuperX Switch (FSX) Layer 2/Layer 3 switch

This guide includes procedures for configuring the software. The software procedures show how to perform tasks using the CLI. This guide also describes how to monitor Foundry products using statistics and summary screens.

This guide applies to the following products:

- FastIron Edge Switch X-Series products:
  - FastIron Edge Switch X424
  - FastIron Edge Switch X448
- FastIron SuperX Switch
- FastIron Workgroup Switch X-Series products:
  - FastIron Workgroup Switch X424
  - FastIron Workgroup Switch X448

NOTE: This guide contains the terms **FastIron Edge Switch X-Series** (**FESX**), **FastIron SuperX Switch (FSX)**, and **FastIron WorkGroup Switch X-Series (FWSX**). Each term refers to a specific set of devices, as shown in Table 1.1.

Table 1.1: FastIron Family of Switches

This Name	Refers to These Devices
FastIron Edge Switch X-Series (FESX)	FESX424 and FESX448
FastIron SuperX Switch (FSX)	FastIron SuperX
FastIron Workgroup Switch X-Series (FWSX)	FWSX424 and FWSX448

What's Included in This Edition?

This edition describes the following software releases:

- For the FastIron Edge Switch X-Series products:
  - 02.3.03 (combined FESX/FSX/FWSX release)
  - 02.3.02 (combined FESX/FSX/FWSX release)
  - 02.3.01 (combined FESX/FSX/FWSX release)
  - 02.2.00 (combined FESX/FWSX release)
  - 02.1.01
  - 02.0.00
  - 01.1.00
  - 01.0.00
- For the FastIron SuperX Switch
  - 02.2.01
  - 02.2.00
  - 02.1.00
  - 02.0.01

NOTE: Software releases for FSX devices were combined with the FESX software releases starting with FESX release 02.3.01.

- For the FastIron Workgroup Switch X-Series products:
  - 02.0.00

NOTE: Software releases for FWSX devices were combined with the FESX software releases starting with FESX release 02.2.00.

Audience

This guide is designed for network installers, system administrators, and resellers who will configure the software for the FastIron family of switches. This guide assumes a working knowledge of Layer 2 and Layer 3 switching and routing concepts.

If you are using Layer 3 code, you should be familiar with the following protocols if applicable to your network – IP, RIP, OSPF, BGP4, DVMRP, MBGP, IGMP, PIM, VRRP, and VRRPE.

Nomenclature

This guide uses the following typographical conventions to show information:

Italic	highlights the title of another publication and occasionally emphasizes a word or phrase.
Bold	highlights a CLI command.
Bold Italic	highlights a term that is being defined.
Underline	highlights a link on the Web management interface.
Capitals	highlights field names and buttons that appear in the Web management interface.

NOTE: A note emphasizes an important fact or calls your attention to a dependency.

WARNING: A warning calls your attention to a possible hazard that can cause injury or death.

CAUTION: A caution calls your attention to a possible hazard that can damage equipment.

Related Publications

The following Foundry Networks documents supplement the information in this guide.

- *Foundry FastIron X-Series Chassis Hardware Installation Guide* – provides hardware installation procedures for the FastIron chassis devices (FSX).
- *Foundry FastIron Stackable Hardware Installation Guide* – provides hardware installation procedures for the FastIron stackable devices (FES, FESX, and FWSX).
- *Foundry Security Guide* – provides procedures for securing management access to Foundry devices and for protecting against Denial of Service (DoS) attacks.
- *Foundry Management Information Base Reference* – contains the Simple Network Management Protocol (SNMP) Management Information Base (MIB) objects supported on Foundry devices.
- Release Notes for the FastIron Edge Switch X-Series – describes features introduced in each software release, lists features that are supported on the FESX, and describes how configuration procedures or defaults differ from those on other Foundry devices, due to the FastIron Edge Switch X-Series' hardware architecture.
- Release Notes for the FastIron SuperX Switch – describes features introduced in each software release, lists features that are supported on the FSX, and describes how configuration procedures or defaults differ from those on other Foundry devices, due to the FSX's hardware architecture.
- Release Notes for the FastIron Workgroup Switch X-Series – describes features introduced in each software release, lists features that are supported on the FWSX, and describes how configuration procedures or defaults differ from those on other Foundry devices, due to the FastIron Workgroup Switch X-Series' hardware architecture.

To order additional copies of these manuals, do one of the following:

- Call 1.877.TURBOCALL (887.2622) in the United States or 1.408.586.1881 outside the United States.
- Send email to info@foundrynet.com.

How to Get Help

Foundry Networks technical support will ensure that the fast and easy access that you have come to expect from your Foundry Networks products will be maintained.

Web Access

- http://www.foundrynetworks.com

Email Access

Technical requests can also be sent to the following email address:

- support@foundrynet.com

Telephone Access

- 1.877.TURBOCALL (887.2622) United States
- 1.408.586.1881 Outside the United States

Warranty Coverage

Contact Foundry Networks using any of the methods listed above for information about the standard and extended warranties.

Chapter 2
Getting Familiar with Management Applications

This chapter describes how to manage a Foundry device using the various user interfaces listed in Table 2.1.

Table 2.1: Chapter Contents

Description	See Page
Command Line Interface (CLI) – a text-based interface accessible through a direct serial connection or a Telnet session.	2-1
Web management interface – A GUI-based management interface accessible through an HTTP (web browser) connection.	2-8
You can also use the ***IronView Network Manager***, an optional SNMP-based standalone GUI application, to manage the Foundry device. See the *Foundry IronView Network Management User's Guide* for information about using IronView Network Manager.	2-11

Logging on Through the CLI

Once an IP address is assigned to a Foundry device running Layer 2 software or to an interface on the Foundry device running Layer 3 software, you can access the CLI either through the direct serial connection to the device or through a local or remote Telnet session.

You can initiate a local Telnet or SNMP connection by attaching a cable to a port and specifying the assigned management station IP address.

The commands in the CLI are organized into the following levels:

- User EXEC – Lets you display information and perform basic tasks such as pings and traceroutes.
- Privileged EXEC – Lets you use the same commands as those at the User EXEC level plus configuration commands that do not require saving the changes to the system-config file.

- CONFIG – Lets you make configuration changes to the device. To save the changes across reboots, you need to save them to the system-config file. The CONFIG level contains sub-levels for individual ports, for VLANs, for routing protocols, and other configuration areas.

NOTE: By default, any user who can open a serial or Telnet connection to the Foundry device can access all these CLI levels. To secure access, you can configure Enable passwords or local user accounts, or you can configure the device to use a RADIUS or TACACS/TACACS+ server for authentication. See the *Foundry Security Guide*.

On-Line Help

To display a list of available commands or command options, enter "?" or press Tab. If you have not entered part of a command at the command prompt, all the commands supported at the current CLI level are listed. If you enter part of a command, then enter "?" or press Tab, the CLI lists the options you can enter at this point in the command string.

If you enter an invalid command followed by ?, a message appears indicating the command was unrecognized. For example:

```
FESX424 Router(config)# rooter ip
Unrecognized command
```

Command Completion

The CLI supports command completion, so you do not need to enter the entire name of a command or option. As long as you enter enough characters of the command or option name to avoid ambiguity with other commands or options, the CLI understands what you are typing.

Scroll Control

By default, the CLI uses a page mode to paginate displays that are longer than the number of rows in your terminal emulation window. For example, if you display a list of all the commands at the global CONFIG level but your terminal emulation window does not have enough rows to display them all at once, the page mode stops the display and lists your choices for continuing the display.

Here is an example:

```
aaa
all-client
appletalk
arp
boot
```

some lines omitted for brevity...

```
ipx
lock-address
logging
mac
--More--, next page: Space, next line:
Return key, quit: Control-c
```

The software provides the following scrolling options:

- Press the Space bar to display the next page (one screen at a time).
- Press the Return or Enter key to display the next line (one line at a time).
- Press Ctrl-C or Ctrl-Q to cancel the display.

Line Editing Commands

The CLI supports the following line editing commands. To enter a line-editing command, use the CTRL-key combination for the command by pressing and holding the CTRL key, then pressing the letter associated with the command.

Table 2.2: CLI Line Editing Commands

Ctrl-Key Combination	Description
Ctrl-A	Moves to the first character on the command line.
Ctrl-B	Moves the cursor back one character.
Ctrl-C	Escapes and terminates command prompts and ongoing tasks (such as lengthy displays), and displays a fresh command prompt.
Ctrl-D	Deletes the character at the cursor.
Ctrl-E	Moves to the end of the current command line.
Ctrl-F	Moves the cursor forward one character.
Ctrl-K	Deletes all characters from the cursor to the end of the command line.
Ctrl-L; Ctrl-R	Repeats the current command line on a new line.
Ctrl-N	Enters the next command line in the history buffer.
Ctrl-P	Enters the previous command line in the history buffer.
Ctrl-U; Ctrl-X	Deletes all characters from the cursor to the beginning of the command line.
Ctrl-W	Deletes the last word you typed.
Ctrl-Z	Moves from any CONFIG level of the CLI to the Privileged EXEC level; at the Privileged EXEC level, moves to the User EXEC level.

For a complete list of CLI commands and syntax information for each command, see the *Foundry Switch and Router Command Line Interface Reference*.

Using Slot and Port Numbers with CLI Commands

Many CLI commands and displays use port numbers, or slot numbers with port numbers. The ports are labeled on the front panel of the device.

The FSX uses chassis-based port numbering which consists of a slot number and a port number. When you enter CLI commands on the FSX, you must specify both the slot number and the port number. The FESX and FWSX devices do not use this type of numbering. When you enter commands on these devices, just specify the port number. The slot numbers used in the FSX CLI examples apply only to Chassis devices.

Here is an example. The following commands change the CLI from the global CONFIG level to the configuration level for the first port on the device.

- FSX commands:

```
FastIron SuperX Switch(config)# interface e 1/1
FastIron SuperX Switch(config-if-1/1)#
```

- FESX and FWSX commands:

```
(config)# interface e 1
(config-if-e1000-1)#
```

Searching and Filtering Output from CLI Commands

You can filter CLI output from **show** commands and at the --More-- prompt. You can search for individual characters, strings, or construct complex regular expressions to filter the output.

Searching and Filtering Output from show commands

You can filter output from **show** commands to display lines containing a specified string, lines that do not contain a specified string, or output starting with a line containing a specified string. The search string is a regular expression consisting of a single character or string of characters. You can use special characters to construct complex regular expressions. See “Using Special Characters in Regular Expressions” on page 2-6 for information on special characters used with regular expressions.

Displaying Lines Containing a Specified String

The following command filters the output of the **show interface** command for port 3/11 so it displays only lines containing the word “Internet”. This command can be used to display the IP address of the interface.

```
FastIron SuperX Switch# show interface e 3/11 | include Internet
  Internet address is 192.168.1.11/24, MTU 1518 bytes, encapsulation ethernet
```

Syntax: <show-command> | include <regular-expression>

NOTE: The vertical bar (|) is part of the command.

Note that the regular expression specified as the search string is case sensitive. In the example above, a search string of “Internet” would match the line containing the IP address, but a search string of “internet” would not.

Displaying Lines That Do Not Contain a Specified String

The following command filters the output of the **show who** command so it displays only lines that do not contain the word “closed”. This command can be used to display open connections to the Foundry device.

```
FESX424 Switch# show who | exclude closed
Console connections:
        established
        you are connecting to this session
        2 seconds in idle
Telnet connections (inbound):
 1      established, client ip address 192.168.9.37
        27 seconds in idle
Telnet connection (outbound):
SSH connections:
```

Syntax: <show-command> | exclude <regular-expression>

Displaying Lines Starting with a Specified String

The following command filters the output of the **show who** command so it displays output starting with the first line that contains the word “SSH”. This command can be used to display information about SSH connections to the Foundry device.

```
FESX424 Switch# show who | begin SSH
SSH connections:
 1      established, client ip address 192.168.9.210
        7 seconds in idle
 2      closed
 3      closed
 4      closed
 5      closed
```

Syntax: <show-command> | begin <regular-expression>

Searching and Filtering Output at the --More-- Prompt

The --More-- prompt displays when output extends beyond a single page. From this prompt, you can press the Space bar to display the next page, the Return or Enter key to display the next line, or Ctrl-C or Q to cancel the display. In addition, you can search and filter output from this prompt.

At the --More-- prompt, you can press the forward slash key (/) and then enter a search string. The Foundry device displays output starting from the first line that contains the search string, similar to the **begin** option for **show** commands. For example:

```
--More--, next page: Space, next line: Return key, quit: Control-c
/telnet
```

The results of the search are displayed:

```
searching...
  telnet              Telnet by name or IP address
  temperature         temperature sensor commands
  terminal            display syslog
  traceroute          TraceRoute to IP node
  undebug             Disable debugging functions (see also 'debug')
  undelete            Undelete flash card files
  whois               WHOIS lookup
  write               Write running configuration to flash or terminal
```

To display lines containing only a specified search string (similar to the **include** option for **show** commands) press the plus sign key (+) at the --More-- prompt and then enter the search string.

```
--More--, next page: Space, next line: Return key, quit: Control-c
+telnet
```

The filtered results are displayed:

```
filtering...
  telnet              Telnet by name or IP address
```


To display lines that do not contain a specified search string (similar to the **exclude** option for **show** commands) press the minus sign key (-) at the --More-- prompt and then enter the search string.

```
--More--, next page: Space, next line: Return key, quit: Control-c
-telnet
```

The filtered results are displayed:

```
filtering...
  temperature         temperature sensor commands
  terminal            display syslog
  traceroute          TraceRoute to IP node
  undebug             Disable debugging functions (see also 'debug')
  undelete            Undelete flash card files
  whois               WHOIS lookup
  write               Write running configuration to flash or terminal
```

As with the commands for filtering output from **show** commands, the search string is a regular expression consisting of a single character or string of characters. You can use special characters to construct complex regular expressions. See the next section for information on special characters used with regular expressions.

Using Special Characters in Regular Expressions

You use a regular expression to specify a single character or multiple characters as a search string. In addition, you can include special characters that influence the way the software matches the output against the search string. These special characters are listed in the following table.

Table 2.3: Special Characters for Regular Expressions

Character	Operation
.	The period matches on any single character, including a blank space. For example, the following regular expression matches “aaz”, “abz”, “acz”, and so on, but not just “az”: a.z
*	The asterisk matches on zero or more sequential instances of a pattern. For example, the following regular expression matches output that contains the string “abc”, followed by zero or more Xs: abcX*
+	The plus sign matches on one or more sequential instances of a pattern. For example, the following regular expression matches output that contains "de", followed by a sequence of “g”s, such as “deg”, “degg”, “deggg”, and so on: deg+

Table 2.3: Special Characters for Regular Expressions (Continued)

Character	Operation
?	The question mark matches on zero occurrences or one occurrence of a pattern. For example, the following regular expression matches output that contains "dg" or "deg": de?g **Note:** Normally when you type a question mark, the CLI lists the commands or options at that CLI level that begin with the character or string you entered. However, if you enter Ctrl-V and then type a question mark, the question mark is inserted into the command line, allowing you to use it as part of a regular expression.
^	A caret (when not used within brackets) matches on the beginning of an input string. For example, the following regular expression matches output that begins with “deg”: ^deg
$	A dollar sign matches on the end of an input string. For example, the following regular expression matches output that ends with “deg”: deg$
_	An underscore matches on one or more of the following: • **,** (comma) • **{** (left curly brace) • **}** (right curly brace) • **(** (left parenthesis) • **)** (right parenthesis) • The beginning of the input string • The end of the input string • A blank space For example, the following regular expression matches on “100” but not on “1002”, “2100”, and so on. _100_
[ ]	Square brackets enclose a range of single-character patterns. For example, the following regular expression matches output that contains “1”, “2”, “3”, “4”, or “5”: [1-5] You can use the following expression symbols within the brackets. These symbols are allowed only inside the brackets. • **^ –** The caret matches on any characters ***except*** the ones in the brackets. For example, the following regular expression matches output that does ***not*** contain “1”, “2”, “3”, “4”, or “5”: [^1-5] • **-** The hyphen separates the beginning and ending of a range of characters. A match occurs if any of the characters within the range is present. See the example above.

Table 2.3: Special Characters for Regular Expressions (Continued)

Character	Operation
|	A vertical bar separates two alternative values or sets of values. The output can match one or the other value. For example, the following regular expression matches output that contains either “abc” or “defg”: abc|defg
( )	Parentheses allow you to create complex expressions. For example, the following complex expression matches on “abc”, “abcabc”, or “defg”, but not on “abcdefgdefg”: ((abc)+)|((defg)?)

If you want to filter for a special character instead of using the special character as described in the table above, enter “****” (backslash) in front of the character. For example, to filter on output containing an asterisk, enter the asterisk portion of the regular expression as “*****”.

```
FESX424 Router# show ip route bgp | include \*
```

Logging On Through the Web Management Interface

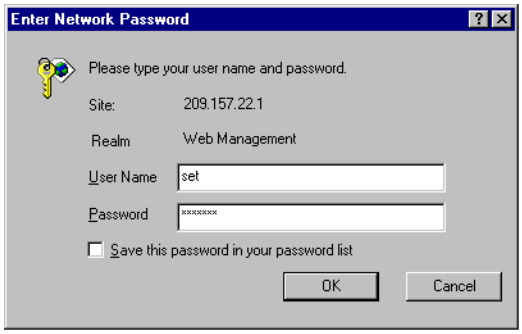
To use the Web management interface, open a web browser and enter the IP address of the Foundry device’s management port in the Location or Address field. The web browser contacts the Foundry device and displays a Login panel, such as the one shown below for the FESX.

Figure 2.1 Web Management Interface Login Panel

NOTE: If you are unable to connect with the device through a Web browser due to a proxy problem, it may be necessary to set your Web browser to direct Internet access instead of using a proxy. For information on how to change a proxy setting, refer to the on-line help provided with your Web browser.

To log in, click on the Login link. The following dialog box is displayed.

Figure 2.2 Web management interface login dialog



The login username and password you enter depends on whether your device is configured with AAA authentication for SNMP. If AAA authentication for SNMP is not configured, you can use the user name “get” and the default read-only password “public” for read-only access. However, for read-write access, you must enter “set” for the user name, and enter a read-write community string you have configured on the device for the password. There is no default read-write community string. You must add one using the CLI. See the *Foundry Security Guide*.

As an alternative to using the SNMP community strings to log in, you can configure the Foundry device to secure Web management access using local user accounts or Access Control Lists (ACLs). See the *Foundry Security Guide*.

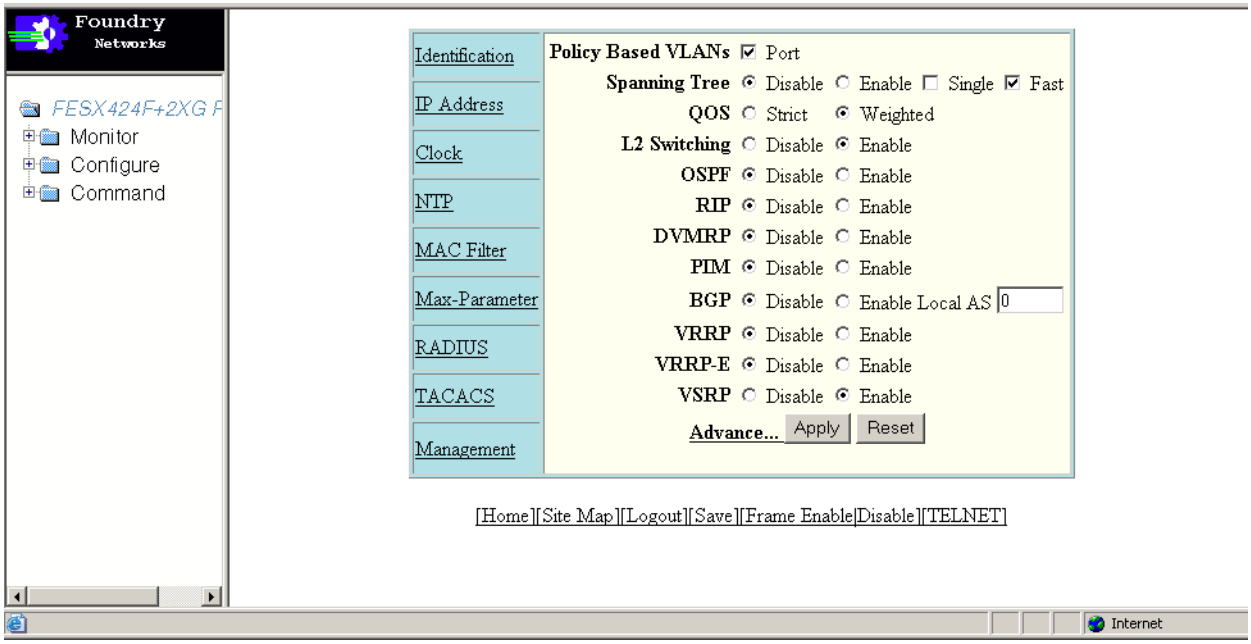
Navigating the Web Management Interface

When you log into a device, the System configuration panel is displayed. This panel allows you to enable or disable major system features. You can return to this panel from any other panel by selecting the Home link.

The Site Map link gives you a view of all available options on a single screen.

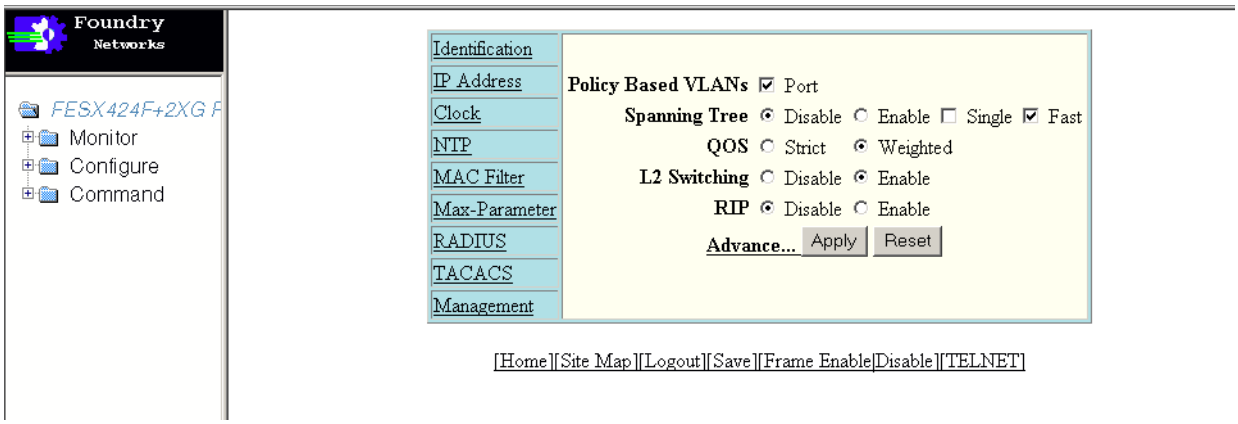
Figure 2.3 displays the first Web management interface panel for Layer 3 Switch features, while Figure 2.4 displays the first panel for Layer 2 Switch features. These panels allow you to configure the features supported by the Layer 3 Switch and Layer 2 Switch software.

Figure 2.3 First Panel for Layer 3 Switch Features



NOTE: If you are using Internet Explorer 6.0 to view the Web management interface, make sure the version you are running includes the latest service pack(s). Otherwise, the navigation tree (the left-most pane in Figure 2.3) will not display properly. For information on how to load the latest service pack(s), refer to the on-line help provided with your Web browser.

Figure 2.4 First Panel for Layer 2 Switch Features



NOTE: If you are using Internet Explorer 6.0 to view the Web management interface, make sure the version you are running includes the latest service pack(s). Otherwise, the navigation tree (the left-most pane in Figure 2.3) will not display properly. For information on how to load the latest service pack(s), refer to the on-line help provided with your Web browser.

The left pane of the Web management interface window contains a “tree view,” similar to the one found in Windows Explorer. Configuration options are grouped into folders in the tree view. These folders, when expanded, reveal additional options. To expand a folder, click on the plus sign to the left of the folder icon.

You can configure the appearance of the Web management interface by using one of the following methods.

Using the CLI, you can modify the appearance of the Web management interface with the **web-management** command.

To cause the Web management interface to display the List view by default:

```
FESX424 Router(config)# web-management list-menu
```

To disable the front panel frame:

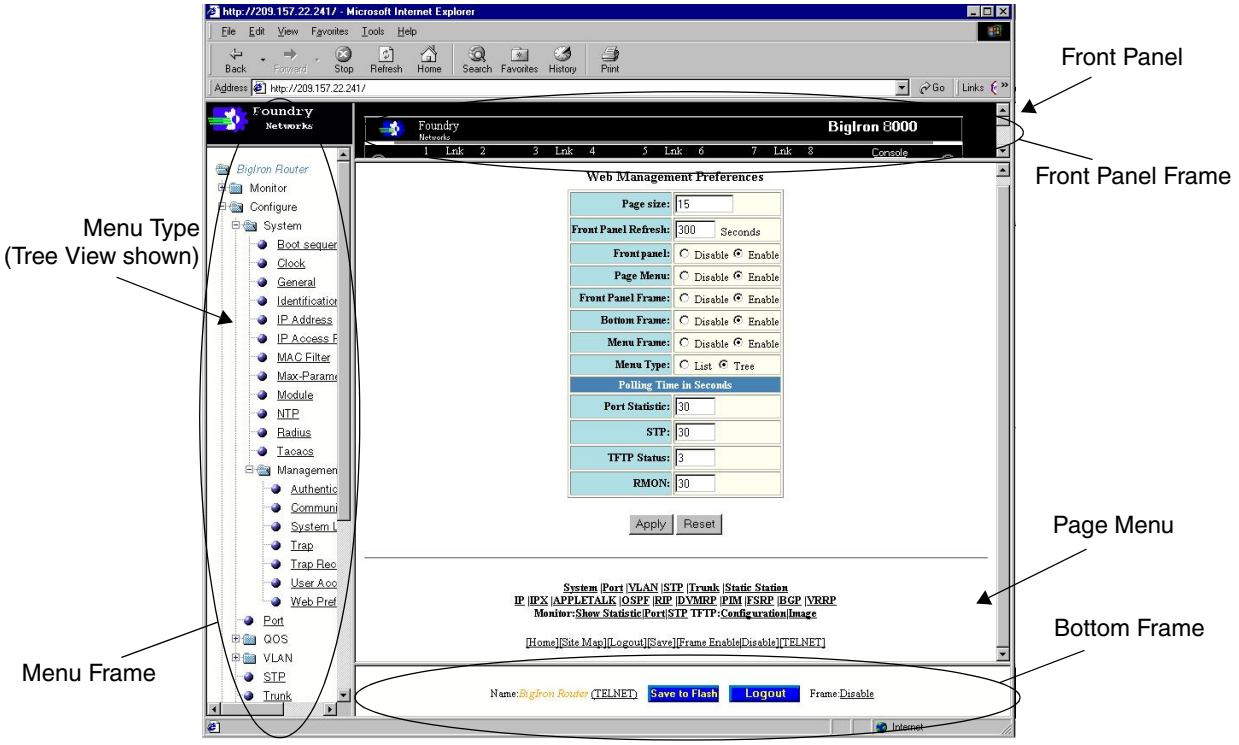
```
FESX424 Router(config)# no web-management front-panel
```

When you save the configuration with the **write memory** command, the changes will take place the next time you start the Web management interface, or if you are currently running the Web management interface, the changes will take place when you click the Refresh button on your browser.

USING THE WEB MANAGEMENT INTERFACE

1. Click on the plus sign next to Configure in the tree view to expand the list of configuration options.
2. Click on the plus sign next to System in the tree view to expand the list of system configuration links.
3. Click on the plus sign next to Management in the tree view to expand the list of system management links.
4. Click on the Web Preference link to display the Web Management Preferences panel.

5. Enable or disable elements on the Web management interface by clicking on the appropriate radio buttons on the panel. The following figure identifies the elements you can change.



NOTE: The tree view is available when you use the Web management interface with Netscape 4.0 or higher or Internet Explorer 4.0 or higher browsers. If you use the Web management interface with an older browser, the Web management interface displays the List view only, and the Web Management Preferences panel does not include an option to display the tree view.

6. When you have finished, click the Apply button on the panel, then click the Refresh button on your browser to activate the changes.
7. To save the configuration, click the plus sign next to the Command folder, then click the Save to Flash link.

NOTE: The only changes that become permanent are the settings to the Menu Type and the Front Panel Frame. Any other elements you enable or disable will go back to their default settings the next time you start the Web management interface.

Logging on Through IronView Network Manager

See the *Foundry IronView Network Management User's Guide* for information about using IronView Network Manager.

Chapter 3
Configuring Basic Software Features

This chapter describes how to configure basic, non-protocol features on the FastIron family of switches.

Foundry devices are configured at the factory with default parameters that allow you to begin using the basic features of the system immediately. However, many of the advanced features such as VLANs or routing protocols for the device must first be enabled at the system (global) level before they can be configured. If you use the Command Line Interface (CLI) to configure system parameters, you can find these system level parameters at the Global CONFIG level of the CLI.

This chapter contains procedures for configuring the following parameters:

Table 3.1: Chapter Contents

Description	See Page
Basic system parameters – This section lists the basic system parameters and gives instructions for configuring them.	3-2
Basic port parameters – This section lists basic port parameters and gives instructions for configuring them.	3-13

NOTE: Before assigning or modifying any router parameters, you must assign the IP subnet (interface) addresses for each port.

NOTE: For information about configuring IP addresses, DNS resolver, DHCP assist, and other IP-related parameters, see the chapter “Configuring IP” on page 16-1.

For information about the Syslog buffer and messages, see the Appendix “Using Syslog” on page A-1.

Configuring Basic System Parameters

The procedures in this section describe how to configure the basic system parameters listed in Table 3.2.

Table 3.2: Basic System Parameters

Basic System Parameter	See Page
System name, contact, and location	3-2
SNMP trap receiver, trap source address, and other parameters	3-3
Single source address for all Telnet packets	3-7
Single source address for all TFTP packets	3-7
System time using a Simple Network Time Protocol (SNTP) server or local system counter	3-8, 3-10
Broadcast, multicast, or unknown-unicast limits, if required to support slower third-party devices	3-11
Banners that are displayed on users' terminals when they enter the Privileged EXEC CLI level or access the device through Telnet	3-11

NOTE: For information about the Syslog buffer and messages, see "Using Syslog" on page A-1.

Entering System Administration Information

You can configure a system name, contact, and location for a Foundry device and save the information locally in the configuration file for future reference. This information is not required for system operation but is suggested. When you configure a system name, the name replaces the default system name in the CLI command prompt.

The name, contact, and location each can be up to 32 alphanumeric characters.

Here is an example of how to configure a system name, system contact, and location:

```
FastIron SuperX Switch(config)# hostname zappa
zappa(config)# snmp-server contact Support Services
zappa(config)# snmp-server location Centerville
zappa(config)# end
zappa# write memory
```

Syntax: hostname <string>

Syntax: snmp-server contact <string>

Syntax: snmp-server location <string>

The text strings can contain blanks. The SNMP text strings do not require quotation marks when they contain blanks but the host name does.

NOTE: The **chassis name** command does not change the CLI prompt. Instead, the command assigns an administrative ID to the device.

Configuring Simple Network Management Protocol (SNMP) Parameters

Use the procedures in this section to perform the following configuration tasks:

- Specify an SNMP trap receiver.
- Specify a source address and community string for all traps sent by the device.
- Change the holddown time for SNMP traps
- Disable individual SNMP traps. (All traps are enabled by default.)
- Disable traps for CLI access that is authenticated by a local user account, a RADIUS server, or a TACACS/ TACACS+ server.

NOTE: To add and modify “get” (read-only) and “set” (read-write) community strings, see the *Foundry Security Guide*.

Specifying an SNMP Trap Receiver

You can specify a trap receiver to ensure that all SNMP traps sent by the Foundry device go to the same SNMP trap receiver or set of receivers, typically one or more host devices on the network. When you specify the host, you also specify a community string. The Foundry device sends all the SNMP traps to the specified host(s) and includes the specified community string. Administrators can therefore filter for traps from a Foundry device based on IP address or community string.

When you add a trap receiver, the software automatically encrypts the community string you associate with the receiver when the string is displayed by the CLI or Web management interface. If you want the software to show the community string in the clear, you must explicitly specify this when you add a trap receiver. In either case, the software does not encrypt the string in the SNMP traps sent to the receiver.

To specify the host to which the device sends all SNMP traps, use one of the following methods.

To add a trap receiver and encrypt the display of the community string, enter commands such as the following:

To specify an SNMP trap receiver and change the UDP port that will be used to receive traps, enter a command such as the following:

```
FESX424 Switch(config)# snmp-server host 2.2.2.2 0 mypublic port 200
FESX424 Switch(config)# write memory
```

Syntax: snmp-server host <ip-addr> [0 | 1] <string> [port <value>]

The <ip-addr> parameter specifies the IP address of the trap receiver.

The **0 | 1** parameter specifies whether you want the software to encrypt the string (**1**) or show the string in the clear (**0**). The default is **0**.

The <string> parameter specifies an SNMP community string configured on the Foundry device. The string can be a read-only string or a read-write string. The string is not used to authenticate access to the trap host but is instead a useful method for filtering traps on the host. For example, if you configure each of your Foundry devices that use the trap host to send a different community string, you can easily distinguish among the traps from different Foundry devices based on the community strings.

The command in the example above adds trap receiver 2.2.2.2 and configures the software to encrypt display of the community string. When you save the new community string to the startup-config file (using the **write memory** command), the software adds the following command to the file:

```
snmp-server host 2.2.2.2 1 <encrypted-string>
```

To add a trap receiver and configure the software to encrypt display of the community string in the CLI and Web management interface, enter commands such as the following:

```
FESX424 Switch(config)# snmp-server host 2.2.2.2 0 FastIron-12
FESX424 Switch(config)# write memory
```

The **port** <value> parameter allows you to specify which UDP port will be used by the trap receiver. This parameter allows you to configure several trap receivers in a system. With this parameter, IronView Network

Manager Network Manager and another network management application can coexist in the same system. Foundry devices can be configured to send copies of traps to more than one network management application.

Specifying a Single Trap Source

You can specify a single trap source to ensure that all SNMP traps sent by the Foundry device use the same source IP address. When you configure the SNMP source address, you specify the Ethernet port, loopback interface, or virtual interface that is the source for the traps. The Foundry device then uses the lowest-numbered IP address configured on the port or interface as the source IP address in the SNMP traps sent by the device.

Identifying a single source IP address for SNMP traps provides the following benefits:

- If your trap receiver is configured to accept traps only from specific links or IP addresses, you can use this feature to simplify configuration of the trap receiver by configuring the Foundry device to always send the traps from the same link or source address.
- If you specify a loopback interface as the single source for SNMP traps, SNMP trap receivers can receive traps regardless of the states of individual links. Thus, if a link to the trap receiver becomes unavailable but the receiver can be reached through another link, the receiver still receives the trap, and the trap still has the source IP address of the loopback interface.

To specify a port, loopback interface, or virtual interface whose lowest-numbered IP address the Foundry device must use as the source for all SNMP traps sent by the device, use the following CLI method.

To configure the device to send all SNMP traps from the first configured IP address on port 4, enter the following commands:

```
FESX424 Switch(config)# snmp trap-source ethernet 4
FESX424 Switch(config)# write memory
```

Syntax: snmp-server trap-source loopback <num> | ethernet [<slotnum>/]<portnum> | ve <num>

The <num> parameter is a loopback interface or virtual interface number.

If you specify an Ethernet port, the <portnum> is the port's number. If you are configuring a chassis device, specify the slot number as well as the port number (<slotnum>/<portnum>).

To specify a loopback interface as the device's SNMP trap source, enter commands such as the following:

```
FESX424 Switch(config)# int loopback 1
FESX424 Switch(config-lbif-1)# ip address 10.0.0.1/24
FESX424 Switch(config-lbif-1)# exit
FESX424 Switch(config)# snmp-server trap-source loopback 1
```

The commands in this example configure loopback interface 1, assign IP address 10.00.1/24 to the loopback interface, then designate the interface as the SNMP trap source for this device. Regardless of the port the Foundry device uses to send traps to the receiver, the traps always arrive from the same source IP address.

Setting the SNMP Trap Holddown Time

When a Foundry device starts up, the software waits for Layer 2 convergence (STP) and Layer 3 convergence (OSPF) before beginning to send SNMP traps to external SNMP servers. Until convergence occurs, the device might not be able to reach the servers, in which case the messages are lost.

By default, a Foundry device uses a one-minute holddown time to wait for the convergence to occur before starting to send SNMP traps. After the holddown time expires, the device sends the traps, including traps such as "cold start" or "warm start" that occur before the holddown time expires.

You can change the holddown time to a value from one second to ten minutes.

To change the holddown time for SNMP traps, enter a command such as the following at the global CONFIG level of the CLI:

```
FESX424 Switch(config)# snmp-server enable traps holddown-time 30
```

The command in this example changes the holddown time for SNMP traps to 30 seconds. The device waits 30 seconds to allow convergence in STP and OSPF before sending traps to the SNMP trap receiver.

Syntax: [no] snmp-server enable traps holddown-time <secs>

The <secs> parameter specifies the number of seconds and can be from 1 – 600 (ten minutes). The default is 60 seconds.

Disabling SNMP Traps

Foundry devices come with SNMP trap generation enabled by default for all traps. You can selectively disable one or more of the following traps.

NOTE: By default, all SNMP traps are enabled at system startup.

Layer 2 Traps

The following traps are generated on devices running Layer 2 software:

- SNMP authentication keys
- Power supply failure
- Fan failure
- Cold start
- Link up
- Link down
- Bridge new root
- Bridge topology change
- Locked address violation

Layer 3 Traps

The following traps are generated on devices running Layer 3 software:

- SNMP authentication key
- Power supply failure
- Fan failure
- Cold start
- Link up
- Link down
- Bridge new root
- Bridge topology change
- Locked address violation
- BGP4
- OSPF
- VRRP
- VRRPE

To stop link down occurrences from being reported, enter the following:

```
FESX424 Router(config)# no snmp-server enable traps link-down
```

Syntax: [no] snmp-server enable traps <trap-type>

NOTE: For a list of the trap values, see the *Foundry Switch and Router Command Line Interface Reference*.

Disabling Syslog Messages and Traps for CLI Access

Foundry devices send Syslog messages and SNMP traps when a user logs into or out of the User EXEC or Privileged EXEC level of the CLI. The feature applies to users whose access is authenticated by an authentication-method list based on a local user account, RADIUS server, or TACACS/TACACS+ server.

NOTE: The Privileged EXEC level is sometimes called the "Enable" level, because the command for accessing this level is **enable**.

The feature is enabled by default.

Examples of Syslog Messages for CLI Access

When a user whose access is authenticated by a local user account, a RADIUS server, or a TACACS/TACACS+ server logs into or out of the CLI's User EXEC or Privileged EXEC mode, the software generates a Syslog message and trap containing the following information:

- The time stamp
- The user name
- Whether the user logged in or out
- The CLI level the user logged into or out of (User EXEC or Privileged EXEC level)

NOTE: Messages for accessing the User EXEC level apply only to access through Telnet. The device does not authenticate initial access through serial connections but does authenticate serial access to the Privileged EXEC level. Messages for accessing the Privileged EXEC level apply to access through the serial connection or Telnet.

The following examples show login and logout messages for the User EXEC and Privileged EXEC levels of the CLI:

```
FESX424 Switch(config)# show logging

Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
Buffer logging: level ACDMEINW, 12 messages logged
level code: A=alert C=critical D=debugging M=emergency E=error
I=informational N=notification W=warning

Static Log Buffer:
Dec 15 19:04:14:A:Fan 1, fan on right connector, failed

Dynamic Log Buffer (50 entries):
Oct 15 18:01:11:info:dg logout from USER EXEC mode
Oct 15 17:59:22:info:dg logout from PRIVILEGE EXEC mode
Oct 15 17:38:07:info:dg login to PRIVILEGE EXEC mode
Oct 15 17:38:03:info:dg login to USER EXEC mode
```

Syntax: show logging

The first message (the one on the bottom) indicates that user "dg" logged in to the CLI's User EXEC level on October 15 at 5:38 PM and 3 seconds (Oct 15 17:38:03). The same user logged into the Privileged EXEC level four seconds later.

The user remained in the Privileged EXEC mode until 5:59 PM and 22 seconds. (The user could have used the CONFIG modes as well. Once you access the Privileged EXEC level, no further authentication is required to access the CONFIG levels.) At 6:01 PM and 11 seconds, the user ended the CLI session.

Disabling the Syslog Messages and Traps

Logging of CLI access is enabled by default. If you want to disable the logging, enter the following commands:

```
FESX424 Router(config)# no logging enable user-login
FESX424 Router(config)# write memory
```



```
FESX424 Router(config)# end
FESX424 Router# reload
```

Syntax: [no] logging enable user-login

Configuring an Interface as the Source for All Telnet Packets

You can designate the lowest-numbered IP address configured on an interface as the source IP address for all Telnet packets from the device. Identifying a single source IP address for Telnet packets provides the following benefits:

- If your Telnet server is configured to accept packets only from specific links or IP addresses, you can use this feature to simplify configuration of the Telnet server by configuring the Foundry device to always send the Telnet packets from the same link or source address.
- If you specify a loopback interface as the single source for Telnet packets, Telnet servers can receive the packets regardless of the states of individual links. Thus, if a link to the Telnet server becomes unavailable but the client or server can be reached through another link, the client or server still receives the packets, and the packets still have the source IP address of the loopback interface.

The software contains separate CLI commands for specifying the source interface for Telnet, TACACS/TACACS+, and RADIUS packets. You can configure a source interface for one or more of these types of packets.

To specify an interface as the source for all Telnet packets from the device, use the following CLI method. The software uses the lowest-numbered IP address configured on the interface as the source IP address for Telnet packets originated by the device.

To specify the lowest-numbered IP address configured on a virtual interface as the device's source for all Telnet packets, enter commands such as the following:

```
FESX424 Switch(config)# int loopback 2
FESX424 Switch(config-lbif-2)# ip address 10.0.0.2/24
FESX424 Switch(config-lbif-2)# exit
FESX424 Switch(config)# ip telnet source-interface loopback 2
```

The commands in this example configure loopback interface 2, assign IP address 10.0.0.2/24 to the interface, then designate the interface as the source for all Telnet packets from the device.

Syntax: ip telnet source-interface ethernet [<slotnum>/]<portnum> | loopback <num> | ve <num>

The following commands configure an IP interface on an Ethernet port and designate the address port as the source for all Telnet packets from the device.

```
FESX424 Switch(config)# interface ethernet 4
FESX424 Switch(config-if-e1000-4)# ip address 209.157.22.110/24
FESX424 Switch(config-if-e1000-4)# exit
FESX424 Switch(config)# ip telnet source-interface ethernet 4
```

Cancelling an Outbound Telnet Session

If you want to cancel a Telnet session from the console to a remote Telnet server (for example, if the connection is frozen), you can terminate the Telnet session by doing the following:

1. At the console, press Ctrl-^ (Ctrl-Shift-6).
2. Press the X key to terminate the Telnet session.

Pressing Ctrl-^ twice in a row causes a single Ctrl-^ character to be sent to the Telnet server. After you press Ctrl-^, pressing any key other than X or Ctrl-^ returns you to the Telnet session.

Configuring an Interface as the Source for All TFTP Packets

You can configure the device to use the lowest-numbered IP address configured on a loopback interface, virtual interface, or Ethernet port as the source for all TFTP packets from the device. The software uses the lowest-numbered IP address configured on the interface as the source IP address for the packets.

For example, to specify the lowest-numbered IP address configured on a virtual interface as the device's source for all TFTP packets, enter commands such as the following:

```
FESX424 Switch(config)# int ve 1
FESX424 Switch(config-vif-1)# ip address 10.0.0.3/24
FESX424 Switch(config-vif-1)# exit

FESX424 Switch(config)# ip tftp source-interface ve 1
```

The commands in this example configure virtual interface 1, assign IP address 10.0.0.3/24 to the interface, then designate the interface's address as the source address for all TFTP packets.

Syntax: [no] ip tftp source-interface ethernet [<slotnum>/]<portnum> | loopback <num> | ve <num>

The default is the lowest-numbered IP address configured on the port through which the packet is sent. The address therefore changes, by default, depending on the port.

Specifying a Simple Network Time Protocol (SNTP) Server

You can configure Foundry devices to consult SNTP servers for the current system time and date.

NOTE: Foundry devices do not retain time and date information across power cycles. Unless you want to reconfigure the system time counter each time the system is reset, Foundry Networks recommends that you use the SNTP feature.

To identify an SNTP server with IP address 208.99.8.95 to act as the clock reference for a Foundry device, enter the following:

```
FESX424 Switch(config)# sntp server 208.99.8.95
```

Syntax: sntp server <ip-addr> | <hostname> [<version>]

The <version> parameter specifies the SNTP version the server is running and can be from 1 – 4. The default is 1. You can configure up to three SNTP servers by entering three separate **sntp server** commands.

By default, the Foundry device polls its SNTP server every 30 minutes (1800 seconds). To configure the Foundry device to poll for clock updates from a SNTP server every 15 minutes, enter the following:

```
FESX424 Switch(config)# sntp poll-interval 900
```

Syntax: [no] sntp poll-interval <1-65535>

To display information about SNTP associations, enter the following command:

```
FESX424 Switch# show sntp associations
  address         ref clock       st  when  poll  delay  disp
 ~207.95.6.102    0.0.0.0         16   202     4    0.0     5.45
 ~207.95.6.101    0.0.0.0         16   202     0    0.0     0.0
* synced, ~ configured
```

Syntax: show sntp associations

The following table describes the information displayed by the **show sntp associations** command.

Table 3.3: Output from the show sntp associations command

This Field...	Displays...
(leading character)	One or both of the following: * Synchronized to this peer ~ Peer is statically configured
address	IP address of the peer
ref clock	IP address of the peer's reference clock
st	NTP stratum level of the peer
when	Amount of time since the last NTP packet was received from the peer
poll	Poll interval in seconds
delay	Round trip delay in milliseconds
disp	Dispersion in seconds

To display information about SNTP status, enter the following command:

```
FESX424 Switch# show sntp status
Clock is unsynchronized, stratum = 0, no reference clock
precision is 2**0
reference time is 0       .0
clock offset is 0.0    msec, root delay is 0.0  msec
root dispersion is 0.0  msec, peer dispersion is 0.0  msec
```

Syntax: show sntp status

The following table describes the information displayed by the **show sntp status** command.

Table 3.4: Output from the show sntp status command

This Field...	Indicates...
unsynchronized	System is not synchronized to an NTP peer.
synchronized	System is synchronized to an NTP peer.
stratum	NTP stratum level of this system
reference clock	IP Address of the peer (if any) to which the unit is synchronized
precision	Precision of this system's clock (in Hz)
reference time	Reference time stamp
clock offset	Offset of clock to synchronized peer
root delay	Total delay along the path to the root clock
root dispersion	Dispersion of the root path

Table 3.4: Output from the show sntp status command (Continued)

This Field...	Indicates...
peer dispersion	Dispersion of the synchronized peer

Setting the System Clock

In addition to SNTP support, Foundry switches and routers also allow you to set the system time counter. The time counter setting is not retained across power cycles and is not automatically synchronized with an SNTP server. The counter merely starts the system time and date clock with the time and date you specify.

NOTE: You can synchronize the time counter with your SNTP server time by entering the **sntp sync** command from the Privileged EXEC level of the CLI.

NOTE: Unless you identify an SNTP server for the system time and date, you will need to re-enter the time and date following each reboot.

For more details about SNTP, see “Specifying a Simple Network Time Protocol (SNTP) Server” on page 3-8.

To set the system time and date to 10:15:05 on October 15, 2003, enter the following command:

```
FESX424 Switch# clock set 10:15:05 10-15-2003
```

Syntax: [no] clock set <hh:mm:ss> <mm-dd-yy> | <mm-dd-yyyy>

By default, Foundry switches and routers do not change the system time for daylight savings time. To enable daylight savings time, enter the following command:

```
FESX424 Switch# clock summer-time
```

Syntax: clock summer-time

Although SNTP servers typically deliver the time and date in Greenwich Mean Time (GMT), you can configure the Foundry device to adjust the time for any one-hour offset from GMT or for one of the following U.S. time zones:

- US Pacific (default)
- Alaska
- Aleutian
- Arizona
- Central
- East-Indiana
- Eastern
- Hawaii
- Michigan
- Mountain
- Pacific
- Samoa

The default is US Pacific.

To change the time zone to Australian East Coast time (which is normally 10 hours ahead of GMT), enter the following command:

```
FESX424 Router(config)# clock timezone gmt+10
```

Syntax: clock timezone gmt | us <time-zone>

You can enter one of the following values for <time-zone>:

- US time zones (**us**): alaska, aleutian, arizona, central, east-indiana, eastern, hawaii, michigan, mountain, pacific, samoa.
- GMT time zones (**gmt**): gmt+12, gmt+11, gmt+10...gmt+01, gmt+00, gmt-01...gmt-10, gmt-11, gmt-12.

Limiting Broadcast, Multicast, and Unknown Unicast Traffic

FastIron devices can forward all traffic at wire speed. However, some third-party networking devices cannot handle high forwarding rates for broadcast, multicast, or unknown-unicast packets. You can limit the number of broadcast, multicast, or unknown-unicast packets a Foundry device forwards each second using the procedures in this section. You can configure limits on individual ports or groups of ports.

On the FESX, FWSX, and FSX, unknown unicast limiting is independent of broadcast and multicast limiting.

When you configure unknown-unicast limiting, the rate applies to all ports in the **port range** for which unknown unicast is enabled. On the FESX, FWSX, and FSX, a 1-Gigabit port range consists of 12 ports. For example, the FESX424 has 2 port ranges; ports 1 – 12 are one port range, and ports 13 – 24 are another port range. If you enable unknown unicast limiting on port 2, the configuration applies to the ports from 1 – 12 that have unknown unicast limiting enabled. 10-Gigabit ports are not grouped into ranges. So if your device has two 10-Gigabit uplinks, you can configure different unknown-unicast limits for each 10-Gigabit port.

Command Syntax

To enable broadcast limiting on a group of ports, enter commands such as the following:

```
FESX424 Switch(config)# interface ethernet 1 to 8
FESX424 Switch(config-mif-e1000-1-8)# broadcast limit 65536
```

These commands configure broadcast limiting on ports 1 – 8. On each port, the total combined number of broadcasts cannot exceed 65,536.

To include multicasts in the 65536 packets per second limit on each of the ports, enter the following command after enabling broadcast limiting:

```
FESX424 Switch(config-mif-e1000-1-8)# multicast limit
```

To enable unknown unicast limiting, enter commands such as the following:

```
FESX424 Switch# config terminal
FESX424 Switch(config)# int e 1
FESX424 Switch(config-if-e1000-1)# unknown unicast limit 65536
The combined number of inbound Unknown Unicast packets permitted
    for ports 1 to 12 is now set to 65536
FESX424 Switch((config-if-e1000-1)#
```

Syntax: [no] broadcast limit <num>

Syntax: [no] multicast limit

Syntax: [no]unknown unicast limit <num>

The <num> parameter specifies the maximum number of packets per second and can be any number that is a multiple of 65536, up to a maximum value of 4294967295. If you enter the **multicast limit** command, multicast packets are included in the limit you specify. If you specify 0, limiting is disabled. If you specify a number that is not a multiple of 65536, the software rounds the number to the next multiple of 65536. Limiting is disabled by default.

Configuring CLI Banners

Foundry devices can be configured to display a greeting message on users' terminals when they enter the Privileged EXEC CLI level or access the device through Telnet. In addition, a Foundry device can display a message on the Console when an incoming Telnet CLI session is detected.

Setting a Message of the Day Banner

You can configure the Foundry device to display a message on a user's terminal when he or she establishes a Telnet CLI session. For example, to display the message "Welcome to FESX!" when a Telnet CLI session is established:

```
FESX424 Switch(config)# banner motd $ (Press Return)
Enter TEXT message, End with the character '$'.
Welcome to FESX! $
```

A delimiting character is established on the first line of the **banner motd** command. You begin and end the message with this delimiting character. The delimiting character can be any character except " (double-quotation mark) and cannot appear in the banner text. In this example, the delimiting character is $ (dollar sign). The text in between the dollar signs is the contents of the banner. The banner text can be up to 2048 characters long and can consist of multiple lines. To remove the banner, enter the **no banner motd** command.

Syntax: [no] banner <delimiting-character> | [motd <delimiting-character>]

NOTE: The **banner** <delimiting-character> command is equivalent to the **banner motd** <delimiting-character> command.

When you access the Web management interface, the banner is displayed:

[Login]

Setting a Privileged EXEC CLI Level Banner

You can configure the Foundry device to display a message when a user enters the Privileged EXEC CLI level. For example:

```
FastIron SuperX Switch(config)# banner exec_mode # (Press Return)
Enter TEXT message, End with the character '#'.
You are entering Privileged EXEC level
Don't foul anything up! #
```

As with the **banner motd** command, you begin and end the message with a delimiting character; in this example, the delimiting character is # (pound sign). To remove the banner, enter the **no banner exec_mode** command.

Syntax: [no] banner exec_mode <delimiting-character>

Displaying a Message on the Console When an Incoming Telnet Session Is Detected

You can configure the Foundry device to display a message on the Console when a user establishes a Telnet session. This message indicates where the user is connecting from and displays a configurable text message.

For example:

```
FastIron SuperX Switch(config)# banner incoming $ (Press Return)
Enter TEXT message, End with the character '$'.
Incoming Telnet Session!! $
```

When a user connects to the CLI using Telnet, the following message appears on the Console:

```
Telnet from 209.157.22.63
Incoming Telnet Session!!
```

Syntax: [no] banner incoming <delimiting-character>

To remove the banner, enter the **no banner incoming** command.

Configuring Basic Port Parameters

The procedures in this section describe how to configure the port parameters shown in Table 3.5

Table 3.5: Basic Port Parameters

Port Parameter	See Page
Name	3-13
Speed	3-13
Auto-negotiation maximum port speed advertisement and port speed down-shift	3-14
Duplex mode	3-15
MDI/MDIX detection	3-16
Port status (enable or disable)	3-16
Flow control	3-17
Gigabit negotiate mode	3-18
QoS priority	3-18
Dynamic configuration of Voice over IP (VoIP) phones	3-18

All Foundry ports are pre-configured with default values that allow the device to be fully operational at initial startup without any additional configuration. However, in some cases, changes to the port parameters may be necessary to adjust to attached devices or other network requirements.

Assigning a Port Name

A port name can be assigned to help identify interfaces on the network. You can assign a port name to physical ports, virtual interfaces, and loopback interfaces.

To assign a name to a port:

```
FESX424 Router(config)# interface e 2
FESX424 Router(config-if-e1000-2)# port-name Marsha
```

Syntax: port-name <text>

The <text> parameter is an alphanumeric string. The name can be up to 64 characters long. The name can contain blanks. You do not need to use quotation marks around the string, even when it contains blanks.

Modifying Port Speed

The Gigabit Ethernet copper ports on the FESX and FWSX are designed to auto-sense and auto-negotiate the speed and mode of the connected device. If the attached device does not support this operation, you can manually enter the port speed to operate at either 10 or 100 Mbps. The default value is 10/100/1000 Auto-sense.

NOTE: You can modify the port speed of copper ports only. This feature does not apply to fiber ports.

Configuration Syntax

To change the port speed of interface 8 from the default of 10/100/1000 auto-sense, to 10 Mbps operating at full-duplex, enter the following:

```
FESX424 Router(config)# interface e 8
FESX424 Router(config-if-e1000-8)# speed-duplex 10-full
```

Syntax: speed-duplex <value>

The <value> can be one of the following:

- 10-full
- 10-half
- 100-full
- 100-half
- auto

The default is auto.

Enabling Auto-negotiation Maximum Port Speed Advertisement and Port Speed Down-shift

Maximum Port speed advertisement and ***Port speed down-shift*** are enhancements to the auto-negotiation feature, a mechanism for accommodating multi-speed network devices by automatically configuring the highest performance mode of inter-operation between two connected devices.

- ***Port speed down-shift*** enables Gigabit copper ports on the Foundry device to establish a link at 1000 Mbps over a 4-pair wire when possible, or to down-shift (reduce the speed) to 100 Mbps if the medium is a 2-pair wire.
- ***Maximum port speed advertisement*** enables you to configure an auto-negotiation maximum speed that Gigabit copper ports on the Foundry device will advertise to the connected device. You can configure a port to advertise a maximum speed of either 100 Mbps or 10 Mbps. When the maximum port speed advertisement feature is enabled on a port that is operating at 100 Mbps maximum speed, the port will advertise 10/100 Mbps capability to the connected device. Similarly, if a port is operating at 10 Mbps maximum speed, the port will advertise 10 Mbps capability to the connected device.

The port speed down-shift and maximum port speed advertisement features operate dynamically at the physical link layer between two connected network devices. It examines the cabling conditions and the physical capabilities of the remote link, then configures the speed of the link segment according to the highest physical-layer technology that both devices can accommodate.

The port speed down-shift and maximum port speed advertisement features operate dynamically at the physical link layer, independent of logical trunk group configurations. Although Foundry recommends that you use the same cable types and auto-negotiation configuration on all members of a trunk group, you could utilize the auto-negotiation features conducive to your cabling environment. For example, in certain circumstances, you could configure each port in a trunk group to have its own auto-negotiation maximum port speed advertisement or port speed down-shift configuration.

Application Notes

- This feature is available in software release 02.3.01 and later.
- Port speed down-shift and maximum port speed advertisement work only when auto-negotiation is enabled (CLI command **speed-duplex auto**). If auto-negotiation is OFF, the device will reject the port speed down-shift and maximum port speed advertisement configuration.
- When port speed down-shift or maximum port speed advertisement is enabled on a port, the device will reject any configuration attempts to set the port to a forced speed mode (100 Mbps or 1000 Mbps).
- When the port speed down-shift feature is enabled on a combo port, the port will not support true media automatic detection, meaning the device will not be able to detect and select the fiber or copper connector

based on link availability.

Enabling Port Speed Down-Shift

To enable port speed down-shift on a port that has auto-negotiation enabled, enter a command such as the following at the Global CONFIG level of the CLI:

```
FESX424 Switch(config)# link-config gig copper autoneg-control down-shift e 1 e 2
```

The above command configures Gigabit copper ports 1 and 2 to establish a link at 1000 Mbps over a 4-pair wire when possible, or to down-shift (reduce the speed) to 100 Mbps when the medium is a 2-pair wire.

Syntax: [no] link-config gig copper autoneg-control down-shift ethernet [<slotnum>/]<portnum> [ethernet [<slotnum>/]<portnum>]

You can enable port speed down-shift on one or two ports at a time.

To disable port speed down-shift after it has been enabled, enter the **no** form of the command.

Configuring Maximum Port Speed Advertisement

To configure a maximum port speed advertisement of 10 Mbps on a port that has auto-negotiation enabled, enter a command such as the following at the Global CONFIG level of the CLI:

```
FESX424 Switch(config)# link-config gig copper autoneg-control 10m e 1
```

To configure a maximum port speed advertisement of 100 Mbps on a port that has auto-negotiation enabled, enter the following command at the Global CONFIG level of the CLI:

```
FESX424 Switch(config)# link-config gig copper autoneg-control 100m e 2
```

Syntax: [no] link-config gig copper autoneg-control 10m | 100m ethernet [<slotnum>/]<portnum> [ethernet [<slotnum>/]<portnum>]

You can enable maximum port speed advertisement on one or two ports at a time.

To disable maximum port speed advertisement after it has been enabled, enter the **no** form of the command.

Modifying Port Duplex Mode

You can manually configure a 10/100 Mbps port to accept either full-duplex (bi-directional) or half-duplex (uni-directional) traffic.

NOTE: You can modify the port duplex mode of copper ports only. This feature does not apply to fiber ports.

Port duplex mode and port speed are modified by the same command.

Configuration Syntax

To change the port speed of interface 8 from the default of 10/100/1000 auto-sense to 10 Mbps operating at full-duplex, enter the following:

```
FESX424 Switch(config)# interface e 8
FESX424 Switch(config-if-e1000-8)# speed-duplex 10-full
```

Syntax: speed-duplex <value>

The <value> can be one of the following:

- 10-full
- 10-half
- 100-full
- 100-half
- auto

The default is auto.

Configuring MDI/MDIX

The Foundry FastIron devices support automatic Media Dependent Interface (MDI) and Media Dependent Interface Crossover (MDIX) detection on all Gigabit Ethernet Copper ports.

MDI/MDIX is a type of Ethernet port connection using twisted pair cabling. The standard wiring for end stations is MDI, whereas the standard wiring for hubs and switches is MDIX. MDI ports connect to MDIX ports using straight-through twisted pair cabling. For example, an end station connected to a hub or a switch uses a straight-through cable. MDI-to-MDI and MDIX-to-MDIX connections use crossover twisted pair cabling. So, two end stations connected to each other, or two hubs or switches connected to each other, use crossover cable.

The auto MDI/MDIX detection feature can automatically correct errors in cable selection, making the distinction between a straight-through cable and a crossover cable insignificant.

Configuration Notes

- This feature applies to copper ports only.
- The **mdi-mdix auto** command works only when auto-negotiation is ON. If auto-negotiation is OFF and you enter the command **mdi-mdix auto**, the device automatically resets the port to an MDIX only port. In this case, although the Foundry device does not apply the **mdi-mdix auto** configuration, it accepts and saves it. Consequently, when auto-negotiation is turned back ON, the Foundry device applies the **mdi-mdix auto** configuration.
- The **mdi-mdix mdi** and **mdi-mdix mdix** commands work independently of auto-negotiation. Thus, these commands work whether auto-negotiation is turned ON or OFF.
- Do not use the **mdi-mdix** commands on ports that are manually configured with a speed/duplex of **100-full**. In this case, make sure the other port (remote end of the connection) is also configured to 100-full and a cross-over cable is used if the connected device is another switch, hub, or router, or a straight-through cable if the connected device is a host NIC.

Configuration Syntax

The auto MDI/MDIX detection feature is enabled on all Gigabit copper ports by default. For each port, you can disable auto MDI/MDIX, designate the port as an MDI port, or designate the port as an MDIX port.

To turn off automatic MDI/MDIX detection and define a port as an MDI only port:

```
FESX424 Router(config-if-e1000-2)# mdi-mdix mdi
```

To turn off automatic MDI/MDIX detection and define a port as an MDIX only port:

```
FESX424 Router(config-if-e1000-2)# mdi-mdix mdix
```

To turn on automatic MDI/MDIX detection on a port that was previously set as an MDI or MDIX port:

```
FESX424 Router(config-if-e1000-2)# mdi-mdix auto
```

Syntax: mdi-mdix <mdi | mdix | auto>

After you enter the **mdi-mdix** command, the Foundry device resets the port and applies the change.

To display the MDI/MDIX settings, including the configured value and the actual resolved setting (for **mdi-mdix auto**), enter the command **show interface** at any level of the CLI.

Disabling or Re-Enabling a Port

A port can be made inactive (disable) or active (enable) by selecting the appropriate status option. The default value for a port is enabled.

To disable port 8 of a Foundry device, enter the following:

```
FESX424 Switch(config)# interface e 8
FESX424 Switch(config-if-e1000-8)# disable
```

Syntax: disable

You also can disable or re-enable a virtual interface. To do so, enter commands such as the following:


```
FESX424 Switch(config)# interface ve v1
FESX424 Switch(config-vif-1)# disable
```

Syntax: disable

To re-enable a virtual interface, enter the **enable** command at the Interface configuration level. For example, to re-enable virtual interface v1, enter the following command:

```
FESX424 Switch(config-vif-1)# enable
```

Syntax: enable

Disabling or Re-Enabling Flow Control

You can configure full-duplex ports on a system to operate with or without flow control (802.3x). Flow control is enabled by default.

To disable flow control on full-duplex ports on a system, enter the following:

```
FESX424 Switch(config)# no flow-control
```

To turn the feature back on:

```
FESX424 Switch(config)# flow-control
```

Syntax: [no] flow-control

Enabling and Disabling Support for 100BaseFX

Foundry's FESX424HF and FSX 100/1000 Interface module support 100BaseFX fiber transceivers. After you physically install a 100BaseFX transceiver, you must enter a CLI command to enable it. Note that the CLI syntax for enabling and disabling 100BaseFX support on the FSX differs than on the FESX. Follow the appropriate instructions, below.

FESX424HF

The FESX424HF and 100BaseFX SFP was introduced in software release 02.3.01.

NOTE: The following procedure applies to the FESX424HF device only. The CLI syntax for enabling and disabling 100BaseFX support on the FESX424HF module differs than on the FSX 100/1000 interface module. Make sure you refer to the appropriate procedures.

Foundry's FESX424HF device supports the following types of SFPs for 100BaseFX:

- Multimode SFP – maximum distance is 2 kilometers
- Bidirectional singlemode SFP – maximum distance is 10 kilometers

To enable support for 100BaseFX on a fiber port, enter the following command at the Global CONFIG level of the CLI:

```
FESX424HF Switch(config)# link-config gig fiber 100base-fx e 16
```

The above command enables 100BaseFX on port 16.

The following command enables 100BaseFX on ports 23 and 24

```
FESX424HF Switch(config)# link-config gig fiber 100base-fx e 23 e 24
```

Syntax: [no] link-config gig fiber 100base-fx ethernet <portnum> ethernet <portnum>

You can specify one or two ethernet ports at a time, as shown in the above examples.

To disable 100BaseFX support on a fiber port, enter the **no** form of the command. Note that you must disable 100BaseFX support before inserting a different type of module In the same port. Otherwise, the device will not recognize traffic traversing the port.

FSX 100/1000 Interface Module

The FSX 100/1000 Interface module and 100BaseFX SFP was introduced in software release 02.4.00.

NOTE: The following procedure applies to the FSX 100/1000 Fiber interface module only. The CLI syntax for enabling and disabling 100BaseFX support on the FSX differs than on the FESX. Make sure you refer to the appropriate procedures.

The FSX 100/1000 fiber interface module supports the following types of SFPs for 100BaseFX:

- Multimode SFP – maximum distance is 2 kilometers
- Bidirectional single mode SFP – maximum distance is 10 kilometers

To enable support for 100BaseFX on an FSX fiber port, enter commands such as the following:

```
FastIron SuperX Switch(config)interface e 1/6
FastIron SuperX Switch(config-if-1/6)# 100-fx
```

The above commands enable 100BaseFX on port 6 in slot 1.

Syntax: [no] 100-fx

To disable 100BaseFX support on a fiber port, enter the **no** form of the command. Note that you must disable 100BaseFX support before inserting a different type of module In the same port. Otherwise, the device will not recognize traffic traversing the port.

Changing the Gigabit Fiber Negotiation Mode

The globally configured Gigabit negotiation mode is the default mode for all Gigabit fiber ports. You can override the globally configured default and set individual ports to the following:

- Negotiate-full-auto – The port first tries to perform a handshake with the other port to exchange capability information. If the other port does not respond to the handshake attempt, the port uses the manually configured configuration information (or the defaults if an administrator has not set the information). This is the default.
- Auto-Gigabit – The port tries to perform a handshake with the other port to exchange capability information.
- Negotiation-off – The port does not try to perform a handshake. Instead, the port uses configuration information manually configured by an administrator.

To change the mode for individual ports, enter commands such as the following:

```
FESX424 Switch(config)# int ethernet 1 to 4
FESX424 Switch(config-mif-1-4)# gig-default auto-gig
```

This command overrides the global setting and sets the negotiation mode to auto-Gigabit for ports 1 – 4.

Syntax: gig-default neg-full-auto | auto-gig | neg-off

Modifying Port Priority (QoS)

You can give preference to the inbound traffic on specific ports by changing the Quality of Service (QoS) level on those ports. For information and procedures, see the chapter “Configuring Quality of Service” on page 13-1.

Enabling Dynamic Configuration of Voice over IP (VoIP) Phones

You can configure a FastIron device to automatically detect and re-configure a VoIP phone when it is physically moved from one port to another within the same device. To do so, you must configure a ***voice VLAN ID*** on the port to which the VoIP phone is connected. The software stores the voice VLAN ID in the port’s database for retrieval by the VoIP phone.

The dynamic configuration of a VoIP phone works in conjunction with the VoIP phone’s discovery process. Upon installation, and sometimes periodically, a VoIP phone will query the Foundry device for VoIP information and will advertise information about itself, such as, device ID, port ID, and platform. When the Foundry device receives the

VoIP phone's query, it sends the voice VLAN ID in a reply packet back to the VoIP phone. The VoIP phone then configures itself within the voice VLAN.

As long as the port to which the VoIP phone is connected has a voice VLAN ID, the phone will configure itself into that voice VLAN. If you change the voice VLAN ID, the software will immediately send the new ID to the VoIP phone, and the VoIP phone will re-configure itself with the new voice VLAN.

Configuration Notes

- This feature is supported in software releases 02.2.00 and later for the FESX, FSX, and FWSX devices.
- This feature works with any VoIP phone that:
  - Runs CDP
  - Sends a VoIP VLAN query message
  - Can configure its voice VLAN after receiving the VoIP VLAN reply
- Automatic configuration of a VoIP phone will not work if one of the following applies:
  - You do not configure a voice VLAN ID for a port with a VoIP phone
  - You remove the configured voice VLAN ID from a port without configuring a new one
  - You remove the port from the voice VLAN
- Make sure the port is able to intercept CDP packets (**cdp run** command).
- Some VoIP phones may require a reboot after configuring or re-configuring a voice VLAN ID. For example, if your VoIP phone queries for VLAN information only once upon boot up, you must reboot the VoIP phone before it can accept the VLAN configuration. If your phone is powered by a PoE device, you can reboot the phone by disabling then re-enabling the port.

Enabling Dynamic Configuration of a Voice over IP (VoIP) phone

You can create a voice VLAN ID for a port, or for a group of ports.

To create a voice VLAN ID for a port, enter commands such as the following:

```
FESX424 Switch(config)# interface e 2
FESX424 Switch(config-if-e1000-2)# voice-vlan 1001
```

To create a voice VLAN ID for a group of ports, enter commands such as the following:

```
FESX424 Switch(config)# interface e 1-8
FESX424 Switch(config-mif-1-8)# voice-vlan 1001
```

Syntax: [no] voice-vlan <voice-vlan-num>

where <voice-vlan-num> is a valid VLAN ID between 1 – 4095.

To remove a voice VLAN ID, use the **no** form of the command.

Viewing Voice VLAN Configurations

You can view the configuration of a voice VLAN for a particular port or for all ports.

To view the voice VLAN configuration for a port, use the **show voice-vlan** <port-num> command. The following example shows the command output results.

```
FESX424 Switch(config)# show voice-vlan ethernet 2
Voice vlan ID for port 2: 1001
```

The following example shows the message that appears when the port does not have a configured voice VLAN.

```
FESX424 Switch(config)# show voice-vlan ethernet 2

Voice vlan is not configured for port 2.
```

To view the voice VLAN for all ports, use the **show voice-vlan** command. The following example shows the command output results.

```
FESX424 Switch(config)# show voice-vlan

Port ID          Voice-vlan
2                1001
8                150
15               200
```

Syntax: show voice-vlan [<port-num>]

Chapter 4
Configuring Basic Layer 2 Features

The procedures in this chapter describe how to configure basic Layer 2 parameters.

Foundry devices are configured at the factory with default parameters that allow you to begin using the basic features of the system immediately. However, many of the advanced features such as VLANs or routing protocols for the device must first be enabled at the system (global) level before they can be configured. If you use the Command Line Interface (CLI) to configure system parameters, you can find these system level parameters at the Global CONFIG level of the CLI.

This chapter contains the topics listed in Table 4.1.

Table 4.1: List of Basic Layer 2 Features

Basic Layer 2 Feature	See Page
About port regions	4-2
Spanning Tree Protocol (STP)	4-2
Aging time for learned MAC address entries	4-3
Static, non-aging MAC address entries	4-4
Port-based VLANs	4-4
MAC address filters	4-5
Port locks	4-8
System parameters	4-8
Mirror ports (for traffic diagnosis and troubleshooting)	4-12

NOTE:

- Before assigning or modifying any router parameters, you must assign the IP subnet (interface) addresses for each port.
- For information about configuring IP addresses, DNS resolver, DHCP assist, and other IP-related parameters, see the chapter “Configuring IP” on page 16-1.

- For information about the Syslog buffer and messages, see “Using Syslog” on page A-1.

About Port Regions

Ports on the X-Series devices are grouped into regions. For a few features, you will need to know the region to which a port belongs. However, for most features, a port’s region does not affect configuration or operation of the feature.

NOTE: Port regions do not apply to trunk group configurations on the X-Series devices. However, port regions do apply to port monitoring and unknown unicast configurations.

FastIron Edge Switch X424 and X424HF, and FastIron Workgroup Switch X424:

- Ports 1 – 12
- Ports 13 – 24
- Port 25
- Port 26

FastIron Edge Switch X448 and FastIron Workgroup Switch X448:

- Ports 1 – 12
- Ports 13 – 24
- Port 25 – 36
- Port 37 – 48
- Port 49
- Port 50

FastIron SuperX:

- Management Module:
  - Ports 1 – 12
- 24-port Gigabit Ethernet Copper Interface Module
  - Ports 1 – 12
  - Ports 13 – 24
- 24-port Gigabit Ethernet Fiber Interface Module:
  - Ports 1 – 12
  - Ports 13 – 24
- 2-port 10-Gigabit Ethernet Fiber Interface Module
  - Port 1
  - Port 2

Enabling or Disabling the Spanning Tree Protocol (STP)

STP (IEEE 802.1d bridge protocol) is supported on all Foundry devices. STP detects and eliminates logical loops in the network. STP also ensures that the least cost path is taken when multiple paths exist between ports or VLANs. If the selected path fails, STP searches for and then establishes an alternate path to prevent or limit retransmission of data.

NOTE: This section provides instructions for enabling and disabling STP. For configuration procedures and information about Foundry's IronClad STP, see the chapter "Configuring Spanning Tree Protocol (STP) and IronSpan Features" on page 7-1 in this guide.

STP must be enabled at the system level to allow assignment of this capability on the VLAN level. On devices running Layer 2 code, STP is enabled by default. On devices running Layer 3 code, STP is disabled by default.

To enable STP for all ports on a Foundry device:

```
FESX424 Switch(config)# spanning tree
```

Syntax: [no] spanning-tree

You can also enable and disable spanning tree on a port-based VLAN and on an individual port basis, and enable advanced STP features. See "Configuring Spanning Tree Protocol (STP) and IronSpan Features" on page 7-1.

Modifying STP Bridge and Port Parameters

You can modify the following STP Parameters:

- Bridge parameters – forward delay, maximum age, hello time, and priority
- Port parameters – priority and path cost

For configuration details, see "Changing STP Bridge and Port Parameters" on page 7-5.

Changing the MAC Age Time

By default, learned MAC entries do not age out until they are unused for 300 – 600 seconds. You can change the MAC age time by entering the following command:

```
FESX424 Router(config)# mac-age-time 60
```

Syntax: [no] mac-age-time <secs>

You can configure 0 or a value from 60 – 600 (seconds), in 60-second intervals. If you set the MAC age time to 0, aging is disabled.

NOTE: The actual age time is from one to two times the configured value. For example, if you set the MAC age time to 60 seconds, learned MAC entries age out after remaining unused for between 60 – 120 seconds.

To display the MAC table, enter the following command:

```
FESX424 Router(config)# show mac-address
Total active entries from all ports = 3
Total static entries from all ports = 1
  MAC-Address     Port      Type    VLAN
1234.1234.1234      15    Static       1
0004.8038.2f24      14   Dynamic       1
0004.8038.2f00      13   Dynamic       1
0010.5a86.b159      10   Dynamic       1
```

In the output of the **show mac-address** command, the *Type* column indicates whether the MAC entry is static or dynamic. A static entry is one you create using the **static-mac-address** command. A dynamic entry is one that is learned by the software from network traffic.

The output of the **show mac-address** command on FESX, FSX, and FWSX devices include an *Index* column which indicates the index where the entry exists in the hardware MAC table.

Configuring Static MAC Entries

Static MAC addresses can be assigned to Foundry devices.

NOTE: Foundry devices running Layer 3 code also support the assignment of static IP Routes, static ARP, and static RARP entries. For details on configuring these types of static entries, see “Configuring Static Routes” on page 16-32 and “Creating Static ARP Entries” on page 16-28.

You can manually input the MAC address of a device to prevent it from being aged out of the system address table.

This option can be used to prevent traffic for a specific device, such as a server, from flooding the network with traffic when it is down. Additionally, the static MAC address entry is used to assign higher priorities to specific MAC addresses.

You can specify traffic priority (QoS) and VLAN membership (VLAN ID) for the MAC Address as well as specify device type of either router or host.

The default and maximum configurable MAC table sizes can differ depending on the device. To determine the default and maximum MAC table sizes for your device, display the system parameter values. See “Displaying and Modifying System Parameter Default Settings” on page 4-8.

Command Syntax

To add a static entry for a server with a MAC address of 1145.5563.67FF and a priority of 7 to port 2, enter the following command:

```
FESX424 Switch(config)# static-mac-address 1145.5563.67FF e 2 priority 7
```

Syntax: [no] static-mac-address <mac-addr> ethernet [<slotnum>/]<portnum> priority <num>

The <slotnum> parameter is required on chassis devices.

The priority <num> can be 0 – 7 (0 is lowest priority and 7 is highest priority). The default priority is 0. The default type is host-type.

NOTE: The location of the **static-mac-address** command in the CLI depends on whether you configure port-based VLANs on the device. If the device does not have more than one port-based VLAN (VLAN 1, which is the default VLAN that contains all the ports), the **static-mac-address** command is at the global CONFIG level of the CLI. If the device has more than one port-based VLAN, then the **static-mac-address** command is not available at the global CONFIG level. In this case, the command is available at the configuration level for each port-based VLAN.

Enabling Port-Based VLANs

When using the CLI, port and protocol-based VLANs are created by entering one of the following commands at the global CONFIG level of the CLI.

To create a port-based VLAN, enter commands such as the following:

```
FESX424 Router(config)# vlan 222 by port
FESX424 Router(config)# vlan 222 name Mktg
```

Syntax: vlan <num> by port

Syntax: vlan <num> name <string>

The <num> parameter specifies the VLAN ID. The valid range for VLAN IDs starts at 1 on all systems but the upper limit of the range differs depending on the device. In addition, you can change the upper limit on some devices using the **system max-vlans...** command. See the *Foundry Switch and Router Command Line Interface Reference*.

The <string> parameter is the VLAN name and can be a string up to 32 characters. You can use blank spaces in the name if you enclose the name in double quotes (for example, “Product Marketing”.)

You can configure up to 4063 port-based VLANs on a device running Layer 2 code or 4061 port-based VLANs on a device running Layer 3 code. Each port-based VLAN can contain either tagged or untagged ports. A port cannot be a member of more than one port-based VLAN unless the port is tagged. On both device types, valid VLAN IDs are 1 – 4095. You can configure up to the maximum number of VLANs within that ID range.

NOTE: VLAN ID 4094 is reserved for use by Single STP. VLAN IDs 4091 and 4092 are reserved for use in the Layer 3 Switch and Base Layer 3 images. You can configure these VLAN IDs in the Layer 2 Switch image.

NOTE: The second command is optional and also creates the VLAN if the VLAN does not already exist. You can enter the first command after you enter the second command if you first exit to the global CONFIG level of the CLI.

Assigning IEEE 802.1Q Tagging to a Port

When a port is tagged, it allows communication among the different VLANs to which it is assigned. A common use for this might be to place an email server that multiple groups may need access to on a tagged port, which in turn, is resident in all VLANs that need access to the server.

NOTE: Tagging does not apply to the default VLAN.

When using the CLI, ports are defined as either tagged or untagged at the VLAN level.

Command Syntax

Suppose you want to make port 5 a member of port-based VLAN 4, a tagged port. To do so, enter the following:

```
FESX424 Router(config)# vlan 4
FESX424 Router(config-vlan-4)# tagged e 5
```

Syntax: tagged ethernet [<slotnum>/]<portnum> [to [<slotnum>/]<portnum> [ethernet [<slotnum>/]<portnum>...]]

The <slotnum> parameter is required on chassis devices.

Defining MAC Address Filters

MAC layer filtering enables you to build access lists based on MAC layer headers in the Ethernet/IEEE 802.3 frame. You can filter on the source and destination MAC addresses. The filters apply to incoming traffic only.

You configure MAC filters globally, then apply them to individual interfaces. To apply MAC filters to an interface, you add the filters to that interface’s MAC filter group.

The device takes the action associated with the first matching filter. If the packet does not match any of the filters in the access list, the default action is to drop the packet. If you want the system to permit traffic by default, you must specifically indicate this by making the last entry in the access list a permit filter. Here is an example:
mac filter <last-index-number> **permit any any**

For devices running Layer 3 code, the MAC filter is applied only to those inbound packets that are to be switched. This includes those ports associated with a virtual routing interface. However, the filter is not applied to the virtual routing interface. It is applied to the physical port.

NOTE: Inbound traffic on a port to which a Layer 2 MAC filter is assigned is sent to the CPU for processing.

When you create a MAC filter, it takes effect immediately. You do not need to reset the system. However, you do need to save the configuration to flash memory to retain the filters across system resets.

For complete MAC filter examples, see the *Foundry Switch and Router Command Line Interface Reference*.

Configuration Notes

- MAC filtering on FastIron devices is performed in hardware.
- Layer 2 MAC filtering on FastIron devices differ from other Foundry devices in that you can only filter on source and destination MAC addresses. Other Foundry devices allow you to also filter on the encapsulation type and frame type.
- Use MAC Layer 2 filters only for switched traffic. If a routing protocol (for example, IP) is configured on an interface, a MAC filter defined on that interface is not applied to inbound packets. If you want to filter inbound route traffic, configure a route filter.
- Layer 2 MAC filtering on the FESX, FSX, and FWSX differs from the FES and BigIron in that MAC filtering applies to all traffic, including management traffic. To exclude management traffic from being filtered, configure a MAC filter that explicitly permits all traffic headed to the management MAC (destination) address. The MAC address for management traffic is always the MAC address of port 1.
- You cannot use Layer 2 filters to filter Layer 4 information. To filter Layer 4 information, use IP access policies. See the appendix “Policies and Filters” on page C-1.
- MAC Layer 2 filters are not supported on tagged ports in the base Layer 3 and full Layer 3 images.

Command Syntax

To configure and apply a MAC filter, enter commands such as the following:

```
FESX424 Switch(config)# mac filter 1 deny 3565.3475.3676 ffff.0000.0000
FESX424 Switch(config)# mac filter 1024 permit any any
FESX424 Switch(config)# int e 1
FESX424 Switch(config-if-e1000-1)# mac filter-group 1
```

These commands configure a filter to deny ARP traffic with a source MAC address that begins with “3565” to any destination. The second filter permits all traffic that is not denied by another filter.

NOTE: Once you apply a MAC filter to a port, the device drops all Layer 2 traffic on the port that does not match a MAC permit filter on the port.

Syntax: mac filter <filter-num> permit | deny any | <H.H.H> any | <H.H.H>

The **permit | deny** argument determines the action the software takes when a match occurs.

The <src-mac> <mask> | **any** parameter specifies the source MAC address. You can enter a specific address value and a comparison mask or the keyword **any** to filter on all MAC addresses. Specify the mask using f’s (ones) and zeros. For example, to match on the first two bytes of the address aabb.ccdd.eeff, use the mask ffff.0000.0000. In this case, the filter matches on all MAC addresses that contain "aabb" as the first two bytes. The filter accepts any value for the remaining bytes of the MAC address. If you specify **any**, do not specify a mask. In this case, the filter matches on all MAC addresses.

The <dest-mac> <mask> | **any** parameter specifies the destination MAC address. The syntax rules are the same as those for the <src-mac> <mask> | **any** parameter.

Syntax: mac filter log-enable

Globally enables logging for filtered packets.

Syntax: mac filter-group log-enable

Enables logging for filtered packets on a specific port.

Syntax: mac filter-group <filter-list>

Applies MAC filters to a port.

NOTE: The filters must be applied as a group. For example, if you want to apply four filters to an interface, they must all appear on the same command line.

NOTE: You cannot add or remove individual filters in the group. To add or remove a filter on an interface, apply the filter group again containing all the filters you want to apply to the port.

NOTE: If you apply a filter group to a port that already has a filter group applied, the older filter group is replaced by the new filter group.

Enabling Logging of Packets Denied by MAC Filters

You can configure the Foundry device to generate Syslog entries and SNMP traps for packets that are denied by Layer 2 MAC filters. You can enable logging of denied packets on a global basis or an individual port basis.

The first time an entry in a MAC filter denies a packet and logging is enabled for that entry, the software generates a Syslog message and an SNMP trap. Messages for packets denied by MAC filters are at the warning level of the Syslog.

When the first Syslog entry for a packet denied by a MAC filter is generated, the software starts a five-minute MAC filter timer. After this, the software sends Syslog messages every five minutes. The messages list the number of packets denied by each MAC filter during the previous five-minute interval. If a MAC filter does not deny any packets during the five-minute interval, the software does not generate a Syslog entry for that MAC filter.

NOTE: For a MAC filter to be eligible to generate a Syslog entry for denied packets, logging must be enabled for the filter. The Syslog contains entries only for the MAC filters that deny packets and have logging enabled.

When the software places the first entry in the log, the software also starts the five-minute timer for subsequent log entries. Thus, five minutes after the first log entry, the software generates another log entry and SNMP trap for denied packets.

Configuration Notes

MAC filter logging is supported in the following FastIron configurations:

- FESX devices running software release 02.1.01 or later
- All FSX devices and associated software releases
- All FWSX devices and associated software releases

These releases support MAC filter logging of management traffic only.

Command Syntax

To configure Layer 2 MAC filter logging globally, enter the following CLI commands at the global CONFIG level:

```
FESX424 Switch(config)# mac filter log-enable
FESX424 Switch(config)# write memory
```

Syntax: [no] mac filter log-enable

To configure Layer 2 MAC filter logging for MAC filters applied to ports 1 and 3, enter the following CLI commands:

```
FESX424 Switch(config)# int ethernet 1
FESX424 Switch(config-if-e1000-1)# mac filter-group log-enable
FESX424 Switch(config-if-e1000-1)# int ethernet 3
FESX424 Switch(config-if-e1000-3)# mac filter-group log-enable
FESX424 Switch(config-if-e1000-3)# write memory
```

Syntax: [no] mac filter-group log-enable

Locking a Port To Restrict Addresses

Address-lock filters allow you to limit the number of devices that have access to a specific port. Access violations are reported as SNMP traps. This feature is disabled by default. A maximum of 2048 entries can be specified for access. The default address count is eight.

Configuration Notes

- Static trunk ports and link-aggregation configured ports on FastIron devices do not support the lock-address option.
- The MAC port security feature is a more robust version of this feature. See "Using the MAC Port Security Feature" in the *Foundry Security Guide*.

Command Syntax

To enable address locking for port 2 and place a limit of 15 entries, enter a command such as the following:

```
FESX424 Switch(config)# lock e 2 addr 15
```

Syntax: lock-address ethernet [<slotnum>/]<portnum> [addr-count <num>]

The <slotnum> parameter is required on chassis devices.

The <num> parameter is a value from 1 – 2048.

Displaying and Modifying System Parameter Default Settings

Foundry devices have default table sizes for the system parameters shown in the following display outputs. The table sizes determine the maximum number of entries the tables can hold. You can adjust individual table sizes to accommodate your configuration needs.

The tables you can configure, as well the defaults and valid ranges for each table, differ depending on the Foundry device you are configuring. To display the adjustable tables on your Foundry device, use the **show default values** command. The following shows example outputs on FESX, FSX, and FWSX devices.

NOTE: If you increase the number of configurable subnet addresses on each port, you might also need to increase the total number of subnets that you can configure on the device.

NOTE: Changing the table size for a parameter reconfigures the device's memory. Whenever you reconfigure the memory on a Foundry device, you must save the change to the startup-config file, then reload the software to place the change into effect.

To display the configurable tables and their defaults and maximum values, enter the following command at any level of the CLI. The following shows an example output on the FESX.

```
FESX424 Router# show default values
sys log buffers:50         mac age time:300 sec       telnet sessions:5

ip arp age:10 min          bootp relay max hops:4     ip ttl:64 hops
ip addr per intf:24


when multicast enabled :
igmp group memb.:140 sec   igmp query:60 sec

when ospf enabled :
ospf dead:40 sec           ospf hello:10 sec          ospf retrans:5 sec
ospf transit delay:1 sec

when bgp enabled :
bgp local pref.:100        bgp keep alive:60 sec      bgp hold:180 sec
bgp metric:10              bgp local as:1             bgp cluster id:0
bgp ext. distance:20       bgp int. distance:200      bgp local distance:200

System Parameters     Default     Maximum     Current
ip-arp                4000        64000       4000
ip-static-arp         512         1024        512
atalk-route           1024        1536        1024
atalk-zone-port       64          255         64
atalk-zone-sys        768         2048        768
multicast-route       64          8192        64
dvmrp-route           2048        32000       2048
dvmrp-mcache          512         4096        512
pim-mcache            1024        4096        1024
igmp-max-group-addr   4096        8192        4096
ip-cache              256000      400000      256000
ip-filter-port        1015        1015        1015
ip-filter-sys         2048        4096        2048
ipx-forward-filter    32          128         32
ipx-rip-entry         2048        8192        2048
ipx-rip-filter        32          128         32
ipx-sap-entry         4096        8192        4096
ipx-sap-filter        32          128         32
l3-vlan               32          1024        32
ip-qos-session        1024        16000       1024
mac                   16000       16000       16000
ip-route              80000       128000      80000
ip-static-route       64          1024        64
vlan                  64          4095        4095
spanning-tree         32          128         32
mac-filter-port       16          256         16
mac-filter-sys        32          512         32
ip-subnet-port        24          128         24
session-limit         65536       160000      65536
view                  10          65535       10
virtual-interface     255         512         255
hw-ip-next-hop       2048         6144        2048
hw-logical-interface4096          4096        4096
hw-ip-mcast-mll      1024         4096        1024
```

The following shows an example output of the **show default values** command on the FSX

```
FastIron SuperX Router# show default values
sys log buffers:50         mac age time:300 sec       telnet sessions:5

ip arp age:10 min          bootp relay max hops:4     ip ttl:64 hops
ip addr per intf:24


when multicast enabled :
igmp group memb.:140 sec   igmp query:60 sec

when ospf enabled :
ospf dead:40 sec           ospf hello:10 sec          ospf retrans:5 sec
ospf transit delay:1 sec

when bgp enabled :
bgp local pref.:100        bgp keep alive:60 sec      bgp hold:180 sec
bgp metric:10              bgp local as:1             bgp cluster id:0
bgp ext. distance:20       bgp int. distance:200      bgp local distance:200
```

System Parameters	Default	Maximum	Current
ip-arp	4000	64000	4000
ip-static-arp	512	1024	512
atalk-route	1024	1536	1024
atalk-zone-port	64	255	64
atalk-zone-sys	768	2048	768
multicast-route	64	8192	64
dvmrp-route	2048	32000	2048
dvmrp-mcache	512	4096	512
pim-mcache	1024	4096	1024
igmp-max-group-addr	4096	8192	4096
ip-cache	256000	400000	256000
ip-filter-port	1015	1015	1015
ip-filter-sys	2048	4096	2048
ipx-forward-filter	32	128	32
ipx-rip-entry	2048	8192	2048
ipx-rip-filter	32	128	32
ipx-sap-entry	4096	8192	4096
ipx-sap-filter	32	128	32
l3-vlan	32	1024	32
ip-qos-session	1024	16000	1024
mac	16000	16000	16000
ip-route	80000	200000	80000
ip-static-route	64	1024	64
vlan	64	4095	4095
spanning-tree	32	255	32
mac-filter-port	16	256	16
mac-filter-sys	32	512	32
ip-subnet-port	24	128	24
session-limit	65536	160000	65536
view	10	65535	10
virtual-interface	255	512	255
hw-ip-next-hop	2048	6144	2048
hw-logical-interface	4096	4096	4096
hw-ip-mcast-mll	1024	4096	1024

The following shows an example output of the **show default values** command on the FWSX

```
FWSX Switch# show default values
sys log buffers:50         mac age time:300 sec       telnet sessions:5

System Parameters     Default     Maximum     Current
igmp-max-group-addr   255         1024        255
l3-vlan               32          1024        32
mac                   16000       16000       16000
vlan                  64          4095        4095
spanning-tree         32          128         32
mac-filter-port       32          256         32
mac-filter-sys        64          512         64
view                  10          65535       10
```

Information for the configurable tables appears under the columns that are shown in bold type in the above examples. To simplify configuration, the command parameter you enter to configure the table is used for the table name. For example, to increase the capacity of the IP route table, enter the following commands:

```
FESX424 Switch(config)# system-max ip-route 120000
FESX424 Switch(config)# write memory
FESX424 Switch(config)# exit
FESX424 Switch# reload
```

NOTE: If you accidentally enter a value that is not within the valid range of values, the CLI will display the valid range for you.

To increase the number of IP subnet interfaces you can configure on each port on a device running Layer 3 code from 24 to 64, then increase the total number of IP interfaces you can configure on the device from 256 to 512, enter the following commands:

```
FESX424 Switch(config)# system-max subnet-per-interface 64
FESX424 Switch(config)# write memory
FESX424 Switch(config)# exit
FESX424 Switch# reload
```

Syntax: system-max subnet-per-interface <num>

The <num> parameter specifies the maximum number of subnet addresses per port and can be from 1 – 64. The default is 24.

Syntax: system-max subnet-per-system <num>

The <num> parameter specifies the maximum number of subnet addresses for the entire device and can be from 1 – 512. The default is 256.

```
FESX424 Switch(config)# system-max subnet-per-system 512
FESX424 Switch(config)# write memory
FESX424 Switch(config)# exit
FESX424 Switch# reload
```

Configuring Port Mirroring and Monitoring

FastIron devices support monitoring of both inbound and outbound traffic on individual ports. To configure port monitoring, specify the ***mirror port***, then enable monitoring on the ***monitored port***.

- The ***mirror port*** is the port to which the monitored traffic is copied. Attach your protocol analyzer to the mirror port.
- The ***monitored port*** is the port whose traffic you want to monitor.

Configuration Considerations

Refer to the following rules when configuring port mirroring and monitoring:

- FESX and FWSX devices support sFlow and inbound port monitoring together on the same device, however, these devices *do not* support port monitoring and sFlow together within the same port region. See the section "About Port Regions" on page 4-2 for a list of valid port ranges on these devices.
- FSX devices running software release 02.2.01 or later support sFlow and inbound port monitoring together on the same device; however, both features cannot coexist within the same port region. See the section "About Port Regions" on page 4-2 for a list of valid port ranges on FSX devices.
- You can configure a mirror port specifically as an ingress port, an egress port, or both.
- You can configure multiple ingress and egress mirror ports. For 1-Gigabit ports, ports in groups of 12 share one ingress mirror port and one egress mirror port. So ports 1 and 2 cannot have different mirror ports, but ports 1 and 13 can. Each 10-Gigabit port can have one ingress mirror port and one egress mirror port.
- You can configure up to eight egress monitored ports.
- You can configure any number of ingress monitored ports.
- Mirror ports can run at any speed and are not related to the speed of the ingress or egress monitored ports.
- The same port cannot be both a monitored port and the mirror port.
- The same port can be monitored by one mirror port for ingress traffic and another mirror port for egress traffic.
- The mirror port cannot be a trunk port.
- The monitored port and its mirror port do not need to belong to the same port-based VLAN.
    - If the mirror port is in a *different* VLAN from the monitored port, the packets are tagged with the monitor port's VLAN ID.
    - If the mirror port is in the *same* VLAN as the monitored port, the packets are tagged or untagged, depending on the mirror port's configuration.
- More than one monitored port can be assigned to the same mirror port.
- If the primary interface of a trunk is enabled for monitoring, the entire trunk will be monitored. You can also enable an individual trunk port for monitoring using the **config-trunk-ind** command.

Command Syntax

To configure port monitoring, enter commands such as the following:

```
FESX424 Switch(config)# mirror-port ethernet 4
FESX424 Switch(config)# interface ethernet 11
FESX424 Switch(config-if-e1000-11)# monitor ethernet 4 both|in|out
```

Syntax: [no] mirror-port ethernet [<slotnum>/]<portnum> [input | output]

Syntax: [no] monitor ethernet [<slotnum>/]<portnum> both | in | out

The <portnum> parameter specifies the mirror port to which the monitored port's traffic will be copied. If you are configuring a chassis device, specify the slot number as well (<slotnum>/<portnum>).

The [input | output] parameters apply to the FESX, FSX, and FWSX devices only. This parameter configures the mirror port exclusively for ingress or egress traffic. If you do not specify one, both types of traffic apply.

The **both | in | out** parameters specify the traffic direction you want to monitor on the mirror port. There is no default.

To display the port monitoring configuration, enter the **show monitor** and **show mirror** commands.

Chapter 5

Configuring Base Layer 3 and Enabling Routing Protocols

The Layer 2 with Base Layer 3 software image contains all the system-level features in the Layer 2 images, along with the following:

- Static IP routes
- RIPv1 and RIPv2 (see note, below)
- Routing between directly connected subnets
- RIP advertisements of the directly connected subnets

NOTE:

- Layer 2 with Base Layer 3 images provide static RIP support. The device does not learn RIP routes from other Layer 3 devices. However, the device does advertise directly connected routes. Foundry Networks recommends that you deploy these devices only at the edge of your network, since incoming traffic can learn directly-connected routes advertised by the Foundry device, but outgoing traffic to other devices must use statically configured or default routes.
- The Base Layer 3 images do not support IP multicasting, OSPF, or BGP4.
- The Base Layer 3 images do not support protocol VLANs.
- FWSX devices are Layer 2 switches only. They do not support Base Layer 3 and full Layer 3 features.

The procedures in this chapter describe how to perform the tasks listed in Table 5.1.

Table 5.1: Procedures in This Chapter

Task	See Page
Adding a static IP route	5-2
Adding a static entry to the ARP table	5-2
Modifying and displaying Layer 3 system parameter limits (FESX and FSX devices only)	5-3
Configuring RIP in the Base Layer 3 software image	5-4
Enabling or disabling other Layer 3 routing protocols in the full Layer 3 software image	5-7

Table 5.1: Procedures in This Chapter (Continued)

Task	See Page
Enabling or disabling Layer 2 switching	5-7

Adding a Static IP Route

To add a static IP route, enter a command such as the following at the global CONFIG level of the CLI:

```
FESX424 Router(config)# ip route 209.157.2.0 255.255.255.0 192.168.2.1
```

This command adds a static IP route to the 209.157.2.x/24 subnet.

Syntax: [no] ip route <dest-ip-addr> <dest-mask> <next-hop-ip-addr> [<metric>]

or

Syntax: [no] ip route <dest-ip-addr>/<mask-bits> <next-hop-ip-addr> [<metric>]

The <dest-ip-addr> is the route's destination. The <dest-mask> is the network mask for the route's destination IP address. Alternatively, you can specify the network mask information by entering a forward slash followed by the number of bits in the network mask. For example, you can enter 192.0.0.0 255.255.255.0 as 192.0.0.0/.24. To configure a default route, enter 0.0.0.0 for <dest-ip-addr> and 0.0.0.0 for <dest-mask> (or 0 for the <mask-bits> if you specify the address in CIDR format). Specify the IP address of the default gateway using the <next-hop-ip-addr> parameter.

The <next-hop-ip-addr> is the IP address of the next-hop router (gateway) for the route.

The <metric> parameter specifies the cost of the route and can be a number from 1 – 16. The default is 1. The metric is used by RIP. If you do not enable RIP, the metric is not used.

NOTE: You cannot specify **null0** or another interface as the next hop in the Base Layer 3 image.

Adding a Static ARP Entry

Static entries are useful in cases where you want to pre-configure an entry for a device that is not connected to the Foundry device, or you want to prevent a particular entry from aging out. The software removes a dynamic entry from the ARP cache if the ARP aging interval expires before the entry is refreshed. Static entries do not age out, regardless of whether the Foundry device receives an ARP request from the device that has the entry's address. The software places a static ARP entry into the ARP cache as soon as you create the entry.

To add a static ARP entry, enter a command such as the following at the global CONFIG level of the CLI:

```
FESX424 Router(config)# arp 1 209.157.22.3 aaaa.bbbb.cccc ethernet 3
```

This command adds a static ARP entry that maps IP address 209.157.22.3 to MAC address aaaa.bbbb.cccc. The entry is for a MAC address connected to FESX424 Router port 3.

Syntax: [no] arp <num> <ip-addr> <mac-addr> ethernet [<slotnum>/]<portnum>

The <num> parameter specifies the entry number. You can specify a number from 1 up to the maximum number of static entries allowed on the device. You can allocate more memory to increase this amount. To do so, enter the **system-max ip-static-arp** <num> command at the global CONFIG level of the CLI.

The <ip-addr> command specifies the IP address of the device that has the MAC address of the entry.

The <mac-addr> parameter specifies the MAC address of the entry.

The <portnum> command specifies the port number attached to the device that has the MAC address of the entry. If you are configuring a chassis device, specify the slot number as well as the port number (<slotnum>/<portnum>).

NOTE: The **clear arp** command clears learned ARP entries but does not remove any static ARP entries.

Modifying and Displaying Layer 3 System Parameter Limits

You can configure the following Layer 3 system parameters:

- number of IP next hops and IP route entries
- number of hardware logical interfaces (physical port and VLAN pairs)
- number of output interfaces (clients)

These parameters are automatically enabled with pre-defined default values. You can however, adjust these values to conform with your network's topology.

To display the current settings for the Layer 3 system parameters, use the **show default value** command. See "Displaying Layer 3 System Parameter Limits" on page 5-4.

To modify the default settings for the Layer 3 system parameters, use the **system max** command at the Global CONFIG level of the CLI. See "Modifying Layer 3 System Parameter Limits" on page 5-3.

Configuration Note

Changing the system parameters reconfigures the device's memory. Whenever you reconfigure the memory on a Foundry device, you must save the change to the startup-config file, then reload the software to place the change into effect.

Modifying Layer 3 System Parameter Limits

The Layer 3 system parameter limits share the same hardware memory space and, by default, consume all of the hardware memory allocated for these Layer 3 limits. Therefore, to increase the limit for one of the parameters, you must first decrease one or both of the other parameters' limits. If you enter a value that exceeds the memory limit, the CLI will display an error message and the configuration will not take effect.

For example, if the network topology has a smaller number of IP next hops and routes, but has numerous multicast output interfaces, you could decrease the number of IP next hops and routes, then increase the number of multicast output interfaces. To do so, enter commands such as the following:

```
FESX424 Router(config)# system-max hw-ip-next-hop 1024
FESX424 Router(config)# system-max hw-ip-mcast-mll 2048
FESX424 Router(config)# write mem
FESX424 Router(config)# reload
```

Likewise, if the network topology does not have a large number of VLANs, and the VLANs configured on physical ports are not widely distributed, you could decrease the number of hardware logical interfaces, then increase the number of IP next hops and multicast output interfaces. To do so, enter commands such as the following:

```
FESX424 Router(config)# system-max hw-logical-interface 2048
FESX424 Router(config)# system-max hw-ip-next-hop 3072
FESX424 Router(config)# system-max hw-ip-mcast-mll 2048
FESX424 Router(config)# write mem
FESX424 Router(config)# reload
```

Syntax: system max hw-ip-next-hop <num>

Syntax: system max hw-logical-interface <num>

Syntax: system max hw-ip-mcast-mll <num>

The **hw-ip-next-hop** <num> parameter specifies the maximum number of IP next hops and routes supported on the device. Note that the maximum number includes unicast next hops and multicast route entries. Enter a number from 100 to 6144. The default is 2048.

The **hw-logical-interface** <num> parameter specifies the number of hardware logical interface pairs (physical port and VLAN pairs) supported on the device. Enter a number from 0 to 4096. When this parameter is set to 4096 (the maximum), the limit is not enforced. If you enter a number less than 4096, the limit is the total number of physical port and VLAN pairs that are IP-enabled in the system. The default is 4096.

The **hw-ip-mcast-mll** <num> parameter specifies the maximum number of multicast output interfaces (clients) supported on the device. If a given source or group has clients in *n* tagged VLANs on the router, then *n* + 1 mll entries are consumed for that source or group entry. Enter a number from 0 to 4096. The default is 1024.

Displaying Layer 3 System Parameter Limits

To display the Layer 3 system parameter defaults, maximum values, and current values, enter the following command at any level of the CLI:

```
FESX424 Router# show default value

sys log buffers:50          mac age time:300 sec       telnet sessions:5

ip arp age:10 min           bootp relay max hops:4     ip ttl:64 hops
ip addr per intf:24

igmp group memb.:140 sec    igmp query:60 sec

ospf dead:40 sec            ospf hello:10 sec          ospf retrans:5 sec
ospf transit delay:1 sec

System Parameters     Default     Maximum     Current
ip-arp                4000        64000       4000
ip-static-arp         512         1024        512

some lines omitted for brevity....


hw-ip-next-hop        2048        6144        2048
hw-logical-interface  4096        4096        4096
hw-ip-mcast-mll       1024        4096        1024
```

Configuring RIP

RIP is disabled by default. If you want the Foundry device to use RIP, you must enable the protocol globally, then enable RIP on individual ports. When you enable RIP on a port, you also must specify the version (version 1 only, version 2 only, or version 1 compatible with version 2).

Optionally, you also can set or change the following parameters:

- Route redistribution – You can enable the software to redistribute static routes from the IP route table into RIP. Redistribution is disabled by default.
- Learning of default routes – The default is disabled.
- Loop prevention (split horizon or poison reverse) – The default is poison reverse.

Enabling RIP

RIP is disabled by default. To enable it, use the following CLI method. You must enable the protocol both globally and on the ports on which you want to use RIP.

To enable RIP globally, enter the following command:


```
FESX424 Router(config)# router rip
```

Syntax: [no] router rip

To enable RIP on a port and specify the RIP version, enter commands such as the following:

```
FESX424 Router(config-rip-router)# interface ethernet 1
FESX424 Router(config-if-e1000-1)# ip rip v1-only
```

This command changes the CLI to the configuration level for port 1and enables RIP version 1 on the interface. You must specify the version.

Syntax: interface ethernet [<slotnum>/]<portnum>

Syntax: [no] ip rip v1-only | v1-compatible-v2 | v2-only

Enabling Redistribution of IP Static Routes into RIP

By default, the software does not redistribute the IP static routes in the route table into RIP. To configure redistribution, perform the following tasks:

- Configure redistribution filters (optional). You can configure filters to permit or deny redistribution for a route based on the route's metric. You also can configure a filter to change the metric. You can configure up to 64 redistribution filters. The software uses the filters in ascending numerical order and immediately takes the action specified by the filter. Thus, if filter 1 denies redistribution of a given route, the software does not redistribute the route, regardless of whether a filter with a higher ID permits redistribution of that route.

  NOTE: The default redistribution action is permit, even after you configure and apply a permit or deny filter. To deny redistribution of specific routes, you must configure a deny filter.

  NOTE: The option to set the metric is not applicable to static routes.

- Enable redistribution.

  NOTE: If you plan to configure redistribution filters, do not enable redistribution until you have configured the filters.

When you enable redistribution, all IP static routes are redistributed by default. If you want to deny certain routes from being redistributed into RIP, configure deny filters for those routes before you enable redistribution. You can configure up to 64 RIP redistribution filters. They are applied in ascending numerical order.

NOTE: The default redistribution action is still permit, even after you configure and apply redistribution filters to the port. If you want to tightly control redistribution, apply a filter to deny all routes as the last filter (filter ID 64), then apply filters with lower filter IDs to allow specific routes.

To configure a redistribution filter, enter a command such as the following:

```
FESX424 Router(config-rip-router)# deny redistribute 1 static address 207.92.0.0
255.255.0.0
```

This command denies redistribution of all 207.92.x.x IP static routes.

Syntax: [no] permit | deny redistribute <filter-num> static address <ip-addr> <ip-mask>
[match-metric <value> | set-metric <value>]

The <filter-num> specifies the redistribution filter ID. Specify a number from 1 – 64. The software uses the filters in ascending numerical order. Thus, if filter 1 denies a route from being redistributed, the software does not redistribute that route even if a filter with a higher ID permits redistribution of the route.

The **address** <ip-addr> <ip-mask> parameters apply redistribution to the specified network and subnet address. Use 0 to specify "any". For example, "207.92.0.0 255.255.0.0" means "any 207.92.x.x subnet". However, to specify any subnet (all subnets match the filter), enter "address 255.255.255.255 255.255.255.255".

The **match-metric** <value> parameter applies redistribution to those routes with a specific metric value; possible values are from 1 – 15.

The **set-metric** <value> parameter sets the RIP metric value that will be applied to the routes imported into RIP.

NOTE: The **set-metric** parameter does not apply to static routes.

The following command denies redistribution of a 207.92.x.x IP static route only if the route's metric is 5.

```
FESX424 Router(config-rip-router)# deny redistribute 2 static address 207.92.0.0
255.255.0.0 match-metric 5
```

The following commands deny redistribution of all routes except routes for 10.10.10.x and 20.20.20.x:

```
FESX424 Router(config-rip-router)# deny redistribute 64 static address
255.255.255.255 255.255.255.255
FESX424 Router(config-rip-router)# permit redistribute 1 static address 10.10.10.0
255.255.255.0
FESX424 Router(config-rip-router)# permit redistribute 2 static address 20.20.20.0
255.255.255.0
```

Enabling Redistribution

After you configure redistribution parameters, you need to enable redistribution.

To enable RIP redistribution, enter the following command:

```
FESX424 Router(config-rip-router)# redistribution
```

Syntax: [no] redistribution

Enabling Learning of Default Routes

By default, the software does not learn RIP default routes.

To enable learning of default RIP routes, enter commands such as the following:

```
FESX424 Router(config)# interface ethernet 1
FESX424 Router(config-if-e1000-1)# ip rip learn-default
```

Syntax: interface ethernet [<slotnum>/]<portnum>

Syntax: [no] ip rip learn-default

The <slotnum>/ parameter applies to chassis devices only.

Changing the Route Loop Prevention Method

RIP can use the following methods to prevent routing loops:

- Split horizon – The Foundry device does not advertise a route on the same interface as the one on which it learned the route.
- Poison reverse – The Foundry device assigns a cost of 16 ("infinite" or "unreachable") to a route before advertising it on the same interface as the one on which it learned the route. This is the default.

NOTE: These methods are in addition to RIP's maximum valid route cost of 15.

To enable split horizon, enter commands such as the following:

```
FESX424 Router(config)# interface ethernet 1
FESX424 Router(config-if-e1000-1)# no ip rip poison-reverse
```

Syntax: [no] ip rip poison-reverse

Other Layer 3 Protocols

For information about other IP configuration commands in the Layer 2 with Base Layer 3 image that are not included in this chapter, see the chapter “Configuring IP” on page 16-1.

For information about enabling or disabling Layer 3 routing protocols, see “Enabling or Disabling Routing Protocols” on page 5-7. For complete configuration information about the routing protocols, see the other chapters in this book.

Enabling or Disabling Routing Protocols

This section describes how to enable or disable routing protocols. For complete configuration information about the routing protocols, see the other chapters in this book.

FESX and FSX devices running full Layer 3 code support the following protocols:

- BGP4
- IGMP
- IP
- IP multicast (DVMRP, PIM-SM, PIM-DM)
- OSPF
- RIPV1 and V2
- VRRP
- VRRPE
- VSRP

IP routing is enabled by default on devices running Layer 3 code. All other protocols are disabled, so you must enable them to configure and use them.

NOTE: The following protocols require a system reset before the protocol will be active on the system: PIM, DVMRP, and RIP. To reset a system, enter the **reload** command at the privileged level of the CLI.

To enable a protocol on a device running full Layer 3 code, enter **router** at the global CONFIG level, followed by the protocol to be enabled. The following example shows how to enable OSPF:

```
FESX424 Router(config)# router ospf
FESX424 Router(config)# end
FESX424 Router# write memory
FESX424 Router# reload
```

Syntax: router bgp | dvmrp | ospf | pim | rip | vrrp | vrrpe | vsrp

Enabling or Disabling Layer 2 Switching

By default, Foundry Layer 3 Switches support Layer 2 switching. These devices switch the routing protocols that are not supported on the devices. If you want to disable Layer 2 switching, you can do so globally or on individual ports, depending on the version of software your device is running.

Configuration Notes

- Make sure you really want to disable all Layer 2 switching operations before you use this option. Consult your reseller or Foundry Networks for information.
- This feature is supported in the following configurations:
  - The FESX running software release 01.1.00 or prior, supports disabling Layer 2 switching on a global basis only. Starting in release 02.1.01, the FESX supports disabling Layer 2 switching on an individual

interface as well as on a global basis.

- The FSX running software release 02.2.00 or later supports disabling Layer 2 switching on an individual interface as well as on a global basis.

Command Syntax

To globally disable Layer 2 switching on a Layer 3 Switch, enter commands such as the following:

```
FESX424 Router(config)# route-only
FESX424 Router(config)# exit
FESX424 Router# write memory
FESX424 Router# reload
```

To re-enable Layer 2 switching on a Layer 3 Switch, enter the following:

```
FESX424 Router(config)# no route-only
FESX424 Router(config)# exit
FESX424 Router# write memory
FESX424 Router# reload
```

Syntax: [no] route-only

To disable Layer 2 switching only on a specific interface, go to the Interface configuration level for that interface, then disable the feature. The following commands show how to disable Layer 2 switching on port 2:

```
FESX424 Router(config)# interface ethernet 2
FESX424 Router(config-if-e1000-2)# route-only
```

Syntax: [no] route-only

To re-enable Layer 2 switching, enter the command with “no”, as in the following example:

```
FESX424 Router(config-if-e1000-2)# no route-only
```


Chapter 6
Configuring Power Over Ethernet

This chapter provides an overview of Power over Ethernet (POE) and describes how to enable or disable POE and how to configure POE parameters using CLI commands.

NOTE: This chapter applies to POE devices only.

This chapter contains the topics listed in Table 6.1.

Table 6.1: Chapter Contents

Power over Ethernet Overview

This section provides an overview of the requirements for delivering power over the LAN, as defined by the Institute of Electrical and Electronics Engineers Inc. (IEEE) in the 802.3af specification.

Foundry's FSX (with POE daughter card) provides Power over Ethernet, compliant with the standards described in the IEEE 802.3af specification for delivering in-line power. The 802.3af specification defines the standard for delivering power over existing network cabling infrastructure, enabling multicast-enabled full streaming audio and video applications for converged services, such as, Voice over IP (VoIP), WLAN access points, IP surveillance cameras, and other IP technology devices.

POE technology eliminates the need for an electrical outlet and dedicated UPS near IP powered devices. With power sourcing devices, such as Foundry's FSX, power is consolidated and centralized in the wiring closets, improving the reliability and resiliency of the network. Because POE can provide power over Ethernet cable, power is continuous, even in the event of a power failure.

Terms Used in This Section

The following terms are introduced in this section:

- **Power sourcing device/equipment** - This is the source of the power, or the device that integrates the power onto the network. Power sourcing devices/equipment have embedded POE technology. In this case, the power sourcing device is Foundry's FSX.
- **IP powered device** or **Power consuming device** - This is the Ethernet device that requires power and is situated on the other end of the cable opposite the power sourcing equipment.

Methods for Delivering POE

There are two methods for delivering power over the network, as defined in the 802.3af specification:

- **Endspan** - Power is supplied through the Ethernet ports on a power sourcing device. With the Endspan solution, power can be carried over the two data pairs (Alternative A) or the two spare pairs (Alternative B).
- **Midspan** - Power is supplied by an intermediate power sourcing device placed between the switch and the powered device. With the Midspan solution, power is carried over the two spare pairs (Alternative B).

With both methods, power is transferred over four conductors, between the two pairs. 802.3af-compliant powered devices are able to accept power from either pairs.

Foundry's FSX POE devices use the Endspan method, compliant with the 802.3af standard.

The Endspan and Midspan methods are described in more detail in the following sections.

NOTE: All 802.3af-compliant power consuming devices are required to support both application methods defined in the 802.3af specification.

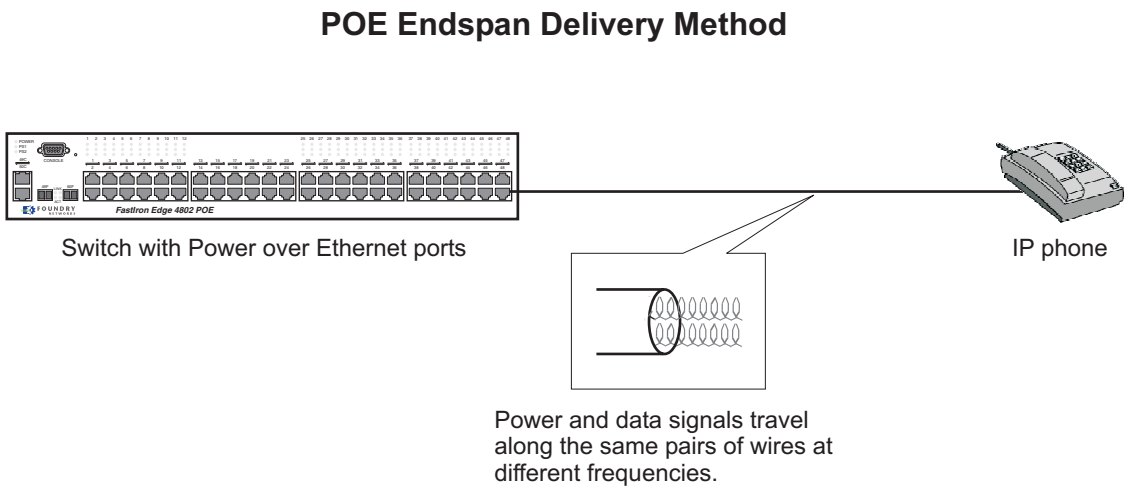
Endspan

The POE Endspan method uses the Ethernet switch ports on power sourcing equipment, such as Foundry's FSX POE, which has embedded POE technology to deliver power over the network.

With the Endspan solution, there are two supported methods of delivering power. In Alternative A, four wires deliver data and power over the network. Specifically, power is carried over the live wire pairs that deliver data, as illustrated in Figure 6.1. In Alternative B, the four wires of the spare pairs are used to deliver power over the network. Foundry's POE devices support Alternative A.

The Endspan method is illustrated in Figure 6.1.

Figure 6.1 POE Endspan Delivery Method

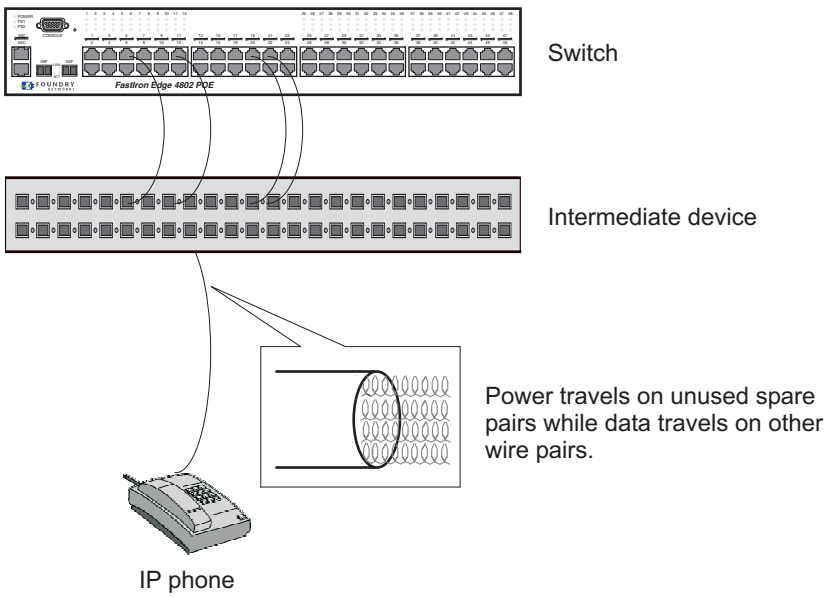


Midspan

The POE Midspan solution uses an intermediate device, usually a powered device, to inject power into the network. The intermediate device is positioned between the switch and the powered device and delivers power over the network using the spare pairs of wires (Alternative B). The intermediate device has multiple channels (typically 6 to 24), and each of the channels has data input and a data plus power RJ-45 output connector.

The Midspan method is illustrated in Figure 6.2.

Figure 6.2 POE Midspan Delivery Method



Autodiscovery

POE autodiscovery is a detection mechanism that identifies whether or not an installed device is 802.3af compatible. When you plug a device into an Ethernet port that is capable of providing in-line power, the autodiscovery mechanism detects whether or not the device requires power and how much power is needed. The autodiscovery mechanism also has a disconnect protection mechanism that shuts down the power once a powered device has been disconnected from the network or when a faulty powered device has been detected. This feature enables safe installation and prevents high-voltage damage to equipment.

POE autodiscovery is achieved by periodically transmitting current or test voltages that can detect when a powered device is attached to the network. When an 802.3af compatible device is plugged into a POE port, the powered device reflects test voltage back to the power sourcing device (the Foundry device), ultimately causing the power to be switched ON. Non-compatible 802.3af devices do not reflect test voltage back to the power sourcing device.

Power Class

Different power classes determine the amount of power a POE powered device receives. When a valid powered device is detected, the Foundry POE device performs power classification by inducing a specific voltage and measuring the current consumption of the powered device. Depending on the measured current, the Foundry device assigns the appropriate class to the powered device. Powered devices that do not support classification are assigned a class of 0 (zero). Table 6.3 shows the different power classes and their respective power consumption needs.

Table 6.2: Power Classes for Powered Devices

Class	Usage	Power (Watts)
0	default	15.4
1	optional	4
2	optional	7
3	optional	15.4
4	future	class 0

Power Specifications

The actual implementation of the 802.3af standard limits power to 15.4W (44V to 57V) from the power sourcing device. This is in compliance with safety standards and existing wiring limitations. Though limited by the 802.3af standard, 15.4 watts of power is ample, as most powered devices consume an average of 5 to 12 watts of power. IP phones, wireless LAN access points, and network surveillance cameras each consume an average of 3.5 to 9 watts of power.

The FSX's 48-volt power supply (part number SX-POE-AC-PWR) provides power to the POE daughter card, and ultimately to POE power-consuming devices. The number of POE power-consuming devices that one 48-volt power supply can support depends on the number of watts required by each power-consuming device. Each 48-volt power supply can provide 1080 watts of power, and each POE port supports a maximum of 15.4 watts of power per POE power-consuming device. For example, if each POE power-consuming device attached to the FSX consumes 10 watts of power, one 48-volt supply will power up to 108 POE ports. You can install a second 48-volt supply for additional POE power. Power supply specifications are covered in the *Foundry FastIron X-Series Chassis Hardware Installation Guide* and in the *Foundry FastIron Stackable Hardware Installation Guide*.

CAUTION: The SX-POE-AC-PWR power supply is designed exclusively for use with the FSX POE devices. The power supply produces extensive power to support 802.3af applications. Installing the power supply in a device other than the FSX POE will cause extensive damage to your equipment.

Cabling Requirements

The 802.3af standard currently supports POE on 10/100/1000 Mbps Ethernet ports operating over standard Category 5 unshielded twisted pair (UTP) cable or better. If your network uses cabling categories less than 5, you cannot implement POE without first upgrading your cables to CAT 5 UTP or better.

Supported Powered Devices

Foundry's FSX POE devices support the following types of IP powered devices:

- Voice over IP (VoIP) phones
- Wireless LAN access points
- IP surveillance cameras

The following sections briefly describe these IP powered devices.

VoIP

Voice over IP (VoIP) is the convergence of traditional telephony networks with data networks, utilizing the existing data network infrastructure as the transport system for both services. Traditionally, voice is transported on a network that uses circuit-switching technology, whereas data networks are built on packet-switching technology. To achieve this convergence, technology has been developed to take a voice signal, which originates as an analog signal and transport it within a digital medium. This is done by devices, such as VoIP Telephones, which receive the originating tones and place them in UDP packets, the size and frequency of which is dependant on the Coding / Decoding (CODEC) technology that has been implemented in the VoIP Telephone / device. The VoIP control packets use the TCP/IP format.

Wireless LAN Access Points

Wireless LANs enable you to establish and maintain a wireless network connection within or between buildings, without the constraints of wires or cables as imposed by a wired LAN. Wireless LAN access points provide the link between the wired LAN and the wireless LAN.

Foundry's IronPoint™ Access Point allows wireless clients to connect to your enterprise network. It is a full-featured access point that can be managed as a single device or by IronView Network Manager, a network management tool that manages several Foundry devices on a network. For more information about Foundry's IronPoint Access Point, see the *Foundry IronPoint Wireless LAN Configuration Guide*.

One of the main concerns with wireless LAN access points is the additional protection needed to secure the network. To help ensure continuous security against unauthorized Wireless LAN Access Points deployment, and deliver advanced security for entry-level WLAN Access Points, the Foundry's POE devices include IEEE 802.1x support for a flexible and dynamic security implementation. All switch ports can be configured as secured, requiring 802.1x authentication, or unsecured, requiring no authentication. For more information about this feature, refer to the *Foundry Security Guide*.

IP Surveillance Cameras

IP surveillance technology provides digital streaming of video over Ethernet, providing real-time, remote access to video feeds from cameras.

The main benefit of using IP surveillance cameras on the network is that you can view surveillance images from any computer on the network. If you have access to the Internet, you can securely connect from anywhere in the world to view a chosen facility or even a single camera from your surveillance system. By using a Virtual Private Network (VPN) or the company intranet, you can manage password-protected access to images from the surveillance system. Similar to secure payment over the Internet, images and information are kept secure and can be viewed only by approved personnel.

Enabling or Disabling Power over Ethernet

To enable a port to receive in-line power for 802.3af-compliant and non-compliant power consuming devices, enter commands such as the following:

```
FastIron SuperX Router# config t
```

```
FastIron SuperX Router(config)# interface e 1/1
FastIron SuperX Router(config-if-e1000-1/1)# inline power
```

After entering the above commands, the console will display the following message:

```
FastIron SuperX Router(config-if-e1000-1/1)# PoE Info: Power enabled on port 1/1.
```

Syntax: [no] inline power

Use the **no** form of the command to disable the port from receiving in-line power.

NOTE: The FSX with POE can automatically detect whether or not a power consuming device is 802.3af-compliant. Therefore, the CLI command **inline power legacy-powerdevice**, which is used on FES POE devices to configure 802.3af non-compliant devices, does not apply on the FSX POE.

Enabling the Detection of POE Power Requirements Advertised via CDP

Many power consuming devices, such as Cisco's VOIP phones and other vendors' devices, use CDP to advertise their power requirements to power sourcing devices, such as Foundry's POE devices. Foundry's power consuming devices are compatible with Cisco's and other vendors' power consuming devices, in that they can detect and process power requirements for these devices automatically.

Configuration Considerations

- This feature is supported in FSX POE devices running software release 02.2.00 or later
- If you configure a port with a maximum power level or a power class for a power consuming device, the power level or power class takes precedence over the CDP power requirement. Therefore, if you want the device to adhere to the CDP power requirement, do not configure a power level or power class on the port.
- The FSX POE will adjust a port's power only if there are available power resources on the device.

Command Syntax

To enable the Foundry device to detect CDP power requirements, enter the following commands:

```
FastIron SuperX Switch# config t
FastIron SuperX Switch(config)# cdp run
```

Syntax: [no] cdp run

Use the **no** form of the command to disable the detection of CDP power requirements.

Setting the Maximum Power Level for a POE Power Consuming Device

When POE is enabled on a port to which a power consuming device is attached, by default, the Foundry POE device will supply 15.4 watts of power at the RJ45 jack, minus any power loss through the cables. For example, a POE port with a default maximum power level of 15.4 watts will receive a maximum of 12.95 watts of power after 2.45 watts of power loss through the cable. This is compliant with the IEEE 802.3af specification for delivering in-line power. Devices that are configured to receive less POE power, for example, 4.0 watts of power, will experience a lower rate of power loss through the cable.

If desired, you can manually configure the maximum amount of power that the Foundry POE device will supply at the RJ45 jack. You can specify from 1 to 15.4 watts of maximum power for each power consuming device connected to the switch.

Configuration Notes

- This feature is supported in FSX POE devices running release 02.2.00 or later

- There are two ways to configure the power level for a POE power consuming device. The first method is discussed in this section. The other method is provided in the section "Setting the Power Class for a POE Power Consuming Device" on page 6-7. For each POE port, you can configure either a maximum power level or a power class. You cannot configure both. You can, however, configure a maximum power level on one port and a power class on another port.
- The CLI commands for this feature differ on the FSX POE compared to the FES POE. On the FES POE, there are separate CLI commands for 802.3af-compliant versus 802.3-af non-compliant power consuming devices. On the FSX, there is one command for all power consuming devices. The command syntax is also different on the FSX. To configure your device, refer to the appropriate section, below.

Command Syntax

To configure the maximum power level for a power consuming device, enter commands such as the following:

```
FastIron SuperX Router# config t
FastIron SuperX Router(config)# interface e 1/1
FastIron SuperX Router(config-if-e1000-1/1)# inline power power-limit 14000
```

These commands enable in-line power on interface e 1 in slot 1 and set the POE power level to 14,000 milliwatts (14 watts).

Syntax: inline power power-limit <power level>

where <power level> is the number of milliwatts, between 1000 and 15400. The default is 15400.

For information about resetting the maximum power level, see "Resetting POE Parameters" on page 6-9.

Setting the Power Class for a POE Power Consuming Device

A power class specifies the maximum amount of power that a Foundry POE device will supply to a power consuming device. Table 6.3 shows the different power classes and their respective maximum power allocations.

Table 6.3: Power Classes for Power Consuming Devices

Class	Maximum Power (Watts)
0	15.4 (default)
1	4
2	7
3	15.4

By default, the power class for all power consuming devices is zero (0). As shown in Table 6.3, a power consuming device with a class of 0 receives 15.4 watts of power.

Configuration Notes

- This feature is supported in the FSX POE devices running release 02.2.00 or later
- The power class sets the maximum power level for a power consuming device. Alternatively, you can set the maximum power level as instructed in the section "Setting the Maximum Power Level for a POE Power Consuming Device" on page 6-6. For each POE port, you can configure either a power class or a maximum power level. You cannot configure both. You can, however, configure a power level on one port and power class on another port.
- The power class includes any power loss through the cables. For example, a POE port with a default power

class of 0 (15.4 watts) will receive a maximum of 12.95 watts of power after 2.45 watts of power loss through the cable. This is compliant with the IEEE 802.3af specification for delivering in-line power. Devices that are configured to receive less POE power, for example, class 1 devices (4.0 watts), will experience a lower rate of power loss through the cable.

- The CLI commands for this feature differ on the FSX POE compared to the FES POE. On the FES POE, there are separate CLI commands for 802.3af-compliant versus 802.3-af non-compliant power consuming devices. On the FSX, there is one command for all power consuming devices. The command syntax is also different on the FSX.

Command Syntax

To configure the power class for a POE power consuming device, enter commands such as the following:

```
FastIron SuperX Switch# config t
FastIron SuperX Switch(config)# interface e 1/1
FastIron SuperX Switch(config-if-e1000-1/1)# inline power power-by-class 2
```

These commands enable in-line power on interface e 1 in slot 1 and set the power class to 2.

Syntax: inline power power-by-class <class value>

where <class value> is the power class. Enter a value from 0 – 3. See Table 6.3 for the power classes and their respective maximum power allocations. The default is 0 (15.4 watts).

For information about resetting the power class, see “Resetting POE Parameters” on page 6-9.

Setting the In-line Power Priority for a POE Port

Each FSX POE (48V) power supply provides a maximum of 1080 watts of power, and each POE port receives a default maximum value of 15.4 watts of power, minus any power loss through the cable. The power capacity of one or two POE power supplies is shared among all POE power consuming devices attached to the FSX POE.

In a configuration where POE power consuming devices collectively have a greater demand for power than the POE power supply or supplies can provide, the FSX must place the POE ports that it cannot power in *standby* or *denied* mode (waiting for power) until the available power increases. The available power increases when one or more POE ports are powered down, or, if applicable, when an additional POE power supply is installed in the FSX.

When POE ports are in *standby* or *denied* mode (waiting for power) and the FSX receives additional power resources, by default, the FSX will allocate newly available power to the standby ports in ascending order, by slot number then by port number, provided enough power is available for the ports. For example, POE port 1/11 should receive power before POE port 2/1. However, if POE port 1/11 needs 12 watts of power and POE port 2/1 needs 10 watts of power, and 11 watts of power become available on the device, the FSX will allocate the power to port 2/1 since it does not have sufficient power for port 1/11.

You can configure an *in-line power priority* on POE ports, whereby ports with a higher in-line power priority will take precedence over ports with a low in-line power priority. For example, if a new POE port comes on-line and the port is configured with a high priority, if necessary (if power is already fully allocated to power consuming devices), the FSX will remove power from a POE port or ports that have a lower priority and allocate the power to the POE port that has the higher value.

Ports that are configured with the same in-line power priority are given precedence based on the slot number and port number in ascending order, provided enough power is available for the port. For example, if both POE port 1/2 and POE port 2/1 have a high in-line power priority value, POE port 1/2 will receive power before POE port 2/1. However, if POE port 1/2 needs 12 watts of power and POE port 2/1 needs 10 watts of power, and 11 watts of power become available on the device, the FSX will allocate the power to POE port 2/1 since it does not have sufficient power for port 1/2. By default, all ports are configured with a low in-line power priority.

Command Syntax

To configure an in-line power priority for a POE port on a FSX, enter commands such as the following:

```
FastIron SuperX Router# config t
FastIron SuperX Router(config)# interface e 1/1
FastIron SuperX Router(config-if-e1000-1/1)# inline power priority 2
```

These commands enable in-line power on interface e 1 in slot 1 and set the in-line power priority level to high.

Syntax: [no] inline power priority <priority num>

where **priority <priority num>** is the in-line power priority number. The default is 3 (low priority). You can specify one of the following values:

- 3 – low priority
- 2 – high priority
- 1 – critical priority

Use the **inline power** command (without a priority number) to reset a port's priority to the default (low) priority.

Use the **no inline power** command to disable the port from receiving in-line power.

For information about resetting the in-line power priority, see "Resetting POE Parameters" on page 6-9.

To view the in-line power priority for all POE ports, issue the **show inline power** command at the Privileged EXEC level of the CLI. See "Displaying POE Operational Status" on page 6-10.

Resetting POE Parameters

NOTE: This feature applies to the FSX POE only.

To override or reset POE port parameters including power priority, power class, and maximum power level, you must specify each POE parameter in the CLI command line. This section provides some examples.

EXAMPLE:

To change a POE port's power priority from high to low (the default value) and keep the current maximum configured power level of 3000, enter commands such as the following:

```
FastIron SuperX Router# config t
FastIron SuperX Router(config)# interface e 1/1
FastIron SuperX Router(config-if-e1000-1/1)# inline power priority 3 power-limit
3000
```

Note that you must specify both the inline power priority and the maximum power level (**power-limit** command), even though you are keeping the current configured maximum power level at 3000. If you do not specify the maximum power level, the device will apply the default value of15400 (15.4 watts). Also, you must specify the inline power priority before specifying the power limit.

EXAMPLE:

To change a port's power class from 2 (4 watts max) to 3 (7 watts max) and keep the current configured power priority of 2, enter commands such as the following:

```
FastIron SuperX Router# config t
FastIron SuperX Router(config)# interface e 1/1
FastIron SuperX Router(config-if-e1000-1/1)# inline power priority 2 power-by-class
3
```

Note that you must specify both the power class and the inline power priority, even though you are not changing the power priority. If you do not specify the power priority, the device will apply the default value of 3 (low priority). Also, you must specify the inline power priority before specifying the power class.

Displaying Power over Ethernet Information

This section lists the CLI commands for viewing POE information.

Displaying POE Operational Status

The **show inline power** command displays operational information about Power over Ethernet.

On the FSX, you can view the POE operational status for the entire device, for a specific POE module only, or for a specific interface only. In addition, on the FSX, you can use the **show inline power detail** command to display in depth information about POE power supplies.

The following shows an example of the **show inline power** display output on a FSX device.

```
FastIron SuperX Switch# show inline power

Power Capacity:         Total is 2160000 mWatts. Current Free is 18800 mWatts.

Power Allocations:      Requests Honored 769 times

 ... some lines omitted for brevity...

 Port   Admin   Oper    ---Power(mWatts)---  PD Type  PD Class  Pri  Fault/
        State   State   Consumed  Allocated                       Error
-------------------------------------------------------------------------
 4/1    On      On          5070       9500  802.3af  n/a         3  n/a
 4/2    On      On          1784       9500  Legacy   n/a         3  n/a
 4/3    On      On          2347       9500  802.3af  n/a         3  n/a
 4/4    On      On          2441       9500  Legacy   n/a         3  n/a
 4/5    On      On          6667       9500  802.3af  Class 3     3  n/a
 4/6    On      On          2723       9500  802.3af  Class 2     3  n/a
 4/7    On      On          2347       9500  802.3af  n/a         3  n/a
 4/8    On      On          2347       9500  802.3af  n/a         3  n/a
 4/9    On      On          2347       9500  802.3af  n/a         3  n/a
 4/10   On      On          4976       9500  802.3af  Class 3     3  n/a
 4/11   On      On          4882       9500  802.3af  Class 3     3  n/a
 4/12   On      On          4413       9500  802.3af  Class 1     3  n/a
 4/13   On      On          7793       9500  802.3af  n/a         3  n/a
 4/14   On      On          7512       9500  802.3af  n/a         3  n/a
 4/15   On      On          8075       9500  802.3af  n/a         3  n/a
 4/16   On      On          4131       9500  802.3af  Class 1     3  n/a
 4/17   On      On          2347       9500  802.3af  n/a         3  n/a
 4/18   On      Off            0       9500  n/a      n/a         3  n/a
 4/19   On      On          5352       9500  Legacy   n/a         3  n/a
 4/20   On      On          7981       9500  802.3af  n/a         3  n/a
 4/21   On      On         12958      13000  802.3af  Class 3     3  n/a
 4/22   On      On         12958      13000  802.3af  Class 3     3  n/a
 4/23   On      On         13052      13000  802.3af  Class 3     3  n/a
 4/24   On      On         12864      13000  802.3af  Class 3     3  n/a
-------------------------------------------------------------------------
 Total                    137367     242000

 ... some lines omitted for brevity...

 Grand Total             1846673    2127400
```

Syntax: show inline power [<slot num>] | [<slot num>/<port num>]

Table 6.4 provides definitions for the statistics.

Table 6.4: Field Definitions for the Show Inline Power Command

This Column...	Displays...
Power Capacity	The total POE power supply capacity and the amount of available power (current free) for POE power consuming devices. Both values are shown in milliwatts.
Power Allocations	The number of times the FSX fulfilled POE requests for power.
Port	The slot number and port number.
Admin State	Specifies whether or not Power over Ethernet has been enabled on the port. This value can be one of the following: • ON – The **inline power** command was issued on the port. • OFF – The **inline power** command has not been issued on the port.
Oper State	Shows the status of in-line power on the port. This value can be one of the following: • ON – The POE power supply is delivering in-line power to the powered device. • OFF – The POE power supply is not delivering in-line power to the powered device. • DENIED – The port is in standby mode (waiting for power) because the FSX does not currently have enough available power for the port.
Power Consumed	The number of current, actual milliwatts that the powered device is consuming.
Power Allocated	The number of milliwatts allocated to the port. This value is either the default or configured maximum power level, or the power class that was automatically detected by the FSX.

PD Type	The type of powered device connected to the port. This value can be one of the following:
	• 802.3AF-PD – The powered device connected to this port is 802.3af-compliant.
	• LEGACY – The powered device connected to this port is a legacy product (not 802.3af-compliant).
	• N/A – Power over Ethernet is configured on this port, and one of the following is true:
	  • The device connected to this port is a non-powered device.
	  • No device is connected to this port.
	  • The port is in *standby* or *denied* mode (waiting for power).

Table 6.4: Field Definitions for the Show Inline Power Command

This Column...	Displays...
PD Class	Determines the maximum amount of power a powered device receives. This value can be one of the following: • Class0 – Receives 15.4 watts maximum. • Class1 – Receives 4 watts maximum • Class2 – Receives 7 watts maximum • Class3 – Receives 15.4 watts maximum • Unknown – The device attached to the port cannot advertise its class.
Pri	The port's *in-line power priority*, which determines the order in which the port will receive power while in standby mode (waiting for power). Ports with a higher priority will receive power before ports with a low priority. This value can be one of the following: • 3 – low priority • 2 – high priority • 1 – critical priority
Total	The total power in milliwatts being *consumed* by all powered devices connected to the Interface module, and the total power in milliwatts *allocated* to all powered devices connected to the Interface module.
Grand Total	The total number of current, actual milliwatts being *consumed* by all powered devices connected to the FSX, and the total number of milliwatts *allocated* to all powered devices connected to the FSX.

Displaying Detailed Information About POE Power Supplies

The **show inline power detail** command displays detailed operational information about the POE power supplies in **FSX** POE devices.

To display detailed POE statistics, enter the following command:

```
FastIron SuperX Switch# show inline power detail

Power Supply Data:
++++++++++++++++++

Power Supply #1:
        Firmware Ver:   0.2
        Date:           3/15/5
        H/W Status:     807
        Max Curr:       26.5 Amps
        Voltage:        50.0 Volts
        Capacity:       1325 Watts
        Consumption:    1144 Watts
Power Supply #2:
        Firmware Ver:   0.2
        Date:           3/15/5
        H/W Status:     807
        Max Curr:       26.5 Amps
        Voltage:        50.0 Volts
        Capacity:       1325 Watts
        Consumption:    949 Watts

General PoE Data:
+++++++++++++++++

Slot  Firmware
      Version
--------------
1     04.0.0
2     04.0.0
3     04.0.0
4     04.0.0
5     04.0.0
6     04.0.0
7     04.0.0
8     04.0.0
```

... continued on next page...

```
... continued from previous page...


Cumulative Port State Data:
+++++++++++++++++++++++++++

Slot  #Ports    #Ports     #Ports   #Ports    #Ports        #Ports     #Ports
      Admin-On  Admin-Off  Oper-On  Oper-Off  Off-Denied    Off-No-PD  Off-Fault
-------------------------------------------------------------------------------
1     24        0          24       0         0             0          0
2     24        0          24       0         0             0          0
3     24        0          23       1         0             1          0
4     24        0          23       1         0             1          1
5     24        0          24       0         0             0          0
6     24        0          24       0         0             0          0
7     24        0          24       0         0             0          0
8     24        0          24       0         0             0          0
-------------------------------------------------------------------------------
Total:192       0          190      2         0             2          1


Cumulative Port Power Data:
+++++++++++++++++++++++++++

Slot  #Ports  #Ports  #Ports        Power        Power
      Pri: 1  Pri: 2  Pri: 3  Consumption  Allocation
-----------------------------------------------------
1     24      0       0         310.146 W   312.0   W
2     0       0       24        308.454 W   312.0   W
3     0       0       24        108.727 W   172.500 W
4     0       0       24        137.366 W   232.500 W
5     24      0       0          56.991 W   145.400 W
6     0       0       24        309.112 W   312.0   W
7     0       0       24        308.548 W   312.0   W
8     24      0       0         307.796 W   312.0   W
-----------------------------------------------------
Total:72      0       120      1847.140 W  2110.400 W
```

Syntax: show inline power detail

Table 6.4 provides definitions for the statistics displayed in the **show inline power detail** command.

Table 6.5: Field Definitions for the Show Inline Power Detail Command

This Column...	Displays...
Power Supply Data	
Firmware ver	The POE power supply's firmware version.
Date	The POE power supply's firmware test date in the format mm/dd/yyyy.
H/W Status	The POE power supply's hardware status code. This field is used by Foundry Technical Support for troubleshooting.
Max Curr	The POE power supply's maximum current capacity.
Voltage	The POE power supply's current input voltage.
Capacity	The POE power supply's total power capacity (in watts).
Consumption	The total number of watts consumed by POE power consuming devices and POE modules in the system, minus any internal or cable power loss.
General POE Data	
Slot	The Interface module / slot number
Firmware Version	The Interface module's / slot number's firmware version.
Cumulative Port State Data	
Slot	The Interface module / slot number
# Ports Admin-On	The number of ports on the Interface module on which the **inline power** command was issued.
# Ports Admin-Off	The number of ports on the Interface module on which the **inline power** command was not issued.
# Ports Oper-On	The number of ports on the Interface module that are receiving in-line power from the POE power supply.
# Ports Oper-Off	The number of ports on the Interface module that are not receiving in-line power from the POE power supply.
# Ports Off-Denied	The number of ports on the Interface module that were denied power because of insufficient power.
# Ports Off-No-PD	The number of ports on the Interface module to which no powered devices are connected.
# Ports Off-Fault	The number of ports on the Interface module that are not receiving power because of a subscription overload.
Total	The totals for all of the fields in the **Cumulative Port State Data** report.
Cumulative Port Power Data	
Slot	The Interface module / slot number
# Ports Pri: 1	The number of POE ports on the Interface module that have a POE port priority of 1.

Table 6.5: Field Definitions for the Show Inline Power Detail Command

This Column...	Displays...
# Ports Pri: 2	The number of POE ports on the Interface module that have a POE port priority of 2.
# Ports Pri: 3	The number of POE ports on the Interface module that have a POE port priority of 3.
Power Consumption	The total number of watts consumed by both POE power consuming devices and the POE module (daughter card) attached to the Interface module.
Power Allocation	The number of watts allocated to the Interface module's POE ports. This value is the sum of the ports' default or configured maximum power levels, or power classes automatically detected by the FSX.
Total	The totals for all of the fields in the **Cumulative Port Power Data** report.

Chapter 7
Configuring Spanning Tree Protocol (STP) and IronSpan Features

This chapter describes how to configure Spanning Tree Protocol (STP) and IronSpan parameters on Foundry Layer 3 Switches using the CLI. IronSpan features extend the operation of standard STP, enabling you to fine tune standard STP and avoid some of its limitations.

This chapter contains the information listed in Table 7.1.

Table 7.1: Chapter Contents

STP Overview

The Spanning Tree Protocol (STP) eliminates Layer 2 loops in networks, by selectively blocking some ports and allowing other ports to forward traffic, based on global (bridge) and local (port) parameters you can configure.

You can enable or disable STP on a global basis (for the entire device), a port-based VLAN basis (for the individual Layer 2 broadcast domain), or an individual port basis.

Configuration procedures are provided for the standard STP bridge and port parameters as well as Foundry IronSpan parameters.

IronSpan is a set of Layer 2 features that enable you to overcome limitations in the standard 802.1d Spanning Tree Protocol (STP). IronSpan includes the features listed in Table 7.1.

Configuring Standard STP Parameters

Foundry Layer 2 Switches and Layer 3 Switches support standard STP as described in the IEEE 802.1D specification. STP is enabled by default on Layer 2 Switches but disabled by default on Layer 3 Switches.

By default, each port-based VLAN on a Foundry device runs a separate spanning tree (a separate instance of STP). A Foundry device has one port-based VLAN (VLAN 1) by default that contains all the device's ports. Thus, by default each Foundry device has one spanning tree. However, if you configure additional port-based VLANs on a Foundry device, then each of those VLANs on which STP is enabled and VLAN 1 all run separate spanning trees.

If you configure a port-based VLAN on the device, the VLAN has the same STP state as the default STP state on the device. Thus, on Layer 2 Switches, new VLANs have STP enabled by default. On Layer 3 Switches, new VLANs have STP disabled by default. You can enable or disable STP in each VLAN separately. In addition, you can enable or disable STP on individual ports.

STP Parameters and Defaults

Table 7.2 lists the default STP states for Foundry devices.

Table 7.2: Default STP States

Device Type	Default STP Type	Default STP State	Default STP State of New VLANs[a]
Layer 2 Switch	MSTP[b]	Enabled	Enabled
Layer 3 Switch	MSTP	Disabled	Disabled

a.When you create a port-based VLAN, the new VLAN's STP state is the same as the default STP state on the device. The new VLAN does not inherit the STP state of the default VLAN.
b.MSTP stands for "Multiple Spanning Tree Protocol". In this type of STP, each port-based VLAN, including the default VLAN, has its own spanning tree. References in this documentation to "STP" apply to MSTP. The Single Spanning Tree Protocol (SSTP) is another type of STP. SSTP includes all VLANs on which STP is enabled in a single spanning tree. See "Single Spanning Tree (SSTP)" on page 7-56.

Table 7.3 lists the default STP bridge parameters. The bridge parameters affect the entire spanning tree. If you are using MSTP, the parameters affect the VLAN. If you are using SSTP, the parameters affect all VLANs that are members of the single spanning tree.

Table 7.3: Default STP Bridge Parameters

Parameter	Description	Default and Valid Values
Forward Delay	The period of time spent by a port in the listening and learning state before moving on to the learning or forwarding state, respectively. The forward delay value is also used for the age time of dynamic entries in the filtering database, when a topology change occurs.	15 seconds Possible values: 4 – 30 seconds
Maximum Age	The interval a bridge will wait for a configuration BPDU from the root bridge before initiating a topology change.	20 seconds Possible values: 6 – 40 seconds
Hello Time	The interval of time between each configuration BPDU sent by the root bridge.	2 seconds Possible values: 1 – 10 seconds
Priority	A parameter used to identify the root bridge in a spanning tree (instance of STP). The bridge with the lowest value has the highest priority and is the root. A higher numerical value means a lower priority; thus, the highest priority is 0.	32768 Possible values: 0 – 65535

NOTE: If you plan to change STP bridge timers, Foundry recommends that you stay within the following ranges, from section 8.10.2 of the IEEE STP specification.

2 * (forward_delay -1) >= max_age

max_age >= 2 * (hello_time +1)

Table 7.4 lists the default STP port parameters. The port parameters affect individual ports and are separately configurable on each port.

Table 7.4: Default STP Port Parameters

Parameter	Description	Default and Valid Values
Priority	The preference that STP gives this port relative to other ports for forwarding traffic out of the spanning tree. A higher numerical value means a lower priority; thus, the highest priority is 8.	128 Possible values: 8 – 252 (configurable in increments of 4)

Table 7.4: Default STP Port Parameters (Continued)

Parameter	Description	Default and Valid Values
Path Cost	The cost of using the port to reach the root bridge. When selecting among multiple links to the root bridge, STP chooses the link with the lowest path cost and blocks the other paths. Each port type has its own default STP path cost.	10 Mbps – 100 100 Mbps – 19 Gigabit – 4 10 Gigabit – 2 Possible values are 0 – 65535

Enabling or Disabling the Spanning Tree Protocol (STP)

STP is *enabled* by default on devices running Layer 2 code. STP is *disabled* by default on devices running Layer 3 code.

You can enable or disable STP on the following levels:

- Globally – Affects all ports and port-based VLANs on the device.
- Port-based VLAN – Affects all ports within the specified port-based VLAN. When you enable or disable STP within a port-based VLAN, the setting overrides the global setting. Thus, you can enable STP for the ports within a port-based VLAN even when STP is globally disabled, or disable the ports within a port-based VLAN when STP is globally enabled.
- Individual port – Affects only the individual port. However, if you change the STP state of the primary port in a trunk group, the change affects all ports in the trunk group.

NOTE: The CLI converts the STP groups into topology groups when you save the configuration. For backward compatibility, you can still use the STP group commands. However, the CLI converts the commands into the topology group syntax. Likewise, the **show stp-group** command displays STP topology groups. See “Topology Groups” on page 1.

Enabling or Disabling STP Globally

Use the following method to enable or disable STP on a device on which you have not configured port-based VLANs.

NOTE: When you configure a VLAN, the VLAN inherits the global STP settings. However, once you begin to define a VLAN, you can no longer configure standard STP parameters globally using the CLI. From that point on, you can configure STP only within individual VLANs.

To enable STP for all ports in all VLANs on a Foundry device, enter the following command:

```
FESX424 Router(config)# spanning-tree
```

This command enables a separate spanning tree in each VLAN, including the default VLAN.

Syntax: [no] spanning-tree

Enabling or Disabling STP in a Port-Based VLAN

Use the following procedure to disable or enable STP on a device on which you have configured a port-based VLAN. Changing the STP state in a VLAN affects only that VLAN.

To enable STP for all ports in a port-based VLAN, enter commands such as the following:

```
FESX424 Router(config)# vlan 10
FESX424 Router(config-vlan-10)# spanning-tree
```

Syntax: [no] spanning-tree

Enabling or Disabling STP on an Individual Port

Use the following procedure to disable or enable STP on an individual port.

NOTE: If you change the STP state of the primary port in a trunk group, the change affects all ports in the trunk group.

To enable STP on an individual port, enter commands such as the following:

```
FastIron SuperX Router(config)# interface 1/1
FastIron SuperX Router(config-if-e1000-1/1)# spanning-tree
```

Syntax: [no] spanning-tree

Changing STP Bridge and Port Parameters

Table 7.3 on page 7-3 and Table 7.4 on page 7-3 list the default STP parameters. If you need to change the default value for an STP parameter, use the following procedures.

Changing STP Bridge Parameters

NOTE: If you plan to change STP bridge timers, Foundry recommends that you stay within the following ranges, from section 8.10.2 of the IEEE STP specification.

2 * (forward_delay -1) >= max_age

max_age >= 2 * (hello_time +1)

To change a Foundry device's STP bridge priority to the highest value to make the device the root bridge, enter the following command:

```
FESX424 Router(config)# spanning-tree priority 0
```

The command in this example changes the priority on a device on which you have not configured port-based VLANs. The change applies to the default VLAN. If you have configured a port-based VLAN on the device, you can configure the parameters only at the configuration level for individual VLANs. Enter commands such as the following:

```
FESX424 Router(config)# vlan 20
FESX424 Router(config-vlan-20)# spanning-tree priority 0
```

To make this change in the default VLAN, enter the following commands:

```
FESX424 Router(config)# vlan 1
FESX424 Router(config-vlan-1)# spanning-tree priority 0
```

Syntax: [no] spanning-tree [forward-delay <value>] | [hello-time <value>] | [maximum-age <value>] | [priority <value>]

The **forward-delay** <value> parameter specifies the forward delay and can be a value from 4 – 30 seconds. The default is 15 seconds.

NOTE: You can configure a Foundry device for faster convergence (including a shorter forward delay) using Fast Span. See "Configuring IronSpan Features" on page 7-16.

The **hello-time** <value> parameter specifies the hello time and can be a value from 1 – 10 seconds. The default is 2 seconds.

NOTE: This parameter applies only when this device or VLAN is the root bridge for its spanning tree.

The **maximum-age** <value> parameter specifies the amount of time the device waits for receipt of a configuration BPDU from the root bridge before initiating a topology change. You can specify from 6 – 40 seconds. The default is 20 seconds.

The **priority** <value> parameter specifies the priority and can be a value from 0 – 65535. A higher numerical value means a lower priority. Thus, the highest priority is 0. The default is 32768.

You can specify some or all of these parameters on the same command line. If you specify more than one parameter, you must specify them in the order shown above, from left to right.

Changing STP Port Parameters

To change the path and priority costs for a port, enter commands such as the following:

```
FESX424 Router(config)# vlan 10
FESX424 Router(config-vlan-10)# spanning-tree ethernet 5 path-cost 15 priority 64
```

Syntax: spanning-tree ethernet [<slotnum>/]<portnum> path-cost <value> | priority <value> | disable | enable

The <portnum> parameter specifies the interface. If you are configuring a chassis device, specify the slot number as well as the port number (<slotnum>/<portnum>).

The **path-cost** <value> parameter specifies the port's cost as a path to the spanning tree's root bridge. STP prefers the path with the lowest cost. You can specify a value from 0 – 65535.

The default depends on the port type:

- 10 Mbps – 100
- 100 Mbps – 19
- Gigabit – 4
- 10 Gigabit – 2

The **priority** <value> parameter specifies the preference that STP gives this port relative to other ports for forwarding traffic out of the spanning tree. You can specify a value from 8 – 252, in increments of 4. If you enter a value that is not divisible by four the software rounds to the nearest value that is. The default is 128. A higher numerical value means a lower priority; thus, the highest priority is 8.

NOTE: If you are upgrading a device that has a configuration saved under an earlier software release, and the configuration contains a value from 0 – 7 for a port's STP priority, the software changes the priority to the default when you save the configuration while running the new release.

The **disable | enable** parameter disables or re-enables STP on the port. The STP state change affects only this VLAN. The port's STP state in other VLANs is not changed.

STP Protection Enhancement

STP protection provides the ability to prohibit an end station from initiating or participating in an STP topology change.

The 802.1W Spanning Tree Protocol (STP) detects and eliminates logical loops in a redundant network by selectively blocking some data paths (ports) and allowing only the best data paths to forward traffic.

In an STP environment, switches, end stations, and other Layer 2 devices use Bridge Protocol Data Units (BPDUs) to exchange information that STP will use to determine the best path for data flow. When a Layer 2 device is powered ON and connected to the network, or when a Layer 2 device goes down, it sends out an STP BPDU, triggering an STP topology change.

In some instances, it is unnecessary for a connected device, such as an end station, to initiate or participate in an STP topology change. In this case, you can enable the STP Protection feature on the Foundry port to which the end station is connected. Foundry's STP Protection feature disables the connected device's ability to initiate or participate in an STP topology change, by dropping all BPDUs received from the connected device.

Configuration Notes

This feature is supported in the following configurations:

- FESX devices running software release 02.1.01 or later
- All FSX and FWSX devices and associated software releases

Enabling STP Protection

You can enable STP Protection on a per-port basis.

To prevent an end station from initiating or participating in STP topology changes, enter the following command at the Interface level of the CLI:

```
FESX424 Switch#(config) interface e 2
FESX424 Switch#(config-if-e1000-2)# stp-protect
```

This command causes the port to drop STP BPDUs sent from the device on the other end of the link.

Syntax: [no] stp-protect

Enter the **no** form of the command to disable STP protection on the port.

Clearing BPDU Drop Counters

For each port that has STP Protection enabled, the Foundry device counts and records the number of dropped BPDUs. You can use CLI commands to clear the BPDU drop counters for all ports on the device, or for a specific port on the device.

To clear the BPDU drop counters for all ports on the device that have STP Protection enabled, enter the following command at the Global CONFIG level of the CLI:

```
FESX424 Switch(config)# clear stp-protect-statistics
```

To clear the BPDU drop counter for a specific port that has STP Protection enabled, enter the following command at the Global CONFIG level of the CLI:

```
FESX424 Switch(config)# clear stp-protect-statistics e 2
```

Syntax: clear stp-protect-statistics [ethernet [<slotnum>/]<port-num>] | [ethernet [<slotnum>/]portnum>]

Viewing the STP Protection Configuration

You can view the STP Protection configuration for all ports on a device, or for a specific port only. The **show stp-protect** command output shows the port number on which STP Protection is enabled, and the number of BPDUs dropped by each port.

To view the STP Protection configuration for all ports on the device, enter the following command at any level of the CLI:

```
FESX424 Switch# show stp-protect
Port ID         BPDU Drop Count
  3                   478
  5                   213
  6                     0
  12                   31
```

To view STP Protection configuration for a specific port, enter the following command at any level of the CLI:

```
FESX424 Switch# show stp-protect e 3
STP-protect is enabled on port 3.  BPDU drop count is 478
```


If you enter the **show stp-protect** command for a port that does not have STP protection enabled, the following message displays on the console:

```
FESX424 Switch# show stp-protect e 4
STP-protect is not enabled on port 4.
```

Syntax: show stp-protect [ethernet [<slotnum>/]<portnum>]

Displaying STP Information

You can display the following STP information:

- All the global and interface STP settings
- CPU utilization statistics
- Detailed STP information for each interface
- STP state information for a port-based VLAN
- STP state information for an individual interface

Displaying STP Information for an Entire Device

To display STP information, enter the following command at any level of the CLI:

```
FastIron SuperX Router# show span

VLAN 1 BPDU cam_index is 3 and the Master DMA Are(HEX)
STP instance owned by VLAN 1

Global STP (IEEE 802.1D) Parameters:

VLAN Root             Root Root Prio Max He- Ho- Fwd Last     Chg  Bridge
 ID   ID              Cost Port rity Age llo ld  dly Chang    cnt  Address
                           Hex  sec sec sec sec sec
   1 800000e0804d4a00 0    Root 8000 20  2   1   15  689      1    00e0804d4a00

Port STP Parameters:

Port  Prio Path  State      Fwd    Design   Designated       Designated
Num   rity Cost             Trans  Cost     Root             Bridge
      Hex
1     80   19    FORWARDING 1      0        800000e0804d4a00 800000e0804d4a00
2     80   0     DISABLED   0      0        0000000000000000 0000000000000000
3     80   0     DISABLED   0      0        0000000000000000 0000000000000000
4     80   0     DISABLED   0      0        0000000000000000 0000000000000000
5     80   19    FORWARDING 1      0        800000e0804d4a00 800000e0804d4a00
6     80   19    BLOCKING   0      0        800000e0804d4a00 800000e0804d4a00
7     80   0     DISABLED   0      0        0000000000000000 0000000000000000
```

<lines for remaining ports excluded for brevity>

Syntax: show span [vlan <vlan-id>] | [pvst-mode] | [<num>] |
[detail [vlan <vlan-id> [ethernet [<slotnum>/]<portnum>] | <num>]]

The **vlan** <vlan-id> parameter displays STP information for the specified port-based VLAN.

The **pvst-mode** parameter displays STP information for the device's Per VLAN Spanning Tree (PVST+) compatibility configuration. See "PVST/PVST+ Compatibility" on page 7-61.

The <num> parameter displays only the entries after the number you specify. For example, on a device with three port-based VLANs, if you enter 1, then information for the second and third VLANs is displayed, but information for the first VLAN is not displayed. Information is displayed according to VLAN number, in ascending order. The entry number is not the same as the VLAN number. For example, if you have port-based VLANs 1, 10, and 2024, then the command output has three STP entries. To display information for VLANs 10 and 2024 only, enter **show span 1**.

The **detail** parameter and its additional optional parameters display detailed information for individual ports. See "Displaying Detailed STP Information for Each Interface" on page 7-12.

The **show span** command shows the following information.

Table 7.5: CLI Display of STP Information

This Field...	Displays...
Global STP Parameters	
VLAN ID	The port-based VLAN that contains this spanning tree (instance of STP). VLAN 1 is the default VLAN. If you have not configured port-based VLANs on this device, all STP information is for VLAN 1.
Root ID	The ID assigned by STP to the root bridge for this spanning tree.
Root Cost	The cumulative cost from this bridge to the root bridge. If this device is the root bridge, then the root cost is 0.
Root Port	The port on this device that connects to the root bridge. If this device is the root bridge, then the value is "Root" instead of a port number.
Priority Hex	This device or VLAN's STP priority. The value is shown in hexadecimal format. **Note**: If you configure this value, specify it in decimal format. See "Changing STP Bridge Parameters" on page 7-5.
Max age sec	The number of seconds this device or VLAN waits for a configuration BPDU from the root bridge before deciding the root has become unavailable and performing a reconvergence.
Hello sec	The interval between each configuration BPDU sent by the root bridge.
Hold sec	The minimum number of seconds that must elapse between transmissions of consecutive Configuration BPDUs on a port.
Fwd dly sec	The number of seconds this device or VLAN waits following a topology change and consequent reconvergence.
Last Chang sec	The number of seconds since the last time a topology change occurred.
Chg cnt	The number of times the topology has changed since this device was reloaded.
Bridge Address	The STP address of this device or VLAN. **Note**: If this address is the same as the Root ID, then this device or VLAN is the root bridge for its spanning tree.

Table 7.5: CLI Display of STP Information (Continued)

This Field...	Displays...
Port STP Parameters	
Port Num	The port number.
Priority Hex	The port's STP priority, in hexadecimal format. **Note**: If you configure this value, specify it in decimal format. See "Changing STP Port Parameters" on page 7-6.
Path Cost	The port's STP path cost.
State	The port's STP state. The state can be one of the following: • BLOCKING – STP has blocked Layer 2 traffic on this port to prevent a loop. The device or VLAN can reach the root bridge using another port, whose state is FORWARDING. When a port is in this state, the port does not transmit or receive user frames, but the port does continue to receive STP BPDUs. • DISABLED – The port is not participating in STP. This can occur when the port is disconnected or STP is disabled on the port. • FORWARDING – STP is allowing the port to send and receive frames. • LISTENING – STP is responding to a topology change and this port is listening for a BPDU from neighboring bridge(s) in order to determine the new topology. No user frames are transmitted or received during this state. • LEARNING – The port has passed through the LISTENING state and will change to the FORWARDING state, depending on the results of STP's reconvergence. The port does not transmit or receive user frames during this state. However, the device can learn the MAC addresses of frames that the port receives during this state and make corresponding entries in the MAC table.
Fwd Trans	The number of times STP has changed the state of this port between BLOCKING and FORWARDING.
Design Cost	The cost to the root bridge as advertised by the designated bridge that is connected to this port. If the designated bridge is the root bridge itself, then the cost is 0. The identity of the designated bridge is shown in the Design Bridge field.
Designated Root	The root bridge as recognized on this port. The value is the same as the root bridge ID listed in the Root ID field.
Designated Bridge	The designated bridge to which this port is connected. The designated bridge is the device that connects the network segment on the port to the root bridge.

Displaying CPU Utilization Statistics

You can display CPU utilization statistics for STP and the IP protocols.

To display CPU utilization statistics for STP for the previous one-second, one-minute, five-minute, and fifteen-minute intervals, enter the following command at any level of the CLI:

```
FastIron SuperX Router# show process cpu
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.01      0.03      0.09      0.22              9
BGP              0.04      0.06      0.08      0.14             13
GVRP             0.00      0.00      0.00      0.00              0
ICMP             0.00      0.00      0.00      0.00              0
IP               0.00      0.00      0.00      0.00              0
OSPF             0.00      0.00      0.00      0.00              0
RIP              0.00      0.00      0.00      0.00              0
STP              0.00      0.03      0.04      0.07              4
VRRP             0.00      0.00      0.00      0.00              0
```

If the software has been running less than 15 minutes (the maximum interval for utilization statistics), the command indicates how long the software has been running. Here is an example:

```
FastIron SuperX Router# show process cpu
The system has only been up for 6 seconds.
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.01      0.00      0.00      0.00              0
BGP              0.00      0.00      0.00      0.00              0
GVRP             0.00      0.00      0.00      0.00              0
ICMP             0.01      0.00      0.00      0.00              1
IP               0.00      0.00      0.00      0.00              0
OSPF             0.00      0.00      0.00      0.00              0
RIP              0.00      0.00      0.00      0.00              0
STP              0.00      0.00      0.00      0.00              0
VRRP             0.00      0.00      0.00      0.00              0
```

To display utilization statistics for a specific number of seconds, enter a command such as the following:

```
FastIron SuperX Router# show process cpu 2
Statistics for last 1 sec and 80 ms
Process Name   Sec(%)   Time(ms)
ARP              0.00          0
BGP              0.00          0
GVRP             0.00          0
ICMP             0.01          1
IP               0.00          0
OSPF             0.00          0
RIP              0.00          0
STP              0.01          0
VRRP             0.00          0
```

When you specify how many seconds' worth of statistics you want to display, the software selects the sample that most closely matches the number of seconds you specified. In this example, statistics are requested for the previous two seconds. The closest sample available is actually for the previous 1 second plus 80 milliseconds.

Syntax: show process cpu [<num>]

The <num> parameter specifies the number of seconds and can be from 1 – 900. If you use this parameter, the command lists the usage statistics only for the specified number of seconds. If you do not use this parameter, the command lists the usage statistics for the previous one-second, one-minute, five-minute, and fifteen-minute intervals.

Displaying the STP State of a Port-Based VLAN

When you display information for a port-based VLAN, that information includes the STP state of the VLAN.

To display information for a port-based VLAN, enter a command such as the following at any level of the CLI. The STP state is shown in bold type in this example.

```
FastIron SuperX Router(config)# show vlans

Total PORT-VLAN entries: 2
Maximum PORT-VLAN entries: 16

legend: [S=Slot]

PORT-VLAN 1, Name DEFAULT-VLAN, Priority level0, Spanning tree On
 Untagged Ports: (S3)  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16
 Untagged Ports: (S3) 17 18 19 20 21 22 23 24
 Untagged Ports: (S4)  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17
 Untagged Ports: (S4) 18 19 20 21 22 23 24
   Tagged Ports: None
   Uplink Ports: None

PORT-VLAN 2, Name greenwell, Priority level0, Spanning tree Off
 Untagged Ports: (S1)  1  2  3  4  5  6  7  8
 Untagged Ports: (S4)  1
   Tagged Ports: None
   Uplink Ports: None
```

Syntax: show vlans [<vlan-id> | ethernet [<slotnum>/]<portnum>

The <vlan-id> parameter specifies a VLAN for which you want to display the configuration information.

The <portnum> parameter specifies a port. If you use this parameter, the command lists all the VLAN memberships for the port. If you use this command on a chassis device, specify the slot number as well as the port number (<slotnum>/]<portnum>).

Displaying Detailed STP Information for Each Interface

To display the detailed STP information, enter the following command at any level of the CLI:

```
FastIron SuperX Router# show span detail
========================================================================
VLAN 1 - MULTIPLE SPANNING TREE (MSTP) ACTIVE
========================================================================
Bridge identifier    - 0x800000e0804d4a00
Active global timers - Hello: 0

Port 1/1 is FORWARDING
  Port - Path cost: 19, Priority: 128, Root: 0x800000e052a9bb00
  Designated - Bridge: 0x800000e052a9bb00, Interface: 1, Path cost: 0
  Active Timers - None
  BPDUs - Sent: 11, Received: 0
Port 1/2 is DISABLED
Port 1/3 is DISABLED
Port 1/4 is DISABLED
<lines for remaining ports excluded for brevity>
```

If a port is disabled, the only information shown by this command is “DISABLED”. If a port is enabled, this display shows the following information.

Syntax: show span detail [vlan <vlan-id> [ethernet [<slotnum>/]<portnum> | <num>]

The **vlan** <vlan-id> parameter specifies a VLAN.

The <portnum> parameter specifies an individual port within the VLAN (if specified). If you use the command on a chassis device, specify the slot number as well as the port number (<slotnum>/<portnum>).

The <num> parameter specifies the number of VLANs you want the CLI to skip before displaying detailed STP information. For example, if the device has six VLANs configured (VLAN IDs 1, 2, 3, 99, 128, and 256) and you enter the command **show span detail 4**, detailed STP information is displayed for VLANs 128 and 256 only.

NOTE: If the configuration includes VLAN groups, the **show span detail** command displays the master VLANs of each group but not the member VLANs within the groups. However, the command does indicate that the VLAN is a master VLAN. The **show span detail vlan** <vlan-id> command displays the information for the VLAN even if it is a member VLAN. To list all the member VLANs within a VLAN group, enter the **show vlan-group** [<group-id>] command.

The **show span detail** command shows the following information.

Table 7.6: CLI Display of Detailed STP Information for Ports

This Field...	Displays...
Active Spanning Tree protocol	The VLAN that contains the listed ports and the active Spanning Tree protocol. The STP type can be one of the following: • MULTIPLE SPANNNG TREE (MSTP) • GLOBAL SINGLE SPANNING TREE (SSTP) **Note**: If STP is disabled on a VLAN, the command displays the following message instead: "Spanning-tree of port-vlan <vlan-id> is disabled."
Bridge identifier	The STP identity of this device.
Active global timers	The global STP timers that are currently active, and their current values. The following timers can be listed: • Hello – The interval between Hello packets. This timer applies only to the root bridge. • Topology Change (TC) – The amount of time during which the topology change flag in Hello packets will be marked, indicating a topology change. This timer applies only to the root bridge. • Topology Change Notification (TCN) – The interval between Topology Change Notification packets sent by a non-root bridge toward the root bridge. This timer applies only to non-root bridges.

Table 7.6: CLI Display of Detailed STP Information for Ports (Continued)

This Field...	Displays...
Port number and STP state	The internal port number and the port's STP state. The internal port number is one of the following: • The port's interface number, if the port is the designated port for the LAN. • The interface number of the designated port from the received BPDU, if the interface is not the designated port for the LAN. The state can be one of the following: • BLOCKING – STP has blocked Layer 2 traffic on this port to prevent a loop. The device or VLAN can reach the root bridge using another port, whose state is FORWARDING. When a port is in this state, the port does not transmit or receive user frames, but the port does continue to receive STP BPDUs. • DISABLED – The port is not participating in STP. This can occur when the port is disconnected or STP is administratively disabled on the port. • FORWARDING – STP is allowing the port to send and receive frames. • LISTENING – STP is responding to a topology change and this port is listening for a BPDU from neighboring bridge(s) in order to determine the new topology. No user frames are transmitted or received during this state. • LEARNING – The port has passed through the LISTENING state and will change to the BLOCKING or FORWARDING state, depending on the results of STP's reconvergence. The port does not transmit or receive user frames during this state. However, the device can learn the MAC addresses of frames that the port receives during this state and make corresponding entries in the MAC table. **Note**: If the state is DISABLED, no further STP information is displayed for the port.
Port Path cost	The port's STP path cost.
Port Priority	This port's STP priority. The value is shown as a hexadecimal number.
Root	The ID assigned by STP to the root bridge for this spanning tree.
Designated Bridge	The MAC address of the designated bridge to which this port is connected. The designated bridge is the device that connects the network segment on the port to the root bridge.
Designated Port	The port number sent from the designated bridge.
Designated Path Cost	The cost to the root bridge as advertised by the designated bridge that is connected to this port. If the bridge is the root bridge itself, then the cost is 0. The identity of the designated bridge is shown in the Designated Bridge field.

Table 7.6: CLI Display of Detailed STP Information for Ports (Continued)

This Field...	Displays...
Active Timers	The current values for the following timers, if active: • Message age – The number of seconds this port has been waiting for a hello message from the root bridge. • Forward delay – The number of seconds that have passed since the last topology change and consequent reconvergence. • Hold time – The number of seconds that have elapsed since transmission of the last Configuration BPDU.
BPDUs Sent and Received	The number of BPDUs sent and received on this port since the software was reloaded.

Displaying Detailed STP Information for a Single Port in a Specific VLAN

Enter a command such as the following to display STP information for an individual port in a specific VLAN.

```
FastIron SuperX Router(config)# show span detail vlan 1 ethernet 7/1
Port 7/1 is FORWARDING
  Port - Path cost: 19, Priority: 128, Root: 0x800000e052a9bb00
  Designated - Bridge: 0x800000e052a9bb00, Interface: 7, Path cost: 0
  Active Timers - None
  BPDUs - Sent: 29, Received: 0
```

Syntax: show span detail [vlan <vlan-id> [ethernet [<slotnum>/]<portnum> | <num>]

Displaying STP State Information for an Individual Interface

To display STP state information for an individual port, you can use the methods in "Displaying STP Information for an Entire Device" on page 7-8 or "Displaying Detailed STP Information for Each Interface". You also can display STP state information for a specific port using the following method.

To display information for a specific port, enter a command such as the following at any level of the CLI:

```
FastIron SuperX Router(config)# show interface ethernet 3/11

FastEthernet3/11 is up, line protocol is up
  Hardware is FastEthernet, address is 00e0.52a9.bb49 (bia 00e0.52a9.bb49)
  Configured speed auto, actual 100Mbit, configured duplex fdx, actual fdx
  Member of L2 VLAN ID 1, port is untagged, port state is FORWARDING
  STP configured to ON, priority is level0, flow control enabled
  mirror disabled, monitor disabled
  Not member of any active trunks
  Not member of any configured trunks
  No port name
  MTU 1518 bytes, encapsulation ethernet
  5 minute input rate: 352 bits/sec, 0 packets/sec, 0.00% utilization
  5 minute output rate: 0 bits/sec, 0 packets/sec, 0.00% utilization
  1238 packets input, 79232 bytes, 0 no buffer
  Received 686 broadcasts, 0 runts, 0 giants
  0 input errors, 0 CRC, 0 frame, 0 ignored
  529 multicast
  918 packets output, 63766 bytes, 0 underruns
  0 output errors, 0 collisions
```


The STP information is shown in bold type in this example.

Syntax: show interfaces [ethernet [<slotnum>/]<portnum>] | [loopback <num>] | [slot <slot-num>] | [ve <num>] | [brief]

You also can display the STP states of all ports by entering a command such as the following, which uses the **brief** parameter:

```
FastIron SuperX Router(config)# show interface brief

Port   Link State      Dupl Speed Trunk Tag Priori MAC             Name
1/1    Down None       None None  None  No  level0 00e0.52a9.bb00
1/2    Down None       None None  None  No  level0 00e0.52a9.bb01
1/3    Down None       None None  None  No  level0 00e0.52a9.bb02
1/4    Down None       None None  None  No  level0 00e0.52a9.bb03
1/5    Down None       None None  None  No  level0 00e0.52a9.bb04
1/6    Down None       None None  None  No  level0 00e0.52a9.bb05
1/7    Down None       None None  None  No  level0 00e0.52a9.bb06
1/8    Down None       None None  None  No  level0 00e0.52a9.bb07


.
.  some rows omitted for brevity
.
3/10   Down None       None None  None  No  level0 00e0.52a9.bb4a
3/11   Up   Forward    Full 100M  None  No  level0 00e0.52a9.bb49
```

In the example above, only one port, 3/11, is forwarding traffic toward the root bridge.

Configuring IronSpan Features

IronSpan features extend the operation of standard STP, enabling you to fine tune standard STP and avoid some of its limitations.

This section describes how to configure IronSpan parameters on Foundry Layer 3 Switches using the CLI.

Fast Port Span

When STP is running on a device, message forwarding is delayed during the spanning tree recalculation period following a topology change. The STP forward delay parameter specifies the period of time a bridge waits before forwarding data packets. The forward delay controls the listening and learning periods of STP reconvergence. You can configure the forward delay to a value from 4 – 30 seconds. The default is 15 seconds. Thus, using the standard forward delay, convergence requires 30 seconds (15 seconds for listening and an additional 15 seconds for learning) when the default value is used.

This slow convergence is undesirable and unnecessary in some circumstances. The Fast Port Span feature allows certain ports to enter the forwarding state in four seconds. Specifically, Fast Port Span allows faster convergence on ports that are attached to end stations and thus do not present the potential to cause Layer 2 forwarding loops. Because the end stations cannot cause forwarding loops, they can safely go through the STP state changes (blocking to listening to learning to forwarding) more quickly than is allowed by the standard STP convergence time. Fast Port Span performs the convergence on these ports in four seconds (two seconds for listening and two seconds for learning).

In addition, Fast Port Span enhances overall network performance in the following ways:

- Fast Port Span reduces the number of STP topology change notifications on the network. When an end station attached to a Fast Span port comes up or down, the Foundry device does not generate a topology change notification for the port. In this situation, the notification is unnecessary since a change in the state of the host does not affect the network's topology.

- Fast Port Span eliminates unnecessary MAC cache aging that can be caused by topology change notifications. Bridging devices age out the learned MAC addresses in their MAC caches if the addresses are unrefreshed for a given period of time, sometimes called the MAC aging interval. When STP sends a topology change notification, devices that receive the notification use the value of the STP forward delay to quickly age out their MAC caches. For example, if a device's normal MAC aging interval is 5 minutes, the aging interval changes temporarily to the value of the forward delay (for example, 15 seconds) in response to an STP topology change.

  In normal STP, the accelerated cache aging occurs even when a single host goes up or down. Because Fast Port Span does not send a topology change notification when a host on a Fast Port Span port goes up or down, the unnecessary cache aging that can occur in these circumstances under normal STP is eliminated.

Fast Port Span is a system-wide parameter and is enabled by default. Thus, when you boot a device, all the ports that are attached only to end stations run Fast Port Span. For ports that are not eligible for Fast Port Span, such as ports connected to other networking devices, the device automatically uses the normal STP settings. If a port matches any of the following criteria, the port is ineligible for Fast Port Span and uses normal STP instead:

- The port is 802.1Q tagged
- The port is a member of a trunk group
- The port has learned more than one active MAC address
- An STP Configuration BPDU has been received on the port, thus indicating the presence of another bridge on the port.

You also can explicitly exclude individual ports from Fast Port Span if needed. For example, if the only uplink ports for a wiring closet switch are Gigabit ports, you can exclude the ports from Fast Port Span.

Disabling and Re-enabling Fast Port Span

Fast Port Span is a system-wide parameter and is enabled by default. Thus all ports that are eligible for Fast Port Span use it.

To disable or re-enable Fast Port Span, enter the following commands:

```
FESX424 Router(config)# no fast port-span
FESX424 Router(config)# write memory
```

Syntax: [no] fast port-span

NOTE: The **fast port-span** command has additional parameters that let you exclude specific ports. These parameters are shown in the following section.

To re-enable Fast Port Span, enter the following commands:

```
FESX424 Router(config)# fast port-span
FESX424 Router(config)# write memory
```

Excluding Specific Ports from Fast Port Span

To exclude a port from Fast Port Span while leaving Fast Port Span enabled globally, enter commands such as the following:

```
FESX424 Router(config)# fast port-span exclude ethernet 1
FESX424 Router(config)# write memory
```

To exclude a set of ports from Fast Port Span, enter commands such as the following:

```
FESX424 Router(config)# fast port-span exclude ethernet 1 ethernet 2 ethernet 3
FESX424 Router(config)# write memory
```

To exclude a contiguous (unbroken) range of ports from Fast Span, enter commands such as the following:

```
FESX424 Router(config)# fast port-span exclude ethernet 1 to 24
FESX424 Router(config)# write memory
```

Syntax: [no] fast port-span [exclude ethernet [<slotnum>/]<portnum> [ethernet [<slotnum>/]<portnum> | to [<slotnum>/]<portnum>]]

To re-enable Fast Port Span on a port, enter a command such as the following:

```
FESX424 Router(config)# no fast port-span exclude ethernet 1
FESX424 Router(config)# write memory
```

This command re-enables Fast Port Span on port 1 only and does not re-enable Fast Port Span on other excluded ports. You also can re-enable Fast Port Span on a list or range of ports using the syntax shown above this example.

To re-enable Fast Port Span on all excluded ports, disable and then re-enable Fast Port Span by entering the following commands:

```
FESX424 Router(config)# no fast port-span
FESX424 Router(config)# fast port-span
FESX424 Router(config)# write memory
```

Disabling and then re-enabling Fast Port Span clears the exclude settings and thus enables Fast Port Span on all eligible ports. To make sure Fast Port Span remains enabled on the ports following a system reset, save the configuration changes to the startup-config file after you re-enable Fast Port Span. Otherwise, when the system resets, those ports will again be excluded from Fast Port Span.

802.1W Rapid Spanning Tree (RSTP)

Foundry's earlier implementation of Rapid Spanning Tree Protocol (RSTP), which was 802.1W Draft 3, provided only a subset of the IEEE 802.1W standard; whereas the 802.1W RSTP feature provides the full standard. The implementation of the 802.1W Draft 3 is referred to as RSTP Draft 3.

RSTP Draft3 will continue to be supported on Foundry devices for backward compatibility. However, customers who are currently using RSTP Draft 3 should migrate to 802.1W.

The 802.1W feature provides rapid traffic reconvergence for point-to-point links within a few milliseconds (0 – 500 milliseconds), following the failure of a bridge or bridge port. This reconvergence occurs more rapidly than the reconvergence provided by the 802.1D (Spanning Tree Protocol (STP)) or by RSTP Draft 3.

NOTE: This rapid convergence will not occur on ports connected to shared media devices, such as hubs. To take advantage of the rapid convergence provided by 802.1W, make sure to explicitly configure all point-to-point links in a topology.

The convergence provided by the standard 802.1W protocol occurs more rapidly than the convergence provided by previous spanning tree protocols because:

- Classic or legacy 802.1D STP protocol requires a newly selected Root port to go through listening and learning stages before traffic convergence can be achieved. The 802.1D traffic convergence time is calculated using the following formula:

  2 x FORWARD_DELAY + BRIDGE_MAX_AGE.

  If default values are used in the parameter configuration, convergence can take up to 50 seconds. (In this document STP will be referred to as 802.1D.)

- RSTP Draft 3 works only on bridges that have Alternate ports, which are the precalculated "next best root port". (Alternate ports provide back up paths to the root bridge.) Although convergence occurs from 0 – 500 milliseconds in RSTP Draft 3, the spanning tree topology reverts to the 802.1D convergence if an Alternate port is not found.

- Convergence in 802.1w bridge is not based on any timer values. Rather, it is based on the explicit handshakes between Designated ports and their connected Root ports to achieve convergence in less than 500 milliseconds.

Bridges and Bridge Port Roles

A bridge in an 802.1W rapid spanning tree topology is assigned as the root bridge if it has the highest priority (lowest bridge identifier) in the topology. Other bridges are referred to as non-root bridges.

Unique roles are assigned to ports on the root and non-root bridges. Role assignments are based on the following information contained in the Rapid Spanning Tree Bridge Packet Data Unit (RST BPDU):

- Root bridge ID
- Path cost value
- Transmitting bridge ID
- Designated port ID

The 802.1W algorithm uses this information to determine if the RST BPDU received by a port is superior to the RST BPDU that the port transmits. The two values are compared in the order as given above, starting with the Root bridge ID. The RST BPDU with a lower value is considered superior. The superiority and inferiority of the RST BPDU is used to assign a role to a port.

If the value of the received RST BPDU is the same as that of the transmitted RST BPDU, then the port ID in the RST BPDUs are compared. The RST BPDU with the lower port ID is superior. Port roles are then calculated appropriately.

The port's role is included in the BPDU that it transmits. The BPDU transmitted by an 802.1W port is referred to as an RST BPDU, while it is operating in 802.1W mode.

Ports can have one of the following roles:

- Root – Provides the lowest cost path to the root bridge from a specific bridge
- Designated – Provides the lowest cost path to the root bridge from a LAN to which it is connected
- Alternate – Provides an alternate path to the root bridge when the root port goes down
- Backup – Provides a backup to the LAN when the Designated port goes down
- Disabled – Has no role in the topology

Assignment of Port Roles

At system start-up, all 802.1W-enabled bridge ports assume a Designated role. Once start-up is complete, 802.1W algorithm calculates the superiority or inferiority of the RST BPDU that is received and transmitted on a port.

On a root bridge, each port is assigned a ***Designated port*** role, except for ports on the same bridge that are physically connected together. In these type of ports, the port that receives the superior RST BPDU becomes the ***Backup port***, while the other port becomes the ***Designated port***.

On non-root bridges, ports are assigned as follows:

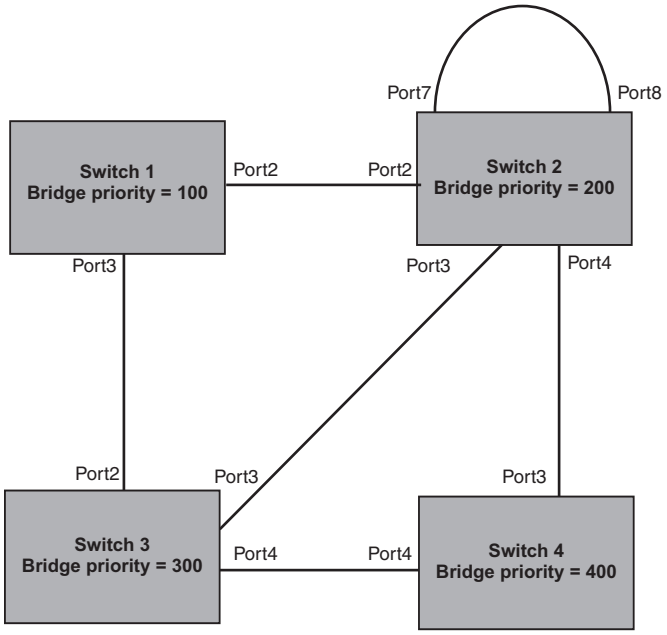
- The port that receives the RST BPDU with the lowest path cost from the root bridge becomes the ***Root port***.
- If two ports on the same bridge are physically connected, the port that receives the superior RST BPDU becomes the ***Backup port***, while the other port becomes the ***Designated port***.
- If a non-root bridge already has a Root port, then the port that receives an RST BPDU that is superior to those it can transmit becomes the ***Alternate port***.
- If the RST BPDU that a port receives is inferior to the RST BPDUs it transmits, then the port becomes a ***Designated port***.
- If the port is down or if 802.1W is disabled on the port, that port is given the role of ***Disabled port***. Disabled ports have no role in the topology. However, if 802.1W is enabled on a port with a link down and the link of that port comes up, then that port assumes one of the following port roles: Root, Designated, Alternate, or Backup.

The following example (Figure 7.1) explains role assignments in a simple RSTP topology.

NOTE: All examples in this document assume that all ports in the illustrated topologies are point-to-point links and are homogeneous (they have the same path cost value) unless otherwise specified.

The topology in Figure 7.1 contains four bridges. Switch 1 is the root bridge since it has the lowest bridge priority. Switch 2 through Switch 4 are non-root bridges.

Figure 7.1 Simple 802.1W Topology



Ports on Switch 1

All ports on Switch 1, the root bridge, are assigned Designated port roles.

Ports on Switch 2

Port2 on Switch 2 directly connects to the root bridge; therefore, Port2 is the Root port.

Switch 2's bridge priority value is superior to that of Switch 3 and Switch 4; therefore, the ports on Switch 2 that connect to Switch 3 and Switch 4 are given the Designated port role.

Furthermore, Port7 and Port8 on Switch 2 are physically connected. The RST BPDUs transmitted by Port7 are superior to those Port8 transmits. Therefore, Port8 is the Backup port and Port7 is the Designated port.

Ports on Switch 3

Port2 on Switch 3 directly connects to the Designated port on the root bridge; therefore, it assumes the Root port role.

The root path cost of the RST BPDUs received on Port4/Switch 3 is inferior to the RST BPDUs transmitted by the port; therefore, Port4/Switch 3 becomes the Designated port.

Similarly Switch 3 has a bridge priority value inferior to Switch 2. Port3 on Switch 3 connects to Port 3 on Switch 2. This port will be given the Alternate port role, since a Root port is already established on this bridge.

Ports Switch 4

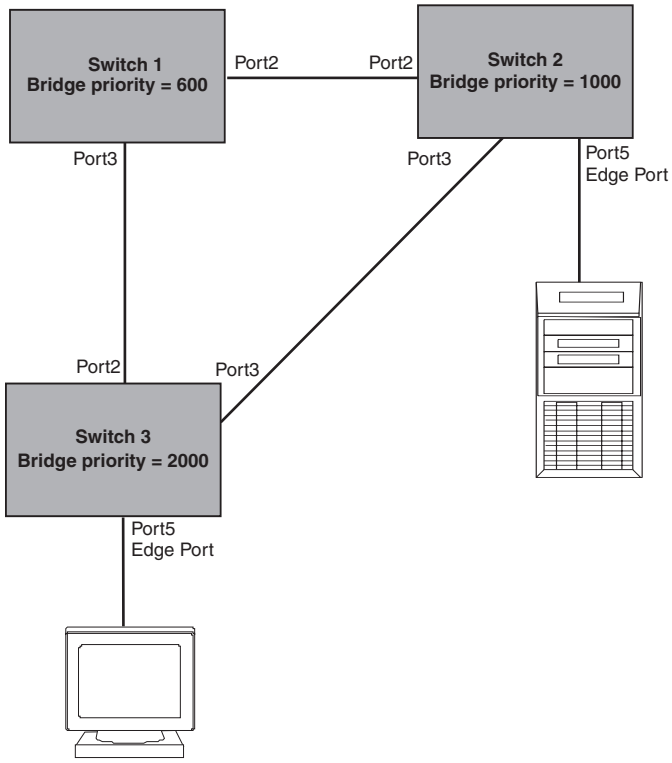
Switch 4 is not directly connected to the root bridge. It has two ports with superior incoming RST BPDUs from two separate LANs: Port3 and Port4. The RST BPDUs received on Port3 are superior to the RST BPDUs received on port 4; therefore, Port3 becomes the Root port and Port4 becomes the Alternate port.

Edge Ports and Edge Port Roles

Foundry's implementation of 802.1W allows ports that are configured as Edge ports to be present in an 802.1W topology. (Figure 7.2). Edge ports are ports of a bridge that connect to workstations or computers. Edge ports do not register any incoming BPDU activities.

Edge ports assume Designated port roles. Port flapping does not cause any topology change events on Edge ports since 802.1W does not consider Edge ports in the spanning tree calculations.

Figure 7.2 Topology with Edge Ports



However, if any incoming RST BPDU is received from a previously configured Edge port, 802.1W automatically makes the port as a non-edge port. This is extremely important to ensure a loop free Layer 2 operation since a non-edge port is part of the active RSTP topology.

The 802.1W protocol can auto-detect an Edge port and a non-edge port. An administrator can also configure a port to be an Edge port using the CLI. It is recommended that Edge ports are configured explicitly to take advantage of the Edge port feature, instead of allowing the protocol to auto-detect them.

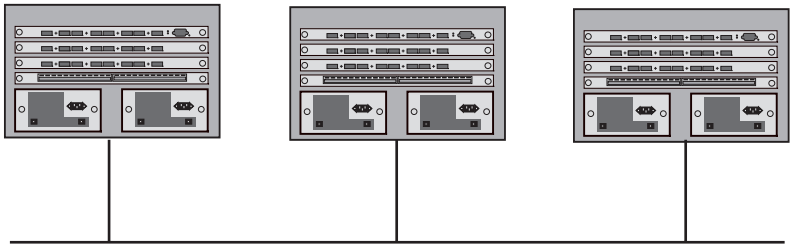
Point-to-Point Ports

To take advantage of the 802.1W features, ports on an 802.1W topology should be explicitly configured as point-to-point links using the CLI. Shared media should not be configured as point-to-point links.

NOTE: Configuring shared media or non-point-to-point links as point-to-point links could lead to Layer 2 loops.

The topology in Figure 7.3 is an example of shared media that should not be configured as point-to-point links. In Figure 7.3, a port on a bridge communicates or is connected to at least two ports.

Figure 7.3 Example of Shared Media



Bridge Port States

Ports roles can have one of the following states:

- Forwarding – 802.1W is allowing the port to send and receive all packets.
- Discarding – 802.1W has blocked data traffic on this port to prevent a loop. The device or VLAN can reach the root bridge using another port, whose state is forwarding. When a port is in this state, the port does not transmit or receive data frames, but the port does continue to receive RST BPDUs. This state corresponds to the listening and blocking states of 802.1D.
- Learning – 802.1W is allowing MAC entries to be added to the filtering database but does not permit forwarding of data frames. The device can learn the MAC addresses of frames that the port receives during this state and make corresponding entries in the MAC table.
- Disabled – The port is not participating in 802.1W. This can occur when the port is disconnected or 802.1W is administratively disabled on the port.

A port on a non-root bridge with the role of Root port is always in a forwarding state. If another port on that bridge assumes the Root port role, then the old Root port moves into a discarding state as it assumes another port role.

A port on a non-root bridge with a Designated role starts in the discarding state. When that port becomes elected to the Root port role, 802.1W quickly places it into a forwarding state. However, if the Designated port is an Edge port, then the port starts and stays in a forwarding state and it cannot be elected as a Root port.

A port with an Alternate or Backup role is always in a discarding state. If the port's role changes to Designated, then the port changes into a forwarding state.

If a port on one bridge has a Designated role and that port is connected to a port on another bridge that has an Alternate or Backup role, the port with a Designated role cannot be given a Root port role until two instances of the forward delay timer expires on that port.

Edge Port and Non-Edge Port States

As soon as a port is configured as an Edge port using the CLI, it goes into a forwarding state instantly (in less than 100 msec):

When the link to a port comes up and 802.1W detects that the port is an Edge port, that port instantly goes into a forwarding state.

If 802.1W detects that port as a non-edge port, the port state is changed as determined by the result of processing the received RST BPDU. The port state change occurs within four seconds of link up or after two hello timer expires on the port.

Changes to Port Roles and States

To achieve convergence in a topology, a port's role and state changes as it receives and transmits new RST BPDUs. Changes in a port's role and state constitute a topology change. Besides the superiority and inferiority of the RST BPDU, bridge-wide and per-port state machines are used to determine a port's role as well as a port's state. Port state machines also determine when port role and state changes occur.

State Machines

The bridge uses the Port Role Selection state machine to determine if port role changes are required on the bridge. This state machine performs a computation when one of the following events occur:

- New information is received on any port on the bridge
- The timer expires for the current information on a port on the bridge

Each port uses the following state machines:

- Port Information – This state machine keeps track of spanning-tree information currently used by the port. It records the origin of the information and ages out any information that was derived from an incoming BPDU.
- Port Role Transition – This state machine keeps track of the current port role and transitions the port to the appropriate role when required. It moves the Root port and the Designated port into forwarding states and moves the Alternate and Backup ports into discarding states.
- Port Transmit – This state machine is responsible for BPDU transmission. It checks to ensure only the maximum number of BPDUs per hello interval are sent every second. Based on what mode it is operating in, it sends out either legacy BPDUs or RST BPDUs. In this document legacy BPDUs are also referred to as STP BPDUs.
- Port Protocol Migration – This state machine deals with compatibility with 802.1D bridges. When a legacy BPDU is detected on a port, this state machine configures the port to transmit and receive legacy BPDUs and operate in the legacy mode.
- Topology Change – This state machine detects, generates, and propagates topology change notifications. It acknowledges Topology Change Notice (TCN) messages when operating in 802.1D mode. It also flushes the MAC table when a topology change event takes place.
- Port State Transition – This state machine transitions the port to a discarding, learning, or forwarding state and performs any necessary processing associated with the state changes.
- Port Timers – This state machine is responsible for triggering any of the state machines described above, based on expiration of specific port timers.

In contrast to the 802.1D standard, the 802.1W standard does not have any bridge specific timers. All timers in the CLI are applied on a per-port basis, even though they are configured under bridge parameters.

802.1W state machines attempt to quickly place the ports into either a forwarding or discarding state. Root ports are quickly placed in forwarding state when both of the following events occur:

- It is assigned to be the Root port.
- It receives an RST BPDU with a proposal flag from a Designated port. The proposal flag is sent by ports with a Designated role when they are ready to move into a forwarding state.

When a the role of Root port is given to another port, the old Root port is instructed to reroot. The old Root port goes into a discarding state and negotiates with its peer port for a new role and a new state. A peer port is the port on the other bridge to which the port is connected. For example, in Figure 7.4, Port1 of Switch 200 is the peer port of Port2 of Switch 100.

A port with a Designated role is quickly placed into a forwarding state if one of the following occurs:

- The Designated port receives an RST BPDU that contains an agreement flag from a Root port
- The Designated port is an Edge port

However, a Designated port that is attached to an Alternate port or a Backup port must wait until the forward delay timer expires twice on that port while it is still in a Designated role, before it can proceed to the forwarding state.

Backup ports are quickly placed into discarding states.

Alternate ports are quickly placed into discarding states.

A port operating in 802.1W mode may enter a learning state to allow MAC entries to be added to the filtering database; however, this state is transient and lasts only a few milliseconds, if the port is operating in 802.1W mode and if the port meets the conditions for rapid transition.

Handshake Mechanisms

To rapidly transition a Designated or Root port into a forwarding state, the Port Role Transition state machine uses handshake mechanisms to ensure loop free operations. It uses one type of handshake if no Root port has been assigned on a bridge, and another type if a Root port has already been assigned.

Handshake When No Root Port is Elected

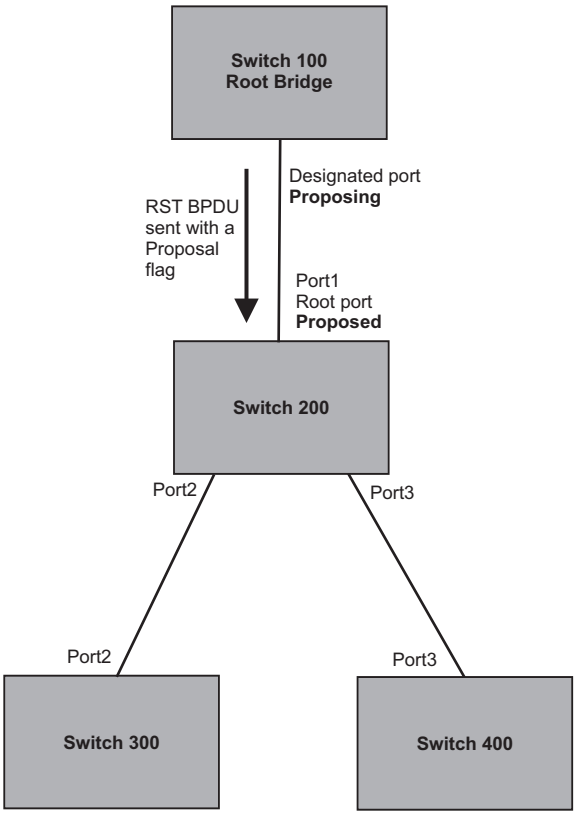
If a Root port has not been assigned on a bridge, 802.1W uses the Proposing -> Proposed -> Sync -> Synced -> Agreed handshake:

- Proposing – The Designated port on the root bridge sends an RST BPDU packet to its peer port that contains a proposal flag. The proposal flag is a signal that indicates that the Designated port is ready to put itself in a forwarding state (Figure 7.4). The Designated port continues to send this flag in its RST BPDU until it is placed in a forwarding state (Figure 7.7) or is forced to operate in 802.1D mode. (See "Compatibility of 802.1W with 802.1D" on page 43.)
- Proposed – When a port receives an RST BPDU with a proposal flag from the Designated port on its point-to-point link, it asserts the Proposed signal and one of the following occurs (Figure 7.4):
  - If the RST BPDU that the port receives is superior to what it can transmit, the port assumes the role of a Root port. (See the section on "Bridges and Bridge Port Roles" on page 7-19.)
  - If the RST BPDU that the port receives is inferior to what it can transmit, then the port is given the role of Designated port.

NOTE: Proposed will never be asserted if the port is connected on a shared media link.

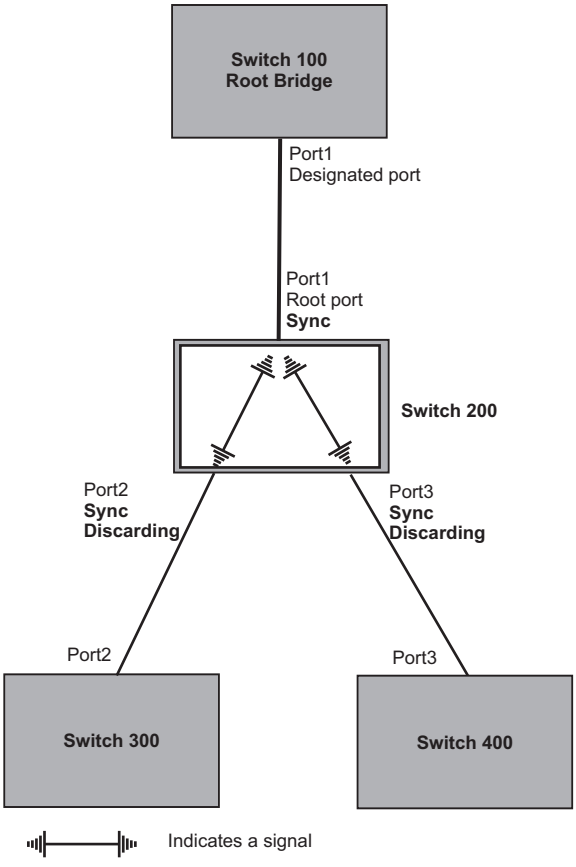
In Figure 7.4, Port3/Switch 200 is elected as the Root port

Figure 7.4 Proposing and Proposed Stage



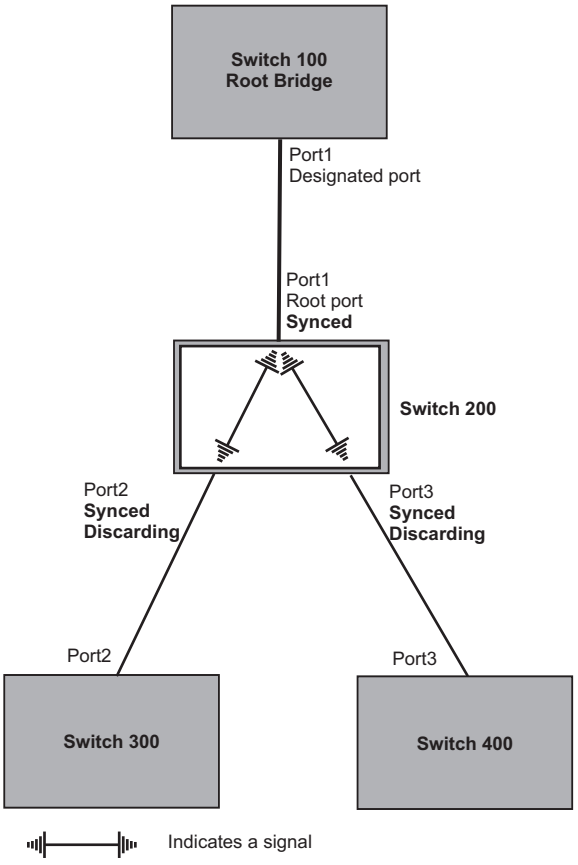
- Sync – Once the Root port is elected, it sets a sync signal on all the ports on the bridge. The signal tells the ports to synchronize their roles and states (Figure 7.5). Ports that are non-edge ports with a role of Designated port change into a discarding state. These ports have to negotiate with their peer ports to establish their new roles and states.

Figure 7.5 Sync Stage



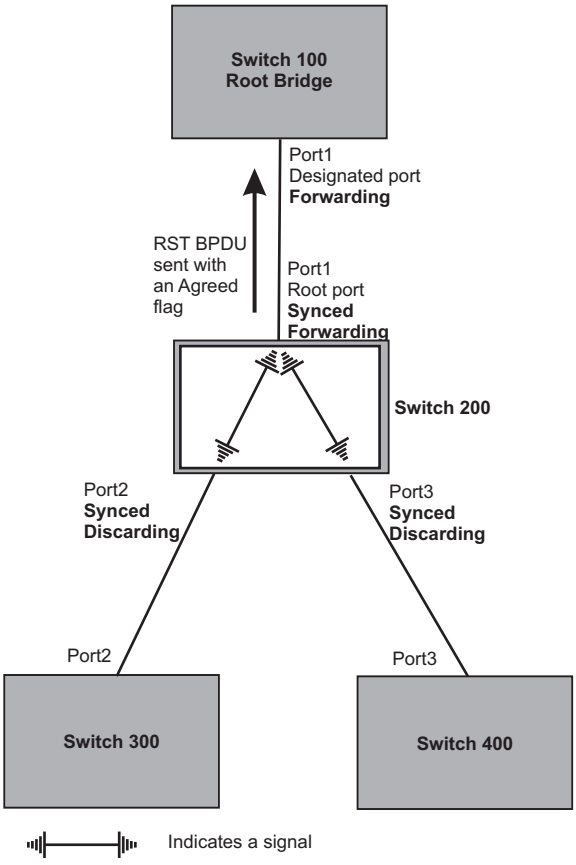
- Synced – Once the Designated port changes into a discarding state, it asserts a synced signal. Immediately, Alternate ports and Backup ports are synced. The Root port monitors the synced signals from all the bridge ports. Once all bridge ports asserts a synced signal, the Root port asserts its own synced signal (Figure 7.6).

Figure 7.6 Synced Stage



- Agreed – The Root port sends back an RST BPDU containing an agreed flag to its peer Designated port and moves into the forwarding state. When the peer Designated port receives the RST BPDU, it rapidly transitions into a forwarding state.

Figure 7.7 Agree Stage



At this point, the handshake mechanism is complete between Switch 100, the root bridge, and Switch 200.

Switch 200 updates the information on the Switch 200's Designated ports (Port2 and Port3) and identifies the new root bridge. The Designated ports send RST BPDUs, containing proposal flags, to their downstream bridges, without waiting for the hello timers to expire on them. This process starts the handshake with the downstream bridges.

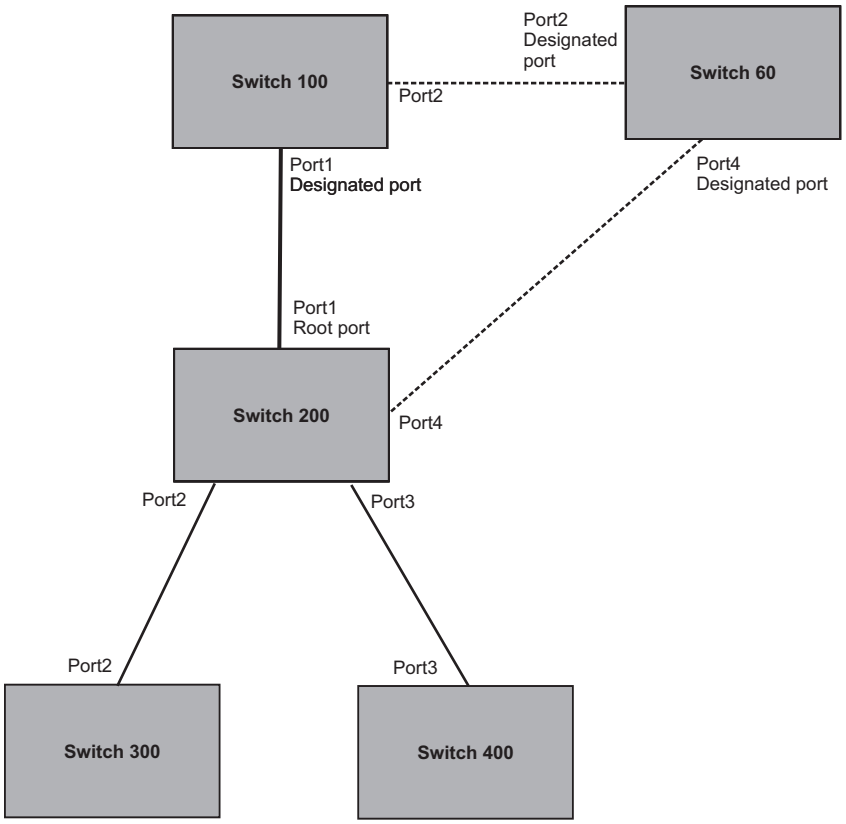
For example, Port2/Switch 200 sends an RST BPDU to Port2/Switch 300 that contains a proposal flag. Port2/ Switch 300 asserts a proposed signal. Ports in Switch 300 then set sync signals on the ports to synchronize and negotiate their roles and states. Then the ports assert a synced signal and when the Root port in Switch 300 asserts it's synced signal, it sends an RST BPDU to Switch 200 with an agreed flag.

This handshake is repeated between Switch 200 and Switch 400 until all Designated and Root ports are in forwarding states.

Handshake When a Root Port Has Been Elected

If a non-root bridge already has a Root port, 802.1W uses a different type of handshake. For example, in Figure 7.8, a new root bridge is added to the topology.

Figure 7.8 Addition of a New Root Bridge

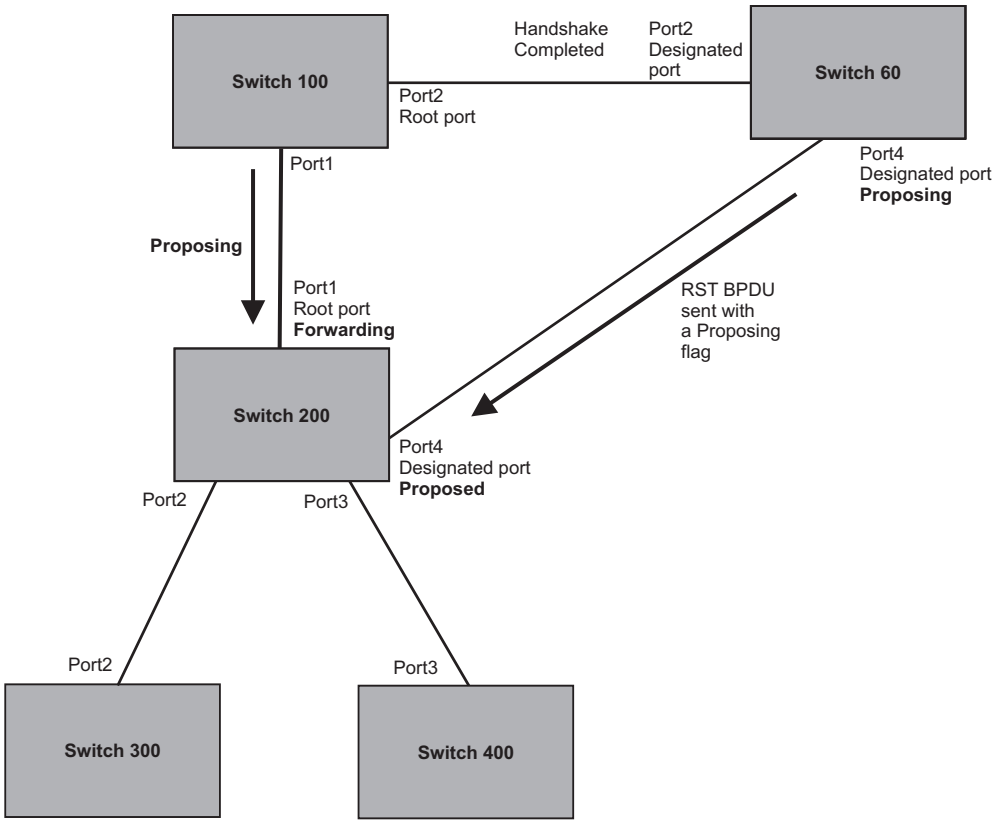


The handshake that occurs between Switch 60 and Switch 100 follows the one described in the previous section ("Handshake When No Root Port is Elected" on page 7-24). The former root bridge becomes a non-root bridge and establishes a Root port (Figure 7.9).

However, since Switch 200 already had a Root port in a forwarding state, 802.1W uses the Proposing -> Proposed -> Sync and Reroot -> Sync and Rerooted -> Rerooted and Synced -> Agreed handshake:

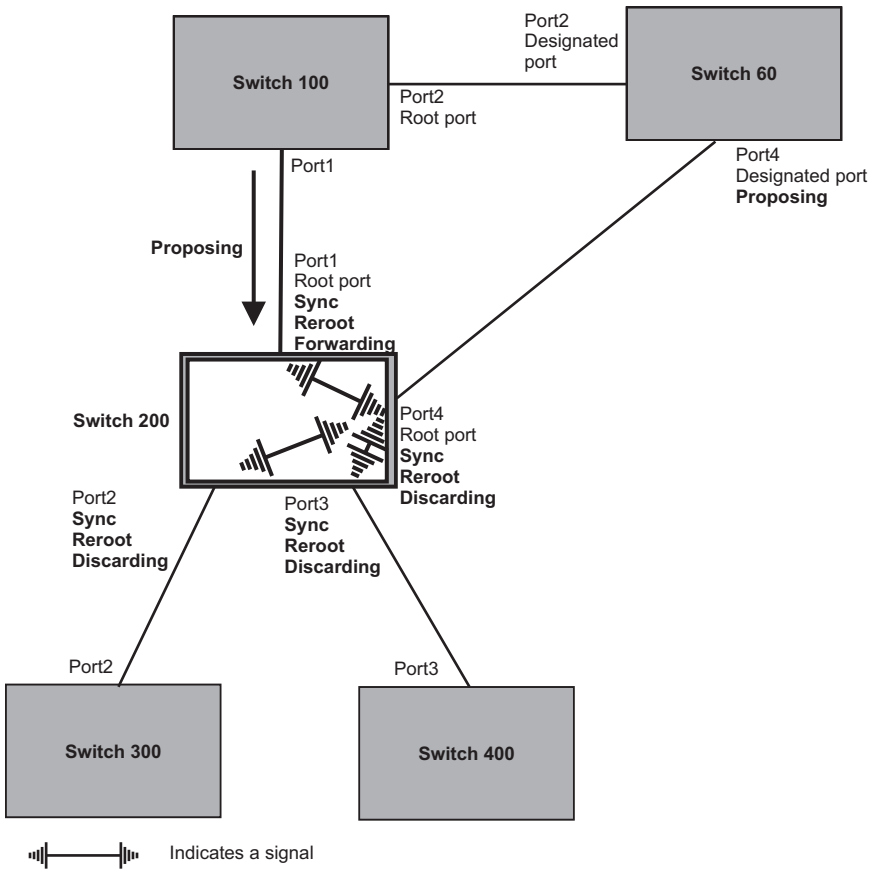
- Proposing and Proposed – The Designated port on the new root bridge (Port4/Switch 60) sends an RST BPDU that contains a proposing signal to Port4/Switch 200 to inform the port that it is ready to put itself in a forwarding state (Figure 7.9). 802.1W algorithm determines that the RST BPDU that Port4/Switch 200 received is superior to what it can generate, so Port4/Switch 200 assumes a Root port role.

Figure 7.9 New Root Bridge Sending a Proposal Flag



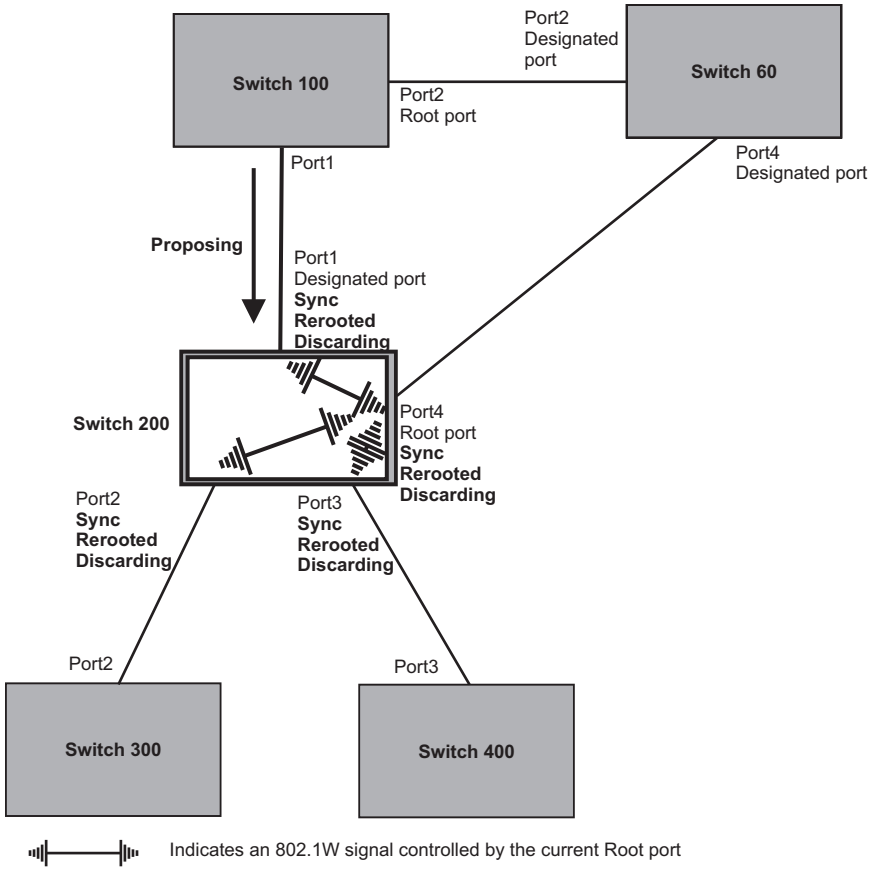
- Sync and Reroot – The Root port then asserts a sync and a reroot signal on all the ports on the bridge. The signal tells the ports that a new Root port has been assigned and they are to renegotiate their new roles and states. The other ports on the bridge assert their sync and reroot signals. Information about the old Root port is discarded from all ports. Designated ports change into discarding states (Figure 7.10).

Figure 7.10 Sync and Reroot



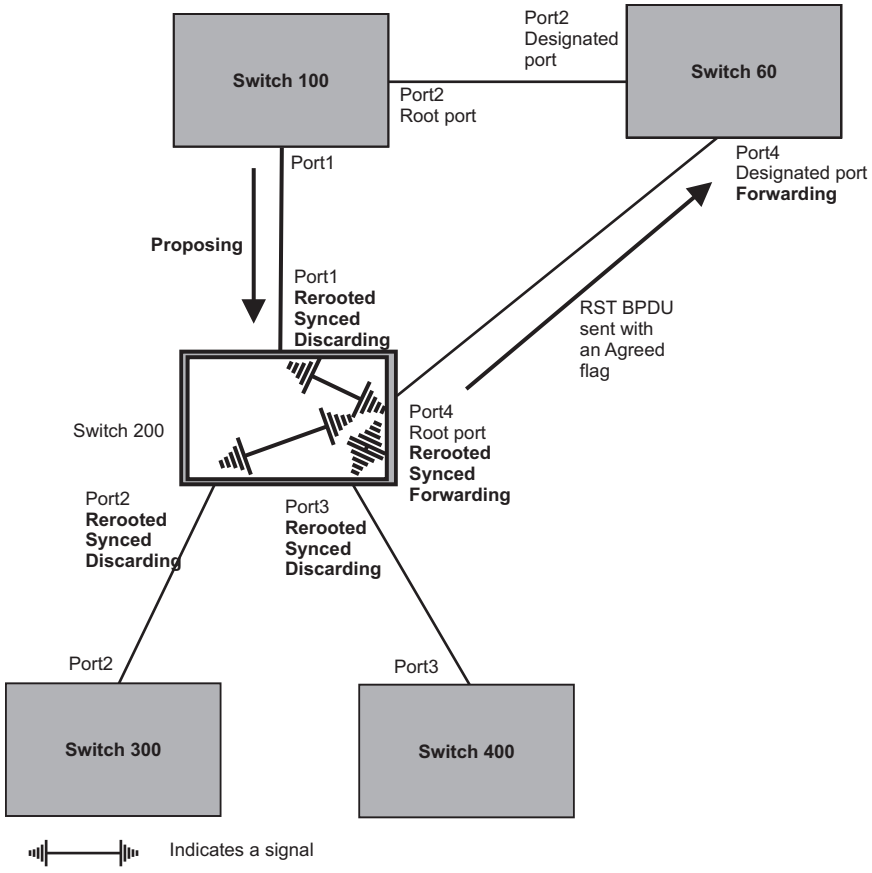
- Sync and Rerooted – When the ports on Switch 200 have completed the reroot phase, they assert their rerooted signals and continue to assert their sync signals as they continue in their discarding states. They also continue to negotiate their roles and states with their peer ports (Figure 7.11).

Figure 7.11 Sync and Rerooted



- Synced and Agree – When all the ports on the bridge assert their synced signals, the new Root port asserts its own synced signal and sends an RST BPDU to Port4/Switch 60 that contains an agreed flag (Figure 7.11). The Root port also moves into a forwarding state.

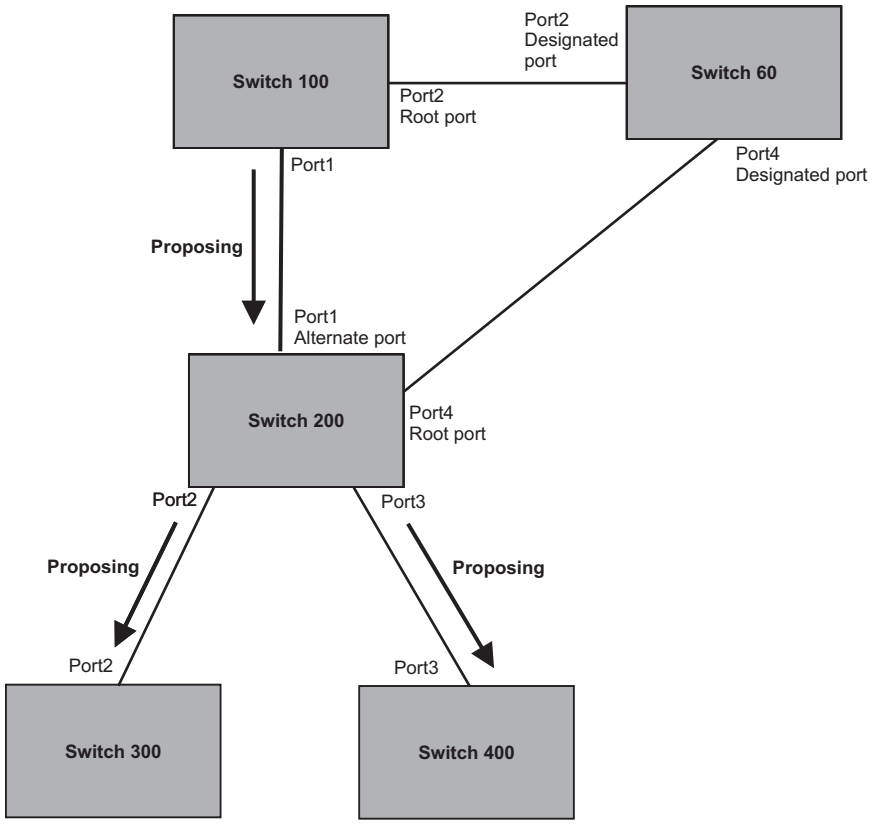
Figure 7.12 Rerooted, Synced, and Agreed



The old Root port on Switch 200 becomes an Alternate Port (Figure 7.13). Other ports on that bridge are elected to appropriate roles.

The Designated port on Switch 60 goes into a forwarding state once it receives the RST BPDU with the agreed flag.

Figure 7.13 Handshake Completed After Election of New Root Port



Recall that Switch 200 sent the agreed flag to Port4/Switch 60 and not to Port1/Switch 100 (the port that connects Switch 100 to Switch 200). Therefore, Port1/Switch 100 does not go into forwarding state instantly. It waits until two instances of the forward delay timer expires on the port before it goes into forwarding state.

At this point the handshake between the Switch 60 and Switch 200 is complete.

The remaining bridges (Switch 300 and Switch 400) may have to go through the reroot handshake if a new Root port needs to be assigned.

Convergence in a Simple Topology

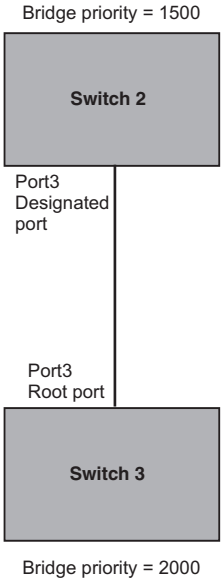
The examples in this section illustrate how 802.1W convergence occurs in a simple Layer 2 topology at start-up.

NOTE: The remaining examples assume that the appropriate handshake mechanisms occur as port roles and states change.

Convergence at Start Up

In Figure 7.14, two bridges Switch 2 and Switch 3 are powered up. There are point-to-point connections between Port3/Switch 2 and Port3/Switch 3.

Figure 7.14 Convergence Between Two Bridges



At power up, all ports on Switch 2 and Switch 3 assume Designated port roles and are at discarding states before they receive any RST BPDU.

Port3/Switch 2, with a Designated role, transmits an RST BPDU with a proposal flag to Port3/Switch 3. A ports with a Designated role sends the proposal flag in its RST BPDU when they are ready to move to a forwarding state.

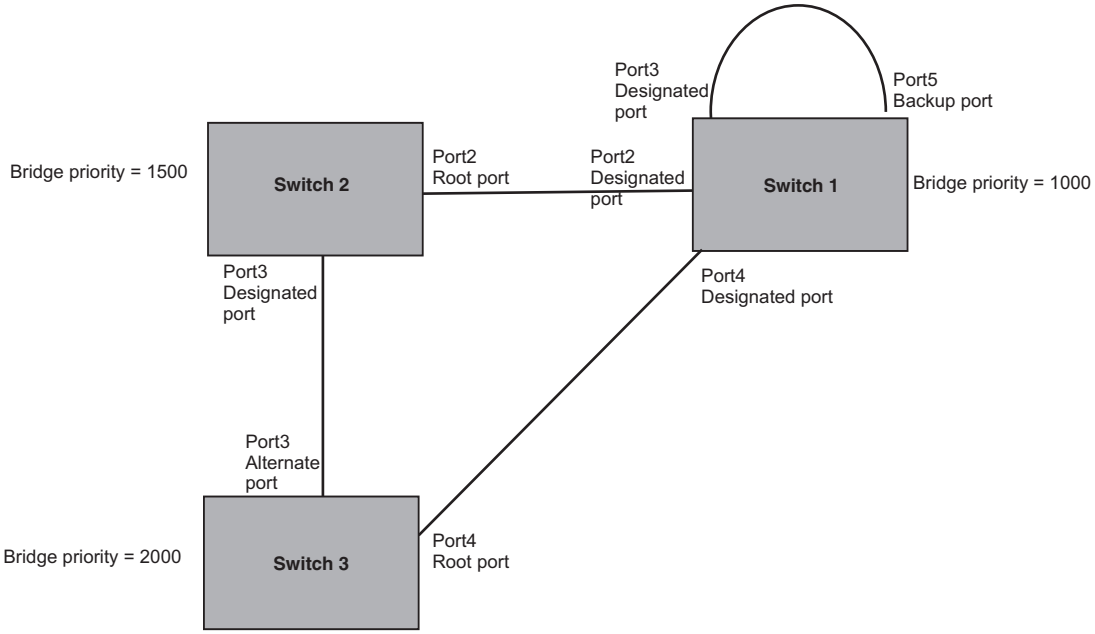
Port3/Switch 3, which starts with a role of Designated port, receives the RST BPDU and finds that it is superior to what it can transmit; therefore, Port3/Switch 3 assumes a new port role, that of a Root port. Port3/Switch 3 transmits an RST BPDU with an agreed flag back to Switch 2 and immediately goes into a forwarding state.

Port3/Switch 2 receives the RST BPDU from Port3/Switch 3 and immediately goes into a forwarding state.

Now 802.1W has fully converged between the two bridges, with Port3/Switch 3 as an operational root port in forwarding state and Port3/Switch 2 as an operational Designated port in forwarding state.

Next, Switch 1 is powered up (Figure 7.15).

Figure 7.15 Simple Layer 2 Topology



The point-to-point connections between the three bridges are as follows:

- Port2/Switch 1 and Port2/Switch 2
- Port4/Switch 1 and Port4/Switch 3
- Port3/Switch 2 and Port3/Switch 3

Ports 3 and 5 on Switch 1 are physically connected together.

At start up, the ports on Switch 1 assume Designated port roles, which are in discarding state. They begin sending RST BPDUs with proposal flags to move into a forwarding state.

When Port4/Switch 3 receives these RST BPDUs 802.1W algorithm determines that they are better than the RST BPDUs that were previously received on Port3/Switch 3. Port4/Switch 3 is now selected as Root port. This new assignment signals Port3/Switch 3 to begin entering the discarding state and to assume an Alternate port role. As it goes through the transition, Port3/Switch 3 negotiates a new role and state with its peer port, Port3/Switch 2.

Port4/Switch 3 sends an RST BPDU with an agreed flag to Port4/Switch 1. Both ports go into forwarding states.

Port2/Switch 2 receives an RST BPDU. The 802.1W algorithm determines that these RST BPDUs that are superior to any that any port on Switch 2 can transmit; therefore, Port2/Switch 2 assumes the role of a Root port.

The new Root port then signals all ports on the bridge to start synchronization. Since none of the ports are Edge ports, they all enter the discarding state and assume the role of Designated ports. Port3/Switch 2, which previously had a Designated role with a forwarding state, starts the discarding state. They also negotiate port roles and states with their peer ports. Port3/Switch 2 also sends an RST BPU to Port3/Switch 3 with a proposal flag to request permission go into a forwarding state.

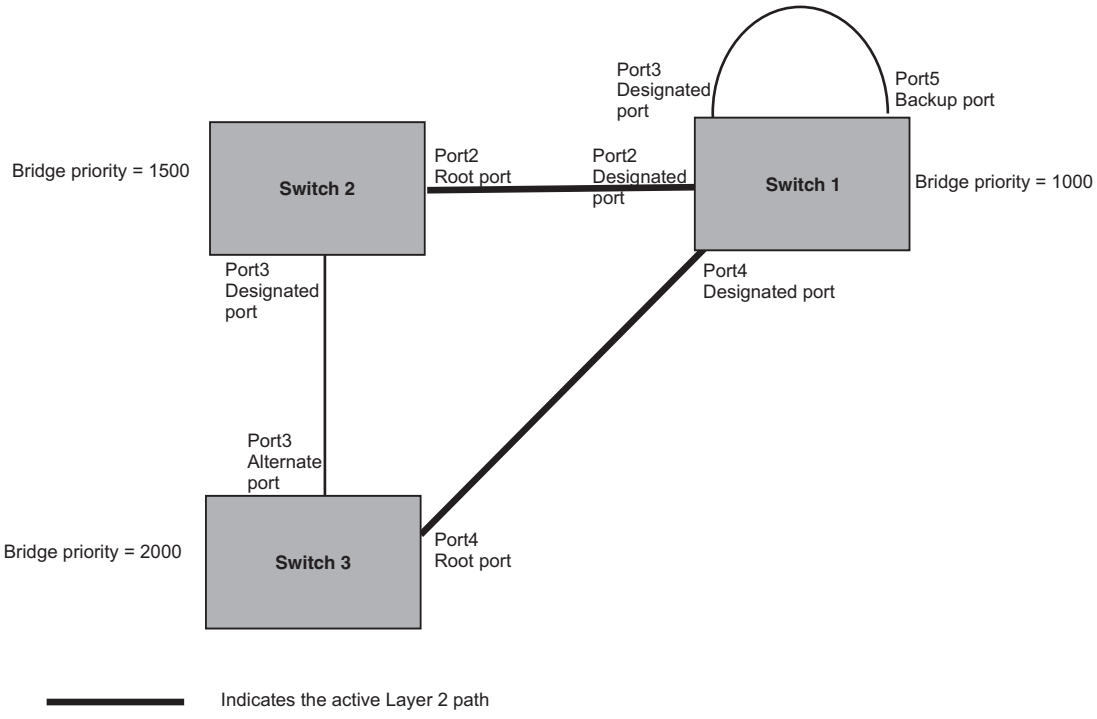
The Port2/Switch 2 bridge also sends an RST BPDU with an agreed flag Port2/Switch 1 that Port2 is the new Root port. Both ports go into forwarding states.

Now, Port3/Switch 3 is currently in a discarding state and is negotiating a port role. It received RST BPDUs from Port3/Switch 2. The 802.1W algorithm determines that the RST BPDUs Port3/Switch 3 received are superior to those it can transmit; however, they are not superior to those that are currently being received by the current Root port (Port4). Therefore, Port3 retains the role of Alternate port.

Ports 3/Switch 1 and Port5/Switch 1 are physically connected. Port5/Switch 1 received RST BPDUs that are superior to those received on Port3/Switch 1; therefore, Port5/Switch 1 is given the Backup port role while Port3 is given the Designated port role. Port3/Switch 1, does not go directly into a forwarding state. It waits until the forward delay time expires twice on that port before it can proceed to the forwarding state.

Once convergence is achieved, the active Layer 2 forwarding path converges as shown in Figure 7.16.

Figure 7.16 Active Layer 2 Path

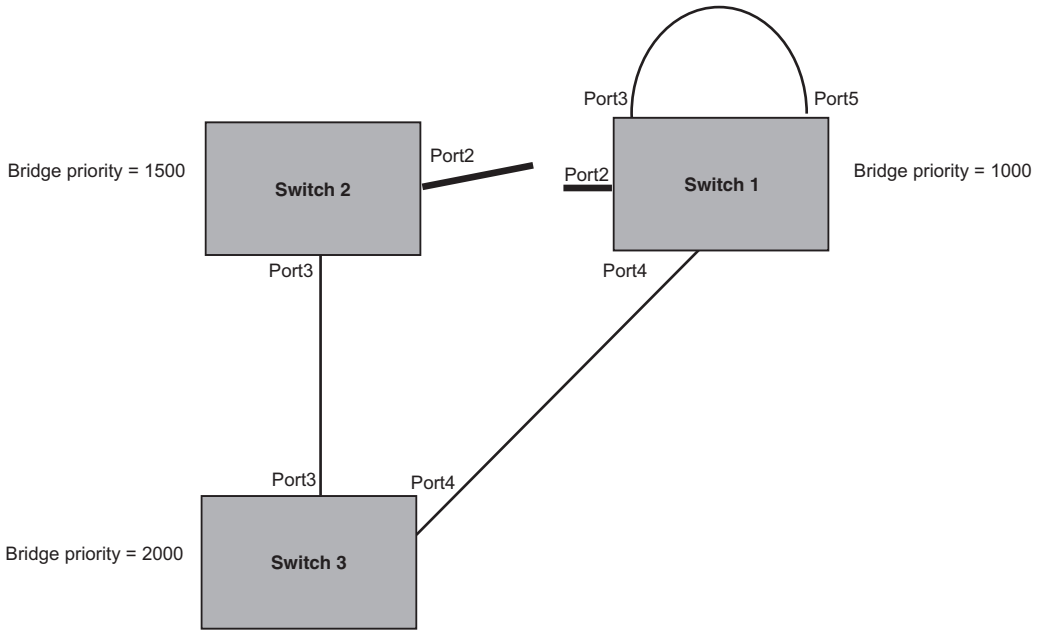


Convergence After a Link Failure

What happens if a link in the 802.1W topology fails?

For example, Port2/Switch, which is the port that connects Switch 2 to the root bridge (Switch 1), fails. Both Switch 2 and Switch 1 notice the topology change (Figure 7.17).

Figure 7.17 Link Failure in the Topology



Switch 1 sets its Port2 into a discarding state.

At the same time, Switch 2 assumes the role of a root bridge since its root port failed and it has no operational Alternate port. Port3/Switch 2, which currently has a Designated port role, sends an RST BPDU to Switch 3. The RST BPDU contains a proposal flag and a bridge ID of Switch 2 as its root bridge ID.

When Port3/Switch 3 receives the RST BPDUs, 802.1W algorithm determines that they are inferior to those that the port can transmit. Therefore, Port3/Switch 3 is given a new role, that of a Designated port. Port3/Switch 3 then sends an RST BPDU with a proposal flag to Switch 2, along with the new role information. However, the root bridge ID transmitted in the RST BPDU is still Switch 1.

When Port3/Switch 2 receives the RST BPDU, 802.1W algorithm determines that it is superior to the RST BPDU that it can transmit; therefore, Port3/Switch 2 receives a new role; that of a Root port. Port3/Switch 2 then sends an RST BPDU with an agreed flag to Port3/Switch 3. Port3/Switch 2 goes into a forwarding state.

When Port3/Switch 3 receives the RST BPDU that Port3/Switch 2 sent, Port3/Switch 3 changes into a forwarding state, which then completes the full convergence of the topology.

Convergence at Link Restoration

When Port2/Switch 2 is restored, both Switch 2 and Switch 1 recognize the change. Port2/Switch 1 starts assuming the role of a Designated port and sends an RST BPDU containing a proposal flag to Port2/Switch 2.

When Port2/Switch 2 receives the RST BPDUs, 802.1W algorithm determines that the RST BPDUs the port received are better than those received on Port3/Switch 3; therefore, Port2/Switch 2 is given the role of a Root port. All the ports on Switch 2 are informed that a new Root port has been assigned which then signals all the ports to synchronize their roles and states. Port3/Switch 2, which was the previous Root port, enters a discarding state and negotiates with other ports on the bridge to establish its new role and state, until it finally assumes the role of a Designated port.

Next, the following happens:

- Port3/Switch 2, the Designated port, sends an RST BPDU, with a proposal flag to Port3/Switch 3.
- Port2/Switch 2 also sends an RST BPDU with an agreed flag to Port2/Switch 1 and then places itself into a forwarding state.

When Port2/Switch 1 receives the RST BPDU with an agreed flag sent by Port2/Switch 2, it puts that port into a forwarding state. The topology is now fully converged.

When Port3/Switch 3 receives the RST BPDU that Port3/Switch 2 sent, 802.1W algorithm determines that these RST BPDUs are superior to those that Port3/Switch 3 can transmit. Therefore, Port3/Switch 3 is given a new role, that of an Alternate port. Port3/Switch 3 immediately enters a discarding state.

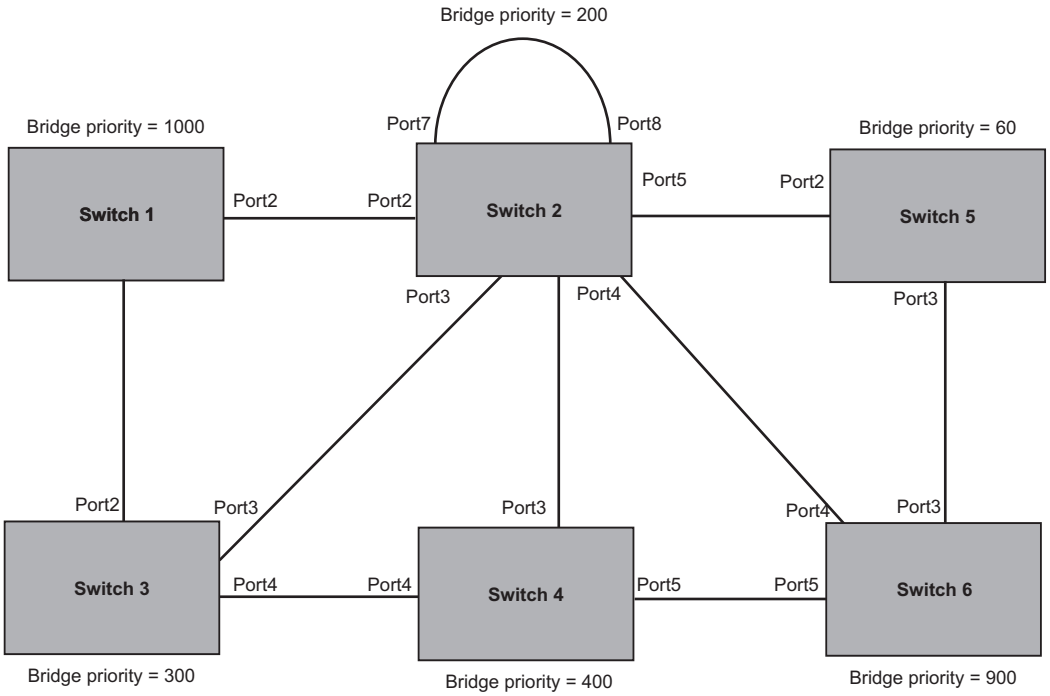
Now Port3/Switch 2 does not go into a forwarding state instantly like the Root port. It waits until the forward delay timer expires twice on that port while it is still in a Designated role, before it can proceed to the forwarding state. The wait, however, does not cause a denial of service, since the essential connectivity in the topology has already been established.

When fully restored, the topology is the same as that shown on Figure 7.15.

Convergence in a Complex 802.1W Topology

The following is an example of a complex 802.1W topology.

Figure 7.18 Complex 802.1W Topology



In Figure 7.18, Switch 5 is selected as the root bridge since it is the bridge with the highest priority. Lines in the figure show the point-to-point connection to the bridges in the topology.

Switch 5 sends an RST BPDU that contains a proposal flag to Port5/Switch 2. When handshakes are completed in Switch 5, Port5/Switch 2 is selected as the Root port on Switch 2. All other ports on Switch 2 are given Designated port role with discarding states.

Port5/Switch 2 then sends an RST BPDU with an agreed flag to Switch 5 to confirm that it is the new Root port and the port enters a forwarding state. Port7 and Port8 are informed of the identity of the new Root port. 802.1W algorithm selects Port7 as the Designated port while Port8 becomes the Backup port.

Port3/Switch 5 sends an RST BPDU to Port3/Switch 6 with a proposal flag. When Port3/Switch 5 receives the RST BPDU, handshake mechanisms select Port3 as the Root port of Switch 6. All other ports are given a Designated port role with discarding states. Port3/Switch 6 then sends an RST BPDU with an agreed flag to Port3/Switch 5 to confirm that it is the Root port. The Root port then goes into a forwarding state.

Now, Port4/Switch 6 receives RST BPDUs that are superior to what it can transmit; therefore, it is given the Alternate port role. The port remains in discarding state.

Port5/Switch 6 receives RST BPDUs that are inferior to what it can transmit. The port is then given a Designated port role.

Next Switch 2 sends RST BPDUs with a proposal flag to Port3/Switch 4. Port3 becomes the Root port for the bridge; all other ports are given a Designated port role with discarding states. Port3/Switch 4 sends an RST BPDU with an agreed flag to Switch 2 to confirm that it is the new Root port. The port then goes into a forwarding state.

Now Port4/Switch 4 receives an RST BPDU that is superior to what it can transmit. The port is then given an Alternate port role, and remains in discarding state.

Likewise, Port5/Switch 4 receives an RST BPDU that is superior to what it can transmit. The port is also given an Alternate port role, and remains in discarding state.

Port2/Switch 2 transmits an RST BPDU with a proposal flag to Port2/Switch 1. Port2/Switch 1 becomes the Root port. All other ports on Switch 1 are given Designated port roles with discarding states.

Port2/Switch 1 sends an RST BPDU with an agreed flag to Port2/Switch 2 and Port2/Switch 1 goes into a forwarding state.

Port3/Switch 1 receives an RST BPDUs that is inferior to what it can transmit; therefore, the port retains its Designated port role and goes into forwarding state only after the forward delay timer expires twice on that port while it is still in a Designated role.

Port3/Switch 2 sends an RST BPDU to Port3/Switch 3 that contains a proposal flag. Port3/Switch 3 becomes the Root port, while all other ports on Switch 3 are given Designated port roles and go into discarding states. Port3/Switch 3 sends an RST BPDU with an agreed flag to Port3/Switch 2 and Port3/Switch 3 goes into a forwarding state.

Now, Port2/Switch 3 receives an RST BPDUs that is superior to what it can transmit so that port is given an Alternate port state.

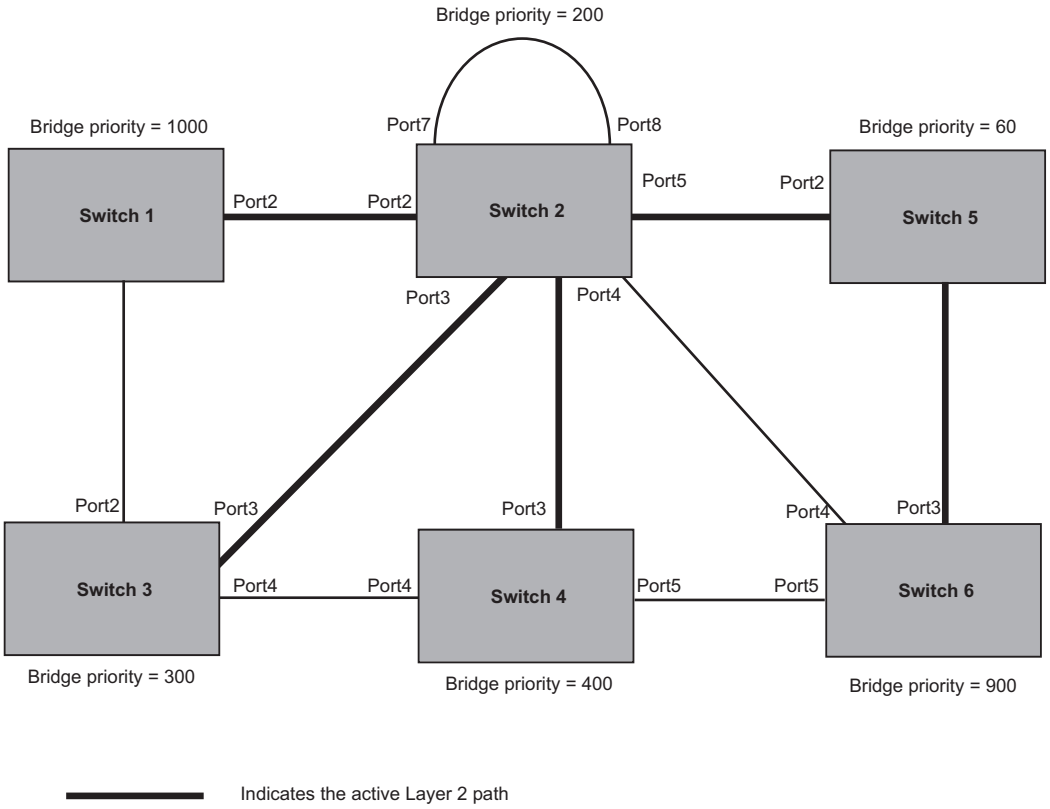
Port4/Switch 3 receives an RST BPDU that is inferior to what it can transmit; therefore, the port retains its Designated port role.

Ports on all the bridges in the topology with Designated port roles that received RST BPDUs with agreed flags go into forwarding states instantly. However, Designated ports that did not receive RST BPDUs with agreed flags must wait until the forward delay timer expires twice on those port. Only then will these port move into forwarding states.

The entire 802.1W topology converges in less than 300 msec and the essential connectivity is established between the designated ports and their connected root ports.

After convergence is complete, Figure 7.19 shows the active Layer 2 path of the topology in Figure 7.18.

Figure 7.19 Active Layer 2 Path in Complex Topology



Propagation of Topology Change

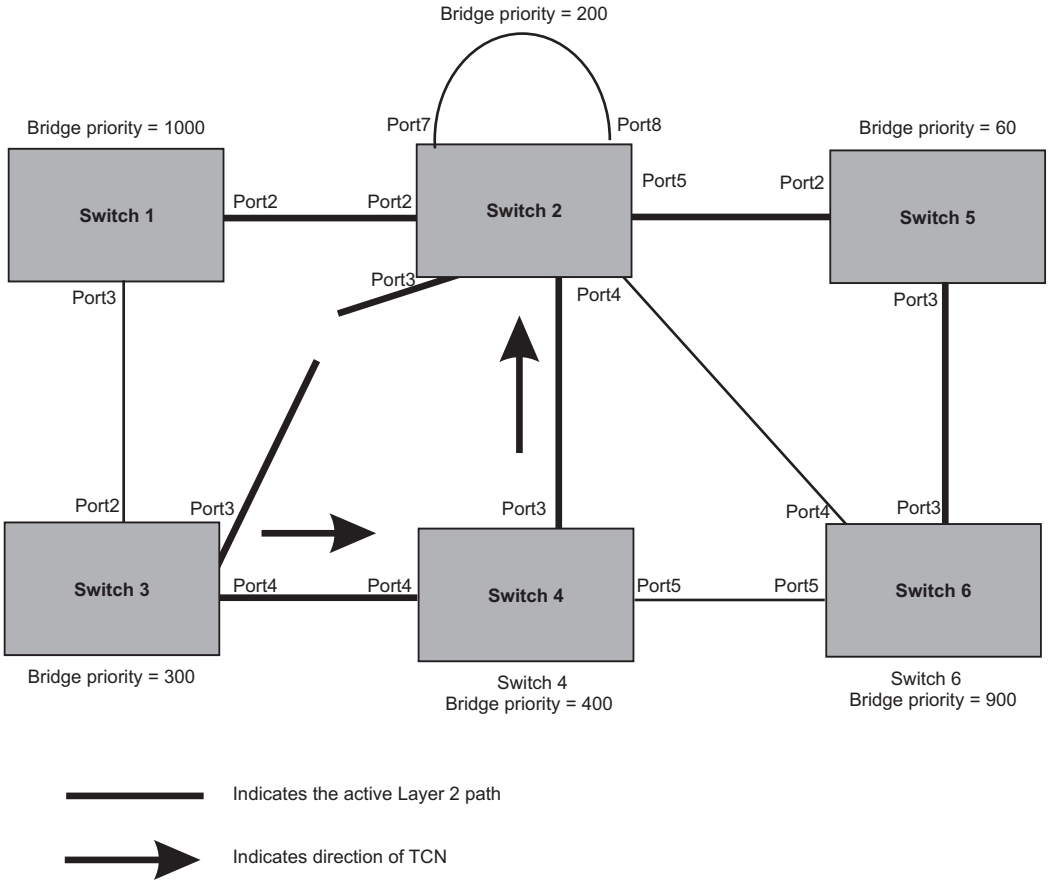
The Topology Change state machine generates and propagates the topology change notification messages on each port. When a Root port or a Designated port goes into a forwarding state, the Topology Change state machine on those ports send a topology change notice (TCN) to all the bridges in the topology to propagate the topology change.

NOTE: Edge ports, Alternate ports, or Backup ports do not need to propagate a topology change.

The TCN is sent in the RST BPDU that a port sends. Ports on other bridges in the topology then acknowledge the topology change once they receive the RST BPDU, and send the TCN to other bridges until all the bridges are informed of the topology change.

For example, Port3/Switch 2 in Figure 7.20, fails. Port4/Switch 3 becomes the new Root port. Port4/Switch 3 sends an RST BPDU with a TCN to Port4/Switch 4. To propagate the topology change, Port4/Switch 4 then starts a TCN timer on itself, on the bridge's Root port, and on other ports on that bridge with a Designated role. Then Port3/Switch 4 sends RST BPDU with the TCN to Port4/Switch 2. (Note the new active Layer 2 path in Figure 7.20.)

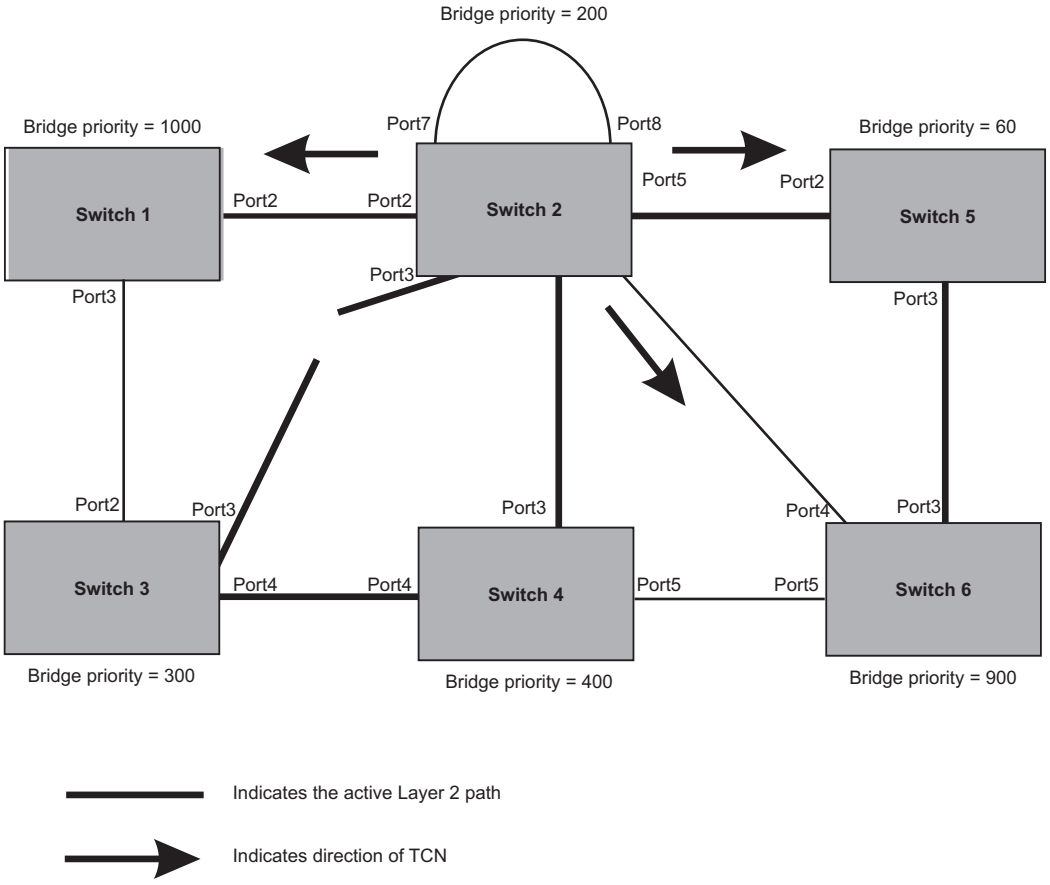
Figure 7.20 Beginning of Topology Change Notice



Switch 2 then starts the TCN timer on the Designated ports and sends RST BPDUs that contain the TCN as follows (Figure 7.21):

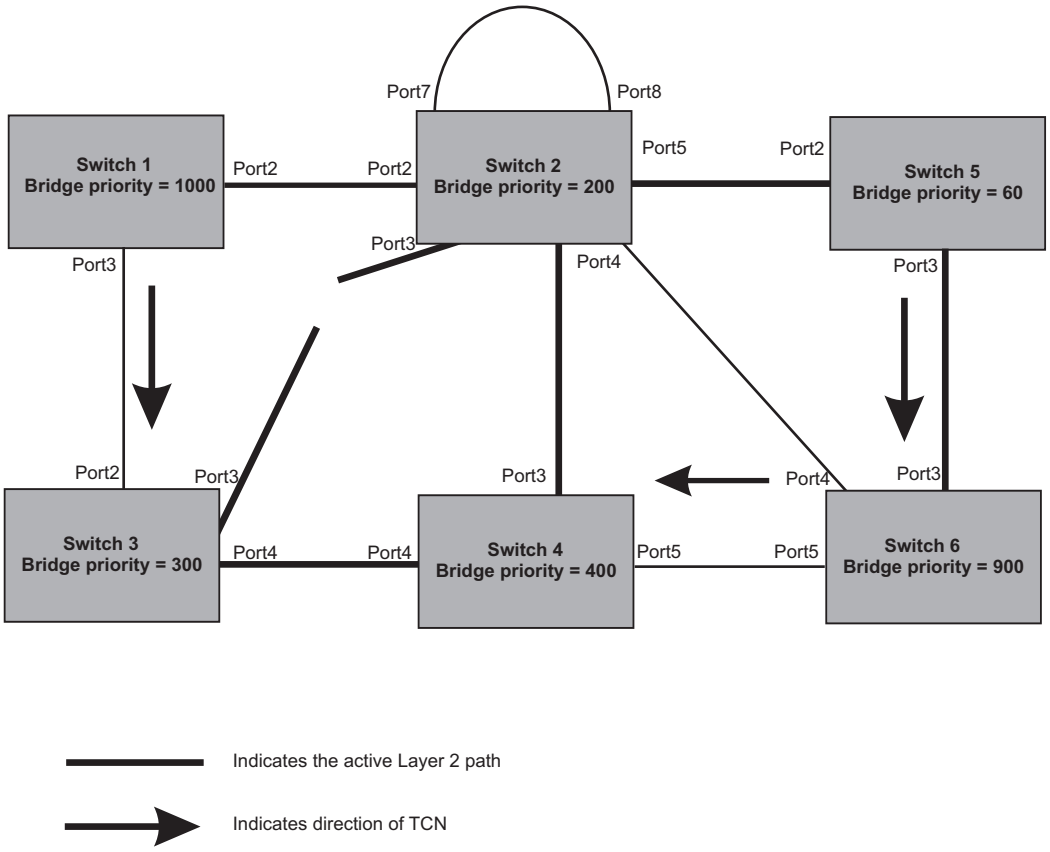
- Port5/Switch 2 sends the TCN to Port2/Switch 5
- Port4/Switch 2 sends the TCN to Port4/Switch 6
- Port2/Switch 2 sends the TCN to Port2/Switch 1

Figure 7.21 Sending TCN to Bridges Connected to Switch 2



Then Switch 1, Switch 5, and Switch 6 send RST BPDUs that contain the TCN to Switch 3 and Switch 4 to complete the TCN propagation (Figure 7.22).

Figure 7.22 Completing the TCN Propagation



Compatibility of 802.1W with 802.1D

802.1W-enabled bridges are backward compatible with IEEE 802.1D bridges. This compatibility is managed on a per-port basis by the Port Migration state machine. **However, intermixing the two types of bridges in the network topology is not advisable if you want to take advantage of the rapid convergence feature.**

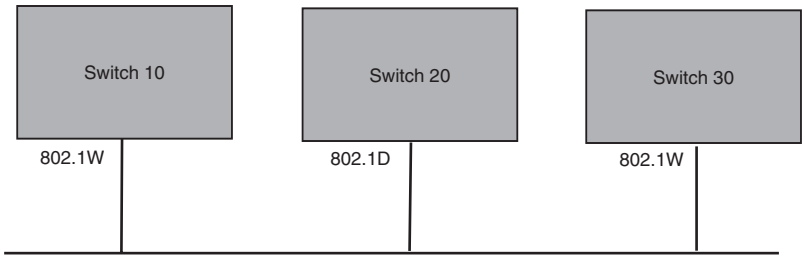
Compatibility with 802.1D means that an 802.1W-enabled port can send BPDUs in the STP or 802.1D format when one of the following events occur:

- The port receives a legacy BPDU. A legacy BPDU is an STP BPDU or a BPDU in an 802.1D format. The port that receives the legacy BPDU automatically configures itself to behave like a legacy port. It sends and receives legacy BPDUs only.
- The entire bridge is configured to operate in an 802.1D mode when an administrator sets the bridge parameter to zero at the CLI, forcing all ports on the bridge to send legacy BPDUs only.

Once a port operates in the 802.1D mode, 802.1D convergence times are used and rapid convergence is not realized.

For example, in Figure 7.23, Switch 10 and Switch 30 receive legacy BPDUs from Switch 20. Ports on Switch 10 and Switch 30 begin sending BPDUs in STP format to allow them to operate transparently with Switch 20.

Figure 7.23 802.1W Bridges with an 802.1D Bridge



Once Switch 20 is removed from the LAN, Switch 10 and Switch 30 receive and transmit BPDUs in the STP format to and from each other. This state will continue until the administrator enables the **force-migration-check** command to force the bridge to send RSTP BPDU during a migrate time period. If ports on the bridges continue to hear only STP BPDUs after this migrate time period, those ports will return to sending STP BPDUs. However, when the ports receive RST BPDUs during the migrate time period, the ports begin sending RST BPDUs. The migrate time period is non-configurable. It has a value of three seconds.

NOTE: The IEEE standards state that 802.1W bridges need to interoperate with 802.1D bridges. IEEE standards set the path cost of 802.1W bridges to be between 1 and 200,000,000; whereas path cost of 802.1D bridges are set between 1 and 65,535. In order for the two bridge types to be able to interoperate in the same topology, the administrator needs to configure the bridge path cost appropriately. Path costs for either 802.1W bridges or 802.1D bridges need to be changed; in most cases, path costs for 802.1W bridges need to be changed.

Configuring 802.1W Parameters on a Foundry Device

The remaining 802.1W sections explain how to configure the 802.1W protocol in a Foundry device.

Foundry devices are shipped from the factory with 802.1W disabled. Use the following methods to enable or disable 802.1W. You can enable or disable 802.1W at the following levels:

- Port-based VLAN – Affects all ports within the specified port-based VLAN. When you enable or disable 802.1W within a port-based VLAN, the setting overrides the global setting. Thus, you can enable 802.1W for the ports within a port-based VLAN even when 802.1W is globally disabled, or disable the ports within a port-based VLAN when 802.1W is globally enabled.
- Individual port – Affects only the individual port. However, if you change the 802.1W state of the primary port in a trunk group, the change affects all ports in the trunk group.

Enabling or Disabling 802.1W in a Port-Based VLAN

Use the following procedure to disable or enable 802.1W on a device on which you have configured a port-based VLAN. Changing the 802.1W state in a VLAN affects only that VLAN.

To enable 802.1W for all ports in a port-based VLAN, enter commands such as the following:

```
FESX424 Router(config)# vlan 10
FESX424 Router(config-vlan-10)# spanning-tree 802-1w
```

Syntax: [no] spanning-tree 802-1w

Enabling or Disabling 802.1W on a Single Spanning Tree

To enable 802.1W for all ports of a single spanning tree, enter a command such as the following:

```
FESX424 Router(config-vlan-10)# spanning-tree single 802-1w
```

Syntax: [no] spanning-tree single 802-1w

Disabling or Enabling 802.1W on an Individual Port

The **spanning-tree 802-1w** or **spanning-tree single 802-1w** command must be used to initially enable 802.1W on ports. Both commands enable 802.1W on all ports that belong to the VLAN or to the single spanning tree.

Once 802.1W is enabled on a port, it can be disabled on individual ports. 802.1W that have been disabled on individual ports can then be enabled as required.

NOTE: If you change the 802.1W state of the primary port in a trunk group, the change affects all ports in that trunk group.

To disable or enable 802.1W on an individual port, enter commands such as the following:

```
FESX424 Router(config)# interface e 1
FESX424 Router(config-if-e1000-1)# no spanning-tree
```

Syntax: [no] spanning-tree

Changing 802.1W Bridge Parameters

When you make changes to 802.1W bridge parameters, the changes are applied to individual ports on the bridge. To change 802.1W bridge parameters, use the following methods.

To designate a priority for a bridge, enter a command such as the following:

```
FESX424 Router(config)# spanning-tree 802-1w priority 10
```

The command in this example changes the priority on a device on which you have not configured port-based VLANs. The change applies to the default VLAN. If you have configured a port-based VLAN on the device, you can configure the parameters only at the configuration level for individual VLANs. Enter commands such as the following:

```
FESX424 Router(config)# vlan 20
FESX424 Router(config-vlan-20)# spanning-tree 802-1w priority 0
```

To make this change in the default VLAN, enter the following commands:

```
FESX424 Router(config)# vlan 1
FESX424 Router(config-vlan-1)# spanning-tree 802-1w priority 0
```

Syntax: spanning-tree 802-1w [forward-delay <value>] | [hello-time <value>] | [max-age <time>] | [force-version <value>] | [priority <value>]

The **forward-delay** <value> parameter specifies how long a port waits before it forwards an RST BPDU after a topology change. This can be a value from 4 – 30 seconds. The default is 15 seconds.

The **hello-time** <value> parameter specifies the interval between two hello packets. This parameter can have a value from 1 – 10 seconds. The default is 2 seconds; however, set this value to at least 4 seconds to provide enough time for BPDUs to reach the root bridge before the timeout period expires on a non-root bridge port.

The **max-age** <value> parameter specifies the amount of time the device waits to receive a hello packet before it initiates a topology change. You can specify a value from 6 – 40 seconds. The default is 20 seconds.

The value of **max-age** must be greater than the value of **forward-delay** to ensure that the downstream bridges do not age out faster than the upstream bridges (those bridges that are closer to the root bridge).

The **force-version** <value> parameter forces the bridge to send BPDUs in a specific format. You can specify one of the following values:

- 0 – The STP compatibility mode. Only STP (or legacy) BPDUs will be sent.
- 2 – The default. RST BPDUs will be sent unless a legacy bridge is detected. If a legacy bridge is detected, STP BPDUs will be sent instead.

The default is 2.

The **priority** <value> parameter specifies the priority of the bridge. You can enter a value from 0 – 65535. A lower numerical value means a the bridge has a higher priority. Thus, the highest priority is 0. The default is 32768.

You can specify some or all of these parameters on the same command line. If you specify more than one parameter, you must specify them in the order shown above, from left to right.

Changing Port Parameters

The 802.1W port commands can be enabled on individual ports or on multiple ports, such as all ports that belong to a VLAN.

The 802.1W port parameters are preconfigured with default values. If the default parameters meet your network requirements, no other action is required.

You can change the following 802.1W port parameters using the following method.

```
FESX424 Router(config)# vlan 10
FESX424 Router(config-vlan-10)# spanning-tree 802-1w ethernet 5 path-cost 15
priority 64
```

Syntax: spanning-tree 802-1w ethernet [<slotnum>/]<portnum> path-cost <value> | priority <value> | [admin-edge-port] | [admin-pt2pt-mac] | [force-migration-check]

The <portnum> parameter specifies the interface used. If you are configuring a chassis device, specify the slot number as well as the port number (<slotnum>/<portnum>).

The **path-cost** <value> parameter specifies the cost of the port's path to the root bridge. 802.1W prefers the path with the lowest cost. You can specify a value from 1 – 20,000,000. Table 7.7 shows the recommended path cost values from the IEEE standards.

Table 7.7: Recommended Path Cost Values of 802.1W

Link Speed	Recommended (Default) 802.1W Path Cost Values	Recommended 802.1W Path Cost Range
Less than 100 kilobits per second	200,000,000	20,000,000 – 200,000,000
1 Megabit per second	20,000,000	2,000,000 – 200,000,000
10 Megabits per second	2,000,000	200,000 – 200,000,000
100 Megabits per second	200,000	20,000 – 200,000,000
1 Gigabit per second	20,000	2,000 – 200,000,000
10 Gigabits per second	2,000	200 – 20,000
100 Gigabits per second	200	20 – 2,000
1 Terabits per second	20	2 – 200
10 Terabits per second	2	1 – 20

The **priority** <value> parameter specifies the preference that 802.1W gives to this port relative to other ports for forwarding traffic out of the topology. You can specify a value from 8 – 252, in increments of 4. If you enter a value that is not divisible by four the software rounds to the nearest value that is. The default is 128. A higher numerical value means a lower priority; thus, the highest priority is 8

Set the **admin-edge-port** to enabled or disabled. If set to enabled, then the port becomes an edge port in the domain.

Set the **admin-pt2pt-mac** to enabled or disabled. If set to enabled, then a port is connected to another port through a point-to-point link. The point-to-point link increases the speed of convergence. This parameter, however, does not auto-detect whether or not the link is a physical point-to-point link.

The **force-migration-check** parameter forces the specified port to sent one RST BPDU. If only STP BPDUs are received in response to the sent RST BPDU, then the port will go return to sending STP BPDUs.

EXAMPLE:

Suppose you want to enable 802.1W on a system with no active port-based VLANs and change the hello-time from the default value of 2 to 8 seconds. Additionally, suppose you want to change the path and priority costs for port 5 only. To do so, enter the following commands.

```
FESX424 Router(config)# spanning-tree 802-1w hello-time 8

FESX424 Router(config)# spanning-tree 802-1w ethernet 5 path-cost 15 priority 64
```

Displaying Information About 802-1W

To display a summary of 802-1W, use the following command:

```
FastIron SuperX Router(config)#show 802-1w

--- VLAN 1 [ STP Instance owned by VLAN 1 ] ----------------------------

VLAN 1 BPDU cam_index is 2 and the IGC and DMA master Are(HEX) 0 1 2 3

Bridge IEEE 802.1W Parameters:

Bridge            Bridge  Bridge Bridge Force   tx
Identifier        MaxAge  Hello  FwdDly Version Hold
hex               sec     sec    sec            cnt
800000e080541700  20      2      15     Default 3


RootBridge        RootPath   DesignatedBri-    Root  Max  Fwd  Hel
Identifier        Cost       dge Identifier    Port  Age  Dly  lo
hex                          hex                     sec  sec  sec
800000e0804c9c00  200000     800000e0804c9c00  1     20   15   2


Port IEEE 802.1W Parameters:

      <--- Config Params -->|<-------------- Current state ----------------->
Port  Pri PortPath P2P Edge Role       State      Designa- Designated
Num       Cost     Mac Port                     ted cost bridge
1     128 200000   F   F    ROOT       FORWARDING 0        800000e0804c9c00
2     128 200000   F   F    DESIGNATED FORWARDING 200000   800000e080541700
3     128 200000   F   F    DESIGNATED FORWARDING 200000   800000e080541700
4     128 200000   F   F    BACKUP     DISCARDING 200000   800000e080541700
```

Syntax: show 802-1w [vlan <vlan-id>]

The **vlan** <vlan-id> parameter displays 802.1W information for the specified port-based VLAN.

The **show 802.1w display** command shows the information listed in Table 7.8.

Table 7.8: CLI Display of 802.1W Summary

This Field...	Displays...
VLAN ID	The port-based VLAN that owns the STP instance. VLAN 1 is the default VLAN. If you have not configured port-based VLANs on this device, all 802.1W information is for VLAN 1.

Table 7.8: CLI Display of 802.1W Summary (Continued)

This Field...	Displays...
Bridge IEEE 802.1W Parameters	
Bridge Identifier	The ID of the bridge.
Bridge Max Age	The configured max age for this bridge. The default is 20.
Bridge Hello	The configured hello time for this bridge.The default is 2.
Bridge FwdDly	The configured forward delay time for this bridge. The default is 15.
Force-Version	The configured force version value. One of the following value is displayed: • 0 – The bridge has been forced to operate in an STP compatibility mode. • 2 – The bridge has been forced to operate in an 802.1W mode. (This is the default.)
txHoldCnt	The number of BPDUs that can be transmitted per Hello Interval. The default is 3.
Root Bridge Identifier	ID of the Root bridge that is associated with this bridge
Root Path Cost	The cost to reach the root bridge from this bridge. If the bridge is the root bridge, then this parameter shows a value of zero.
Designated Bridge Identifier	The bridge from where the root information was received.It can be from the root bridge itself, but it could also be from another bridge.
Root Port	The port on which the root information was received. This is the port that is connected to the Designated Bridge.
Max Age	The ***max age*** is derived from the Root port. An 802.1W-enabled bridge uses this value, along with the hello and message age parameters to compute the effective age of an RST BPDU. The ***message age*** parameter is generated by the Designated port and transmitted in the RST BPDU. RST BPDUs transmitted by a Designated port of the root bridge contains a message value of zero. ***Effective age*** is the amount of time the Root port, Alternate port, or Backup port retains the information it received from its peer Designated port. Effective age is reset every time a port receives an RST BPDU from its Designated port. If a Root port does not receive an RST BPDU from its peer Designated port for a duration more than the effective age, the Root port ages out the existing information and recomputes the topology. If the port is operating in 802.1D compatible mode, then max age functionality is the same as in 802.1D (STP).

Table 7.8: CLI Display of 802.1W Summary (Continued)

This Field...	Displays...
Fwd Dly	The number of seconds a non-edge Designated port waits until it can apply any of the following transitions, if the RST BPDU it receives does not have an agreed flag: • Discarding state to learning state • Learning state to forwarding state When a non-edge port receives the RST BPDU it goes into forwarding state within 4 seconds or after two hello timers expire on the port. Fwd Dly is also the number of seconds that a Root port waits for an RST BPDU with a proposal flag before it applies the state transitions listed above. If the port is operating in 802.1D compatible mode, then forward delay functionality is the same as in 802.1D (STP).
Hello	The hello value derived from the Root port. It is the number of seconds between two Hello packets.
Port IEEE 802.1W Parameters	
Port Num	The port number shown in a slot#/port# format.
Pri	The configured priority of the port. The default is 128 or 0x80.
Port Path Cost	The configured path cost on a link connected to this port.
P2P Mac	Indicates if the point-to-point-mac parameter is configured to be a point-to-point link: • T – The link is configured as a point-to-point link. • F – The link is not configured as a point-to-point link. This is the default.
Edge port	Indicates if the port is configured as an operational Edge port: • T – The port is configured as an Edge port. • F – The port is not configured as an Edge port. This is the default.
Role	The current role of the port: • Root • Designated • Alternate • Backup • Disabled Refer to “Bridges and Bridge Port Roles” on page 7-19 for definitions of the roles.

Table 7.8: CLI Display of 802.1W Summary (Continued)

This Field...	Displays...
State	The port's current 802.1W state. A port can have one of the following states: • Forwarding • Discarding • Learning • Disabled Refer to "Bridge Port States" on page 7-22 and "Edge Port and Non-Edge Port States" on page 7-22.
Designated Cost	The best root path cost that this port received, including the best root path cost that it can transmit.
Designated Bridge	The ID of the bridge that sent the best RST BPDU that was received on this port.

To display detailed information about 802-1W, using the following command:

```
FESX424 Router(config)#show 802-1w detail
================================================================================
VLAN 1 - MULTIPLE SPANNING TREE (MSTP - IEEE 802.1W) ACTIVE
================================================================================
BridgeId 800000e080541700, forceVersion 2, txHoldCount 3

Port 1 - Role: ROOT - State: FORWARDING
  PathCost 200000, Priority 128, AdminOperEdge F, AdminPt2PtMac F
  DesignatedPriority - Root: 0x800000e0804c9c00, Bridge: 0x800000e080541700
  ActiveTimers - rrWhile 4 rcvdInfoWhile 4
  MachineStates - PIM: CURRENT, PRT: ROOT_PORT, PST: FORWARDING
  TCM: ACTIVE, PPM: SENDING_STP, PTX: TRANSMIT_IDLE
  Received - RST BPDUs 0, Config BPDUs 1017, TCN BPDUs 0


Port 2 - Role: DESIGNATED - State: FORWARDING
  PathCost 200000, Priority 128, AdminOperEdge F, AdminPt2PtMac F
  DesignatedPriority - Root: 0x800000e0804c9c00, Bridge: 0x800000e080541700
  ActiveTimers - helloWhen 0
  MachineStates - PIM: CURRENT, PRT: DESIGNATED_PORT, PST: FORWARDING
  TCM: ACTIVE, PPM: SENDING_RSTP, PTX: TRANSMIT_IDLE
  Received - RST BPDUs 0, Config BPDUs 0, TCN BPDUs 0
```

Syntax: show 802-1w detail [vlan <vlan-id>]

The **vlan** <vlan-id> parameter displays 802.1W information for the specified port-based VLAN.

The **show spanning-tree 802.1W** command shows the following information.

This Field...	Displays...
VLAN ID	ID of the VLAN that owns the instance of 802.1W and whether or not it is active.

This Field...	Displays...
Bridge ID	ID of the bridge.
forceVersion	the configured version of the bridge: • 0 – The bridge has been forced to operate in an STP compatible mode. • 2 – The bridge has been forced to operate in an 802.1W mode.
txHoldCount	The number of BPDUs that can be transmitted per Hello Interval. The default is 3.
Port	ID of the port in slot#/port# format.
Role	The current role of the port: • Root • Designated • Alternate • Backup • Disabled Refer to “Bridges and Bridge Port Roles” on page 7-19 for definitions of the roles.
State	The port’s current 802.1W state. A port can have one of the following states: • Forwarding • Discarding • Learning • Disabled Refer to “Bridge Port States” on page 7-22 and “Edge Port and Non-Edge Port States” on page 7-22.
Path Cost	The configured path cost on a link connected to this port.
Priority	The configured priority of the port. The default is 128 or 0x80.
AdminOperEdge	Indicates if the port is an operational Edge port. Edge ports may either be auto-detected or configured (forced) to be Edge ports using the CLI: • T – The port is and Edge port. • F – The port is not an Edge port. This is the default.
AdminP2PMac	Indicates if the point-to-point-mac parameter is configured to be a point-to-point link: • T – The link is a point-to-point link • F – The link is not a point-to-point link. This is the default.

This Field...	Displays...
DesignatedPriority	Shows the following: • Root – Shows the ID of the root bridge for this bridge. • Bridge – Shows the ID of the Designated bridge that is associated with this port.
ActiveTimers	Shows what timers are currently active on this port and the number of seconds they have before they expire: • rrWhile – Recent root timer. A non-zero value means that the port has recently been a Root port. • rcvdInfoWhile – Received information timer. Shows the time remaining before the information held by this port expires (ages out). This timer is initialized with the effective age parameter. (See "Max Age" on page 7-48.) • rbWhile – Recent backup timer. A non-zero value means that the port has recently been a Backup port. • helloWhen – Hello period timer. The value shown is the amount of time between hello messages. • tcWhile – Topology change timer. The value shown is the interval when topology change notices can be propagated on this port. • fdWhile – Forward delay timer. (See the explanation for Fwd Dly on page 49.) • mdelayWhile – Migration delay timer. The amount of time that a bridge on the same LAN has to synchronize its migration state with this port before another BPDU type can cause this port to change the BPDU that it transmits.
Machine States	The current states of the various state machines on the port: • PIM – State of the Port Information state machine. • PRT – State of the Port Role Transition state machine. • PST – State of the Port State Transition state machine. • TCM – State of the Topology Change state machine. • PPM – State of the Port Protocol Migration. • PTX – State of the Port Transmit state machine. Refer to the section "State Machines" on page 7-23 for details on state machines.
Received	Shows the number of BPDU types the port has received: • RST BPDU – BPDU in 802.1W format. • Config BPDU – Legacy configuration BPDU (802.1D format). • TCN BPDU – Legacy topology change BPDU (802.1D format).

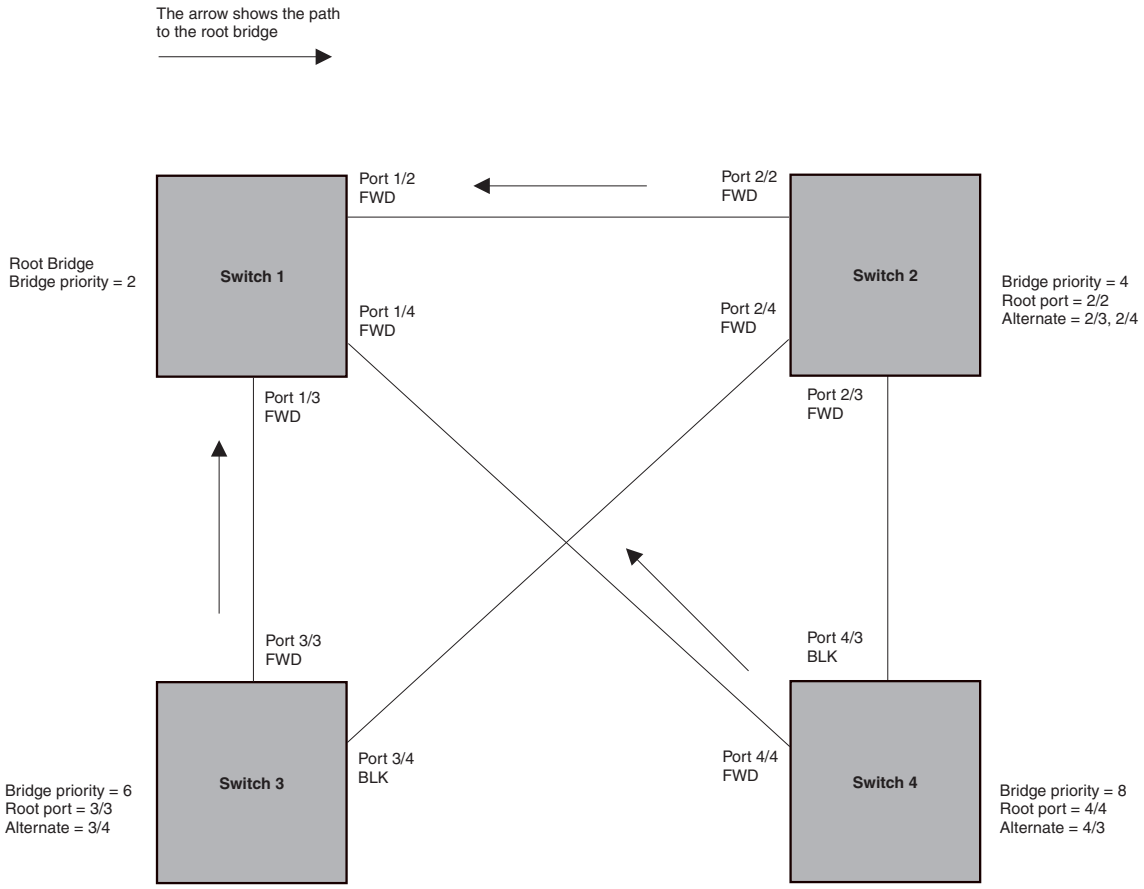
802.1W Draft 3

As an alternative to full 802.1W, you can configure 802.1W Draft 3. 802.1W Draft 3 provides a subset of the RSTP capabilities described in the 802.1W STP specification.

802.1W Draft 3 support is disabled by default. When the feature is enabled, if a root port on a Foundry device that is not the root bridge becomes unavailable, the device can automatically Switch over to an alternate root port, without reconvergence delays. 802.1W Draft 3 does not apply to the root bridge, since all the root bridge's ports are always in the forwarding state.

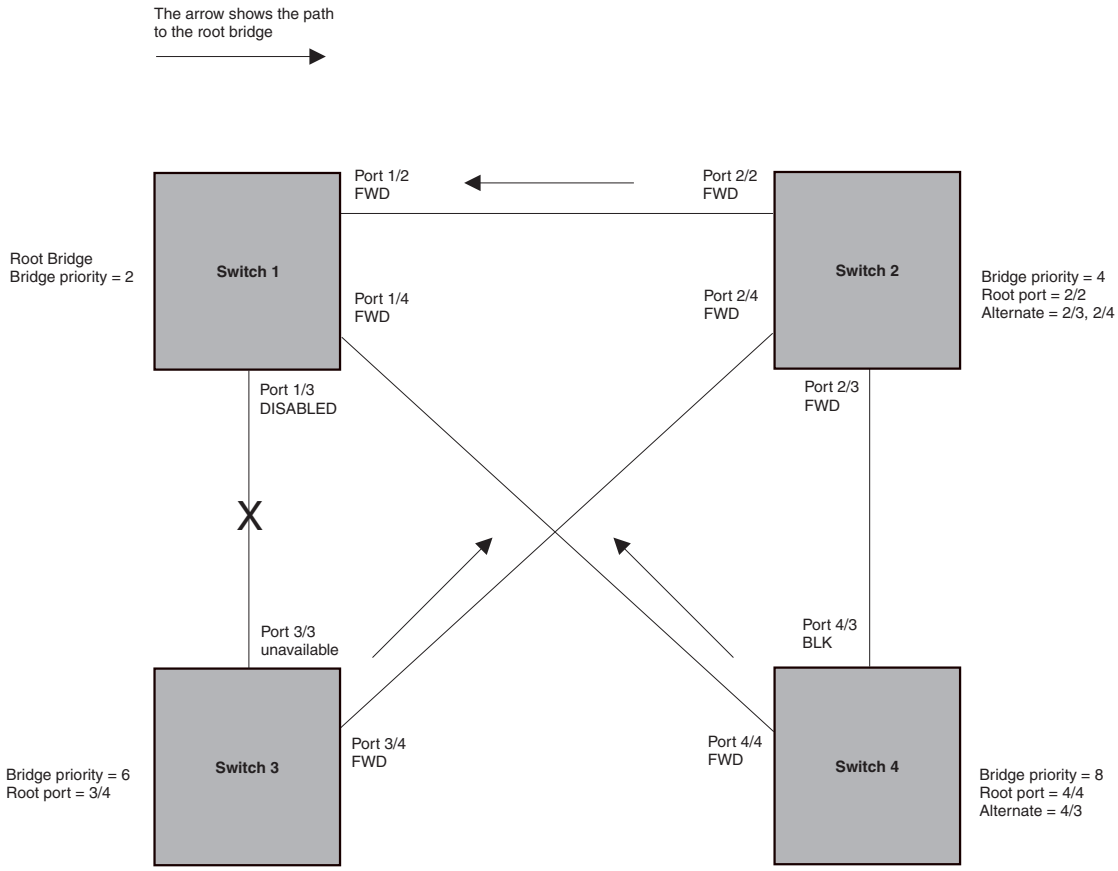
Figure 7.24 shows an example of an optimal STP topology. In this topology, all the non-root bridges have at least two paths to the root bridge (Switch 1 in this example). One of the paths is through the root port. The other path is a backup and is through the alternate port. While the root port is in the forwarding state, the alternate port is in the blocking state.

Figure 7.24 802.1W Draft 3 RSTP ready for failover



If the root port on a Switch becomes unavailable, 802.1W Draft 3 immediately fails over to the alternate port, as shown in Figure 7.25.

Figure 7.25 802.1W Draft 3 RSTP failover to alternate root port



In this example, port 3/3 on Switch 3 has become unavailable. In standard STP (802.1D), if the root port becomes unavailable, the Switch must go through the listening and learning stages on the alternate port to reconverge with the spanning tree. Thus, port 3/4 must go through the listening and learning states before entering the forwarding state and thus reconverging with the spanning tree.

802.1W Draft 3 avoids the reconvergence delay by calculating an alternate root port, and immediately failing over to the alternate port if the root port becomes unavailable. The alternate port is in the blocking state as long as the root port is in the forwarding state, but moves immediately to the active state if the root port becomes unavailable. Thus, using 802.1W Draft 3, Switch 3 immediately fails over to port 3/4, without the delays caused by the listening and learning states.

802.1W Draft 3 selects the port with the next-best cost to the root bridge. For example, on Switch 3, port 3/3 has the best cost to the root bridge and thus is selected by STP as the root port. Port 3/4 has the next-best cost to the root bridge, and thus is selected by 802.1W Draft 3 as the alternate path to the root bridge.

Once a failover occurs, the Switch no longer has an alternate root port. If the port that was an alternate port but became the root port fails, standard STP is used to reconverge with the network. You can minimize the reconvergence delay in this case by setting the forwarding delay on the root bridge to a lower value. For example, if the forwarding delay is set to 15 seconds (the default), change the forwarding delay to a value from 3 – 10 seconds.

During failover, 802.1W Draft 3 flushes the MAC addresses leaned on the unavailable root port, selects the alternate port as the new root port, and places that port in the forwarding state. If traffic is flowing in both directions on the new root port, addresses are flushed (moved) in the rest of the spanning tree automatically.

Reconvergence Time

Spanning tree reconvergence using 802.1W Draft 3 can occur within one second.

After the spanning tree reconverges following the topology change, traffic also must reconverge on all the bridges attached to the spanning tree. This is true regardless of whether 802.1W Draft 3 or standard STP is used to reconverge the spanning tree.

Traffic reconvergence happens after the spanning tree reconvergence, and is achieved by flushing the Layer 2 information on the bridges.

- Following 802.1W Draft 3 reconvergence of the spanning tree, traffic reconvergence occurs in the time it takes for the bridge to detect the link changes plus the STP maximum age set on the bridge.
- If standard STP reconvergence occurs instead, traffic reconvergence takes two times the forward delay plus the maximum age.

NOTE: 802.1W Draft 3 does not apply when a failed root port comes back up. In this case, standard STP is used.

Configuration Considerations

802.1W Draft 3 is disabled by default. To ensure optimal performance of the feature before you enable it:

- Configure the bridge priorities so that the root bridge is one that supports 802.1W Draft 3. (Use a Foundry device or third-party device that supports 802.1W Draft 3.)
- Change the forwarding delay on the root bridge to a value lower than the default 15 seconds. Foundry recommends a value from 3 – 10 seconds. The lower forwarding delay helps reduce reconvergence delays in cases where 802.1W Draft 3 is not applicable, such as when a failed root port comes back up.
- Configure the bridge priorities and root port costs so that each device has an active path to the root bridge if its root port becomes unavailable. For example, port 3/4 is connected to port 2/4 on Switch 2, which has the second most favorable bridge priority in the spanning tree.

NOTE: If reconvergence involves changing the state of a root port on a bridge that supports 802.1D STP but not 802.1W Draft 3, then reconvergence still requires the amount of time it takes for the ports on the 802.1D bridge to change state to forwarding (as needed), and receive BPDUs from the root bridge for the new topology.

Enabling 802.1W Draft 3

802.1W Draft 3 is disabled by default. The procedure for enabling the feature differs depending on whether single STP is enabled on the device.

NOTE: STP must be enabled before you can enable 802.1W Draft 3.

Enabling 802.1W Draft 3 When Single STP Is Not Enabled

By default, each port-based VLAN on the device has its own spanning tree. To enable 802.1W Draft 3 in a port-based VLAN, enter commands such as the following:

```
FESX424 Router(config)# vlan 10
FESX424 Router(config-vlan-10)# spanning-tree rstp
```

Syntax: [no] spanning-tree rstp

This command enables 802.1W Draft 3. You must enter the command separately in each port-based VLAN in which you want to run 802.1W Draft 3.

NOTE: This command does not also enable STP. To enable STP, first enter the **spanning-tree** command without the **rstp** parameter. After you enable STP, enter the **spanning-tree rstp** command to enable 802.1W Draft 3.

To disable 802.1W Draft 3, enter the following command:

```
FESX424 Router(config-vlan-10)# no spanning-tree rstp
```

Enabling 802.1W Draft 3 When Single STP Is Enabled

To enable 802.1W Draft 3 on a device that is running single STP, enter the following command at the global CONFIG level of the CLI:

```
FESX424 Router(config)# spanning-tree single rstp
```

Syntax: [no] spanning-tree single rstp

This command enables 802.1W Draft 3 on the whole device.

NOTE: This command does not also enable single STP. To enable single STP, first enter the **spanning-tree single** command without the **rstp** parameter. After you enable single STP, enter the **spanning-tree single rstp** command to enable 802.1W Draft 3.

To disable 802.1W Draft 3 on a device that is running single STP, enter the following command:

```
FESX424 Router(config)# no spanning-tree single rstp
```

Single Spanning Tree (SSTP)

By default, each port-based VLAN on a Foundry device runs a separate spanning tree, which you can enable or disable on an individual VLAN basis.

Alternatively, you can configure a Foundry device to run a single spanning tree across all ports and VLANs on the device. The Single STP feature (SSTP) is especially useful for connecting a Foundry device to third-party devices that run a single spanning tree in accordance with the 802.1Q specification.

SSTP uses the same parameters, with the same value ranges and defaults, as the default STP support on Foundry devices. See “STP Parameters and Defaults” on page 7-2.

SSTP Defaults

SSTP is disabled by default. When you enable the feature, all VLANs on which STP is enabled become members of a single spanning tree. All VLANs on which STP is disabled are excluded from the single spanning tree.

- To add a VLAN to the single spanning tree, enable STP on that VLAN.
- To remove a VLAN from the single spanning tree, disable STP on that VLAN.

When you enable SSTP, all the ports that are in port-based VLANs with STP enabled become members of a single spanning tree domain. Thus, the ports share a single BPDU broadcast domain. The Foundry device places all the ports in a non-configurable VLAN, 4094, to implement the SSTP domain. However, this VLAN does not affect port membership in the port-based VLANs you have configured. Other broadcast traffic is still contained within the individual port-based VLANs. Therefore, you can use SSTP while still using your existing VLAN configurations without changing your network. In addition, SSTP does not affect 802.1Q tagging. Tagged and untagged ports alike can be members of the single spanning tree domain.

NOTE: When SSTP is enabled, the BPDUs on tagged ports go out untagged.

If you disable SSTP, all VLANs that were members of the single spanning tree run MSTP instead. In MSTP, each VLAN has its own spanning tree. VLANs that were not members of the single spanning tree were not enabled for STP. Therefore, STP remains disabled on those VLANs.

Enabling SSTP

To enable SSTP, use one of the following methods.

NOTE: If the device has only one port-based VLAN (the default VLAN), then the device is already running a single instance of STP. In this case, you do not need to enable SSTP. You need to enable SSTP only if the device contains more than one port-based VLAN and you want all the ports to be in the same STP broadcast domain.

To configure the Foundry device to run a single spanning tree, enter the following command at the global CONFIG level.

```
FESX424 Router(config)# spanning-tree single
```

NOTE: If the device has only one port-based VLAN, the CLI command for enabling SSTP is not listed in the CLI. The command is listed only if you have configured a port-based VLAN.

To change a global STP parameter, enter a command such as the following at the global CONFIG level:

```
FESX424 Router(config) spanning-tree single priority 2
```

This command changes the STP priority for all ports to 2.

To change an STP parameter for a specific port, enter commands such as the following:

```
FESX424 Router(config) spanning-tree single ethernet 1 priority 10
```

The commands shown above override the global setting for STP priority and set the priority to 10 for port 1/1.

Here is the syntax for the global STP parameters.

Syntax: [no] spanning-tree single [forward-delay <value>]
[hello-time <value>] | [maximum-age <time>] | [priority <value>]

Here is the syntax for the STP port parameters.

Syntax: [no] spanning-tree single [ethernet [<slotnum>/]<portnum> path-cost <value> | priority <value>]

NOTE: Both commands listed above are entered at the global CONFIG level.

Displaying SSTP information

To verify that SSTP is in effect, enter the following commands at any level of the CLI:

```
FESX424 Router(config)# show span
```

Syntax: show span [vlan <vlan-id>] | [pvst-mode] | [<num>] |
[detail [vlan <vlan-id> [ethernet [<slotnum>/]<portnum>] | <num>]]

The **vlan** <vlan-id> parameter displays STP information for the specified port-based VLAN.

The **pvst-mode** parameter displays STP information for the device's Per VLAN Spanning Tree (PVST+) compatibility configuration. See "PVST/PVST+ Compatibility" on page 7-61.

The <num> parameter displays only the entries after the number you specify. For example, on a device with three port-based VLANs, if you enter 1, then information for the second and third VLANs is displayed, but information for the first VLAN is not displayed. Information is displayed according to VLAN number, in ascending order. The entry number is not the same as the VLAN number. For example, if you have port-based VLANs 1, 10, and 2024, then the command output has three STP entries. To display information for VLANs 10 and 2024 only, enter **show span 1**.

The **detail** parameter and its additional optional parameters display detailed information for individual ports. See "Displaying Detailed STP Information for Each Interface" on page 7-12.

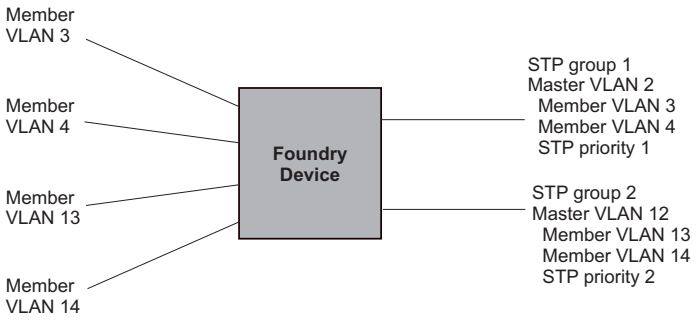
STP per VLAN Group

STP per VLAN group is an STP enhancement that provides scalability while overcoming the limitations of the following scalability alternatives:

- Standard STP – You can configure only 128 instances of standard STP on a Foundry device. It is possible to need more instances of STP than this in large configurations. Using STP per VLAN group, you can aggregate STP instances.
- Single STP – Single STP allows all the VLANs to run STP, but each VLAN runs the same instance of STP, resulting in numerous blocked ports that do not pass any Layer 2 traffic. STP per VLAN group uses all available links by load balancing traffic for different instances of STP on different ports. A port that blocks traffic for one spanning tree forwards traffic for another spanning tree.

STP per VLAN group allows you to group VLANs and apply the same STP parameter settings to all the VLANs in the group. Figure 7.26 shows an example of a STP per VLAN group implementation.

Figure 7.26 STP per VLAN Group Example



A master VLAN contains one or more member VLANs. Each of the member VLANs in the STP Group runs the same instance of STP and uses the STP parameters configured for the master VLAN. In this example, the Foundry device is configured with VLANs 3, 4, 13, and 14. VLANs 3 and 4 are grouped in master VLAN 2, which is in STP group 1. VLANs 13 and 14 are grouped in master VLAN 12, which is in STP group 2. The VLANs in STP group 1 all share the same spanning tree. The VLANs in STP group 2 share a different spanning tree.

All the ports in the VLANs are tagged. The ports must be tagged so that they can be in both a member VLAN and the member's master VLAN. For example, ports 1/1 – 1/4 are in member VLAN 3 and also in master VLAN 2 (since master VLAN 2 contains member VLAN 3).

STP Load Balancing

Notice that the STP groups each have different STP priorities. In configurations that use the STP groups on multiple devices, you can use the STP priorities to load balance the STP traffic. By setting the STP priorities for the same STP group to different values on each device, you can cause each of the devices to be the root bridge for a different STP group. This type of configuration distributes the traffic evenly across the devices and also ensures that ports that are blocked in one STP group's spanning tree are used by another STP group's spanning tree for forwarding. See "Configuration Example for STP Load Sharing" on page 7-60 for an example using STP load sharing.

Configuring STP per VLAN Group

To configure STP per VLAN group:

- Configure the member VLANs.
- Optionally, configure master VLANs to contain the member VLANs. This is useful when you have a lot of member VLANs and you do not want to individually configure STP on each one. Each of the member VLANs in the STP group uses the STP settings of the master VLAN.
- Configure the STP groups. Each STP group runs a separate instance of STP.

Here are the CLI commands for implementing the STP per VLAN group configuration shown in Figure 7.26. The following commands configure the member VLANs (3, 4, 13, and 14) and the master VLANs (2 and 12). Notice that changes to STP parameters are made in the master VLANs only, not in the member VLANs.

```
FastIron SuperX Router(config)# vlan 2
FastIron SuperX Router(config-vlan-2)# spanning-tree priority 1
FastIron SuperX Router(config-vlan-2)# tagged ethernet 1/1 to 1/4
FastIron SuperX Router(config-vlan-2)# vlan 3
FastIron SuperX Router(config-vlan-3)# tagged ethernet 1/1 to 1/4
FastIron SuperX Router(config-vlan-3)# vlan 4
FastIron SuperX Router(config-vlan-4)# tagged ethernet 1/1 to 1/4
FastIron SuperX Router(config-vlan-4)# vlan 12
FastIron SuperX Router(config-vlan-12)# spanning-tree priority 2
FastIron SuperX Router(config-vlan-12)# tagged ethernet 1/1 to 1/4
FastIron SuperX Router(config-vlan-12)# vlan 13
FastIron SuperX Router(config-vlan-13)# tagged ethernet 1/1 to 1/4
FastIron SuperX Router(config-vlan-13)# vlan 14
FastIron SuperX Router(config-vlan-14)# tagged ethernet 1/1 to 1/4
FastIron SuperX Router(config-vlan-14)# exit
```

The following commands configure the STP groups.

```
FastIron SuperX Router(config)# stp-group 1
FastIron SuperX Router(config-stp-group-1)# master-vlan 2
FastIron SuperX Router(config-stp-group-1)# member-vlan 3 to 4
FastIron SuperX Router(config-stp-group-1)# exit
FastIron SuperX Router(config)# stp-group 2
FastIron SuperX Router(config-stp-group-2)# master-vlan 12
FastIron SuperX Router(config-stp-group-2)# member-vlan 13 to 14
```

Syntax: [no] stp-group <num>

This command changes the CLI to the STP group configuration level. The following commands are valid at this level. The <num> parameter specifies the STP group ID and can be from 1 – 32.

Syntax: [no] master-vlan <num>

This command adds a master VLAN to the STP group. The master VLAN contains the STP settings for all the VLANs in the STP per VLAN group. The <num> parameter specifies the VLAN ID. An STP group can contain one master VLAN.

NOTE: If you delete the master VLAN from an STP group, the software automatically assigns the first member VLAN in the group to be the new master VLAN for the group.

Syntax: [no] member-vlan <num> [to <num>]

This command adds additional VLANs to the STP group. These VLANs also inherit the STP settings of the master VLAN in the group.

Syntax: [no] member-group <num>

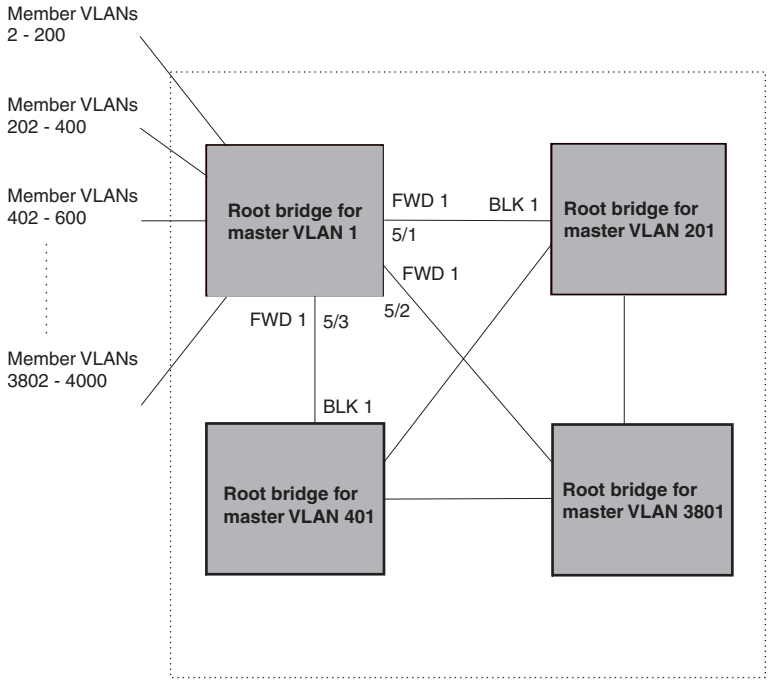
This command adds a member group (a VLAN group) to the STP group. All the VLANs in the member group inherit the STP settings of the master VLAN in the group. The <num> parameter specifies the VLAN group ID.

NOTE: This command is optional and is not used in the example above. For an example of this command, see "Configuration Example for STP Load Sharing".

Configuration Example for STP Load Sharing

Figure 7.27 shows another example of a STP per VLAN group implementation.

Figure 7.27 More Complex STP per VLAN Group Example



In this example, each of the devices in the core is configured with a common set of master VLANs, each of which contains one or more member VLANs. Each of the member VLANs in an STP group runs the same instance of STP and uses the STP parameters configured for the master VLAN.

The STP group ID identifies the STP instance. All VLANs within an STP group run the same instance of STP. The master VLAN specifies the bridge STP parameters for the STP group, including the bridge priority. In this example, each of the devices in the core is configured to be the default root bridge for a different master VLAN. This configuration ensures that each link can be used for forwarding some traffic. For example, all the ports on the root bridge for master VLAN 1 are configured to forward BPDUs for master VLAN's spanning tree. Ports on the other devices block or forward VLAN 1's traffic based on STP convergence. All the ports on the root bridge for VLAN 2 forward VLAN 2's traffic, and so on.

All the ports in the VLANs are tagged. The ports must be tagged so that they can be in both a member VLAN and the member's master VLAN. For example, port 1/1 – and ports 5/1, 5/2, and 5/3 are in member VLAN 2 and master VLAN 1 (since master VLAN a contains member VLAN 2).

Here are the commands for configuring the root bridge for master VLAN 1 in figure Figure 7.26 for STP per VLAN group. The first group of commands configures the master VLANs. Notice that the STP priority is set to a different value for each VLAN. In addition, the same VLAN has a different STP priority on each device. This provides load balancing by making each of the devices a root bridge for a different spanning tree.

```
FastIron SuperX Router(config)# vlan 1
FastIron SuperX Router(config-vlan-1)# spanning-tree priority 1
FastIron SuperX Router(config-vlan-1)# tag ethernet 1/1 ethernet 5/1 to 5/3
FastIron SuperX Router(config-vlan-1)# vlan 201
FastIron SuperX Router(config-vlan-201)# spanning-tree priority 2
FastIron SuperX Router(config-vlan-201)# tag ethernet 1/2 ethernet 5/1 to 5/3
FastIron SuperX Router(config-vlan-201)# vlan 401
FastIron SuperX Router(config-vlan-401)# spanning-tree priority 3
FastIron SuperX Router(config-vlan-401)# tag ethernet 1/3 ethernet 5/1 to 5/3
...
```



```
FastIron SuperX Router(config-vlan-3601)# vlan 3801
FastIron SuperX Router(config-vlan-3801)# spanning-tree priority 20
FastIron SuperX Router(config-vlan-3801)# tag ethernet 1/20 ethernet 5/1 to 5/3
FastIron SuperX Router(config-vlan-3801)# exit
```

The next group of commands configures VLAN groups for the member VLANs. Notice that the VLAN groups do not contain the VLAN numbers assigned to the master VLANs. Also notice that no STP parameters are configured for the groups of member VLANs. Each group of member VLANs will inherit its STP settings from its master VLAN.

Set the bridge priority for each master VLAN to the highest priority (1) on one of the devices in the STP per VLAN group configuration. By setting the bridge priority to the highest priority, you make the device the default root bridge for the spanning tree. To ensure STP load balancing, make each of the devices the default root bridge for a different master VLAN.

```
FastIron SuperX Router(config)# vlan-group 1 vlan 2 to 200
FastIron SuperX Router(config-vlan-group-1)# tag ethernet 1/1 ethernet 5/1 to 5/3
FastIron SuperX Router(config-vlan-group-1)# vlan-group 2 vlan 202 to 400
FastIron SuperX Router(config-vlan-group-2)# tag ethernet 1/2 ethernet 5/1 to 5/3
FastIron SuperX Router(config-vlan-group-2)# vlan-group 3 vlan 402 to 600
FastIron SuperX Router(config-vlan-group-2)# tag ethernet 1/3 ethernet 5/1 to 5/3
...
FastIron SuperX Router(config-vlan-group-19)# vlan-group 20 vlan 3082 to 4000
FastIron SuperX Router(config-vlan-group-20)# tag ethernet 1/20 ethernet 5/1 to 5/3
FastIron SuperX Router(config-vlan-group-20)# exit
```

The following group of commands configures the STP groups. Each STP group in this configuration contains one master VLAN, which contains a VLAN group. This example shows that an STP group also can contain additional VLANs (VLANs not configured in a VLAN group).

```
FastIron SuperX Router(config)# stp-group 1
FastIron SuperX Router(config-stp-group-1)# master-vlan 1
FastIron SuperX Router(config-stp-group-1)# member-group 1
FastIron SuperX Router(config-stp-group-1)# member-vlan 4001 4004 to 4010
FastIron SuperX Router(config-stp-group-1)# stp-group 2
FastIron SuperX Router(config-stp-group-2)# master-vlan 201
FastIron SuperX Router(config-stp-group-2)# member-group 2
FastIron SuperX Router(config-stp-group-2)# member-vlan 4002 4003 4011 to 4015
FastIron SuperX Router(config-stp-group-2)# stp-group 3
FastIron SuperX Router(config-stp-group-3)# master-vlan 401
FastIron SuperX Router(config-stp-group-3)# member-group 3
...
FastIron SuperX Router(config-stp-group-19)# stp-group 20
FastIron SuperX Router(config-stp-group-20)# master-vlan 3081
FastIron SuperX Router(config-stp-group-20)# member-group 20
```

PVST/PVST+ Compatibility

The FastIron family of switches provide support for Cisco's Per VLAN Spanning Tree plus (PVST+), by allowing the device to run multiple spanning trees (MSTP) while also interoperating with IEEE 802.1Q devices[1].

NOTE: Foundry ports automatically detect PVST+ BPDUs and enable support for the BPDUs once detected. You do not need to perform any configuration steps to enable PVST+ support. However, to support the IEEE 802.1Q BPDUs, you might need to enable dual-mode support.

1.Cisco user documentation for PVST/PVST+ refers to the IEEE 802.1Q spanning tree as the ***Common Spanning Tree (CST)***.

Foundry's support for Cisco's Per VLAN Spanning Tree plus (PVST+), allows a Foundry device to run multiple spanning trees (MSTP) while also interoperating with IEEE 802.1Q devices. Foundry ports automatically detect PVST+ BPDUs and enable support for the BPDUs once detected. The enhancement allows a port that is in PVST+ compatibility mode due to auto-detection to revert to the default MSTP mode when one of the following events occurs:

- The link is disconnected or broken
- The link is administratively disabled
- The link is disabled by interaction with the link-keepalive protocol

This enhancement allows a port that was originally interoperating with PVST+ to revert to MSTP when connected to a Foundry device.

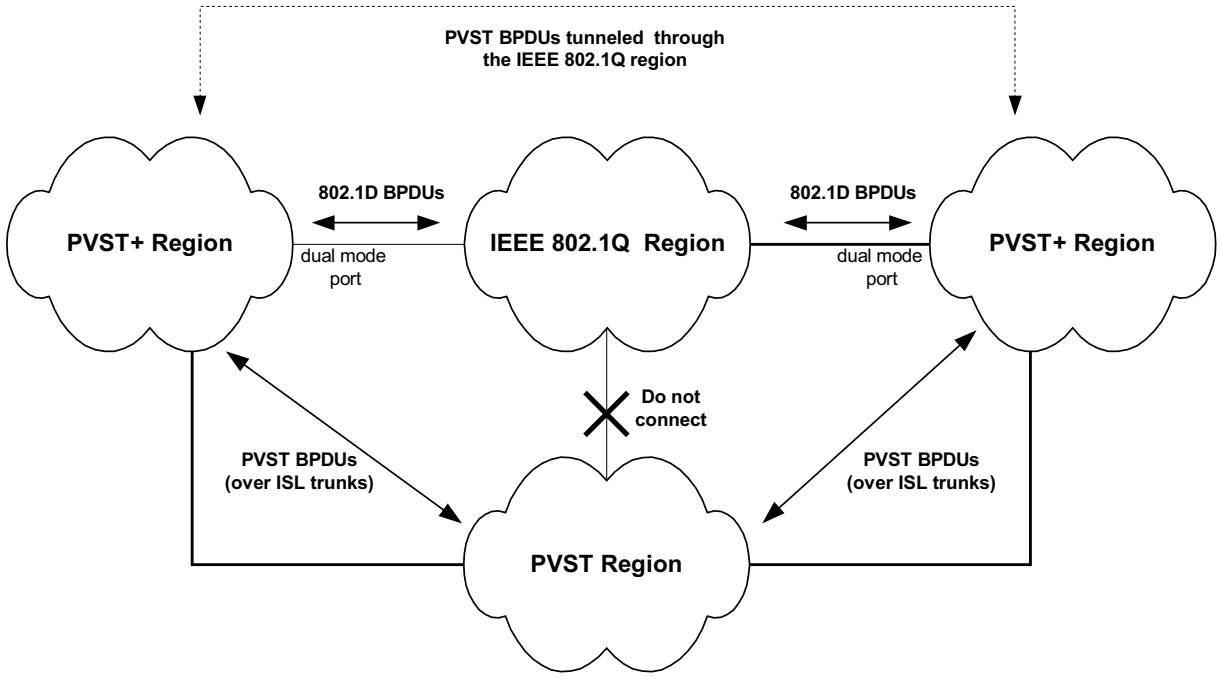
Overview of PVST and PVST+

Per VLAN Spanning Tree (PVST) is a Cisco proprietary protocol that allows a Cisco device to have multiple spanning trees. The Cisco device can interoperate with spanning trees on other PVST devices but cannot interoperate with IEEE 802.1Q devices. An IEEE 802.1Q device has all its ports running a single spanning tree. ***PVST+*** is an extension of PVST that allows a Cisco device to also interoperate with devices that are running a single spanning tree (IEEE 802.1Q).

The enhanced PVST+ support 1allows a Foundry device to interoperate with PVST spanning trees and the IEEE 802.1Q spanning tree at the same time.

IEEE 802.1Q and PVST regions cannot interoperate directly but can interoperate indirectly through PVST+ regions. PVST BPDUs are tunnelled through 802.1Q regions, while PVST BPDUs for VLAN 1 (the IEEE 802.1Q VLAN) are processed by PVST+ regions. Figure 7.28 shows the interaction of IEEE 802.1Q, PVST, and PVST+ regions.

Figure 7.28 Interaction of IEEE 802.1Q, PVST, and PVST+ regions



VLAN Tags and Dual Mode

To support the IEEE 802.1Q (Common Spanning Tree) portion of PVST+, a port must be a member of VLAN 1. Cisco devices always use VLAN 1 to support the IEEE 802.1Q portion of PVST+.

For the port to also support the other VLANs (the PVST+ VLANs) in tagged mode, the dual-mode feature must be enabled on the port. The ***dual-mode*** feature enables the port to send and receive both tagged and untagged frames. When the dual-mode feature is enabled, the port is an untagged member of one of its VLANs and is at the same time a tagged member of all its other VLANs.

The untagged frames are supported on the port's ***Port Native VLAN***. By default, the Port Native VLAN is the same as the device's ***Default VLAN***[1], which by default is VLAN 1. Thus, to support IEEE 802.1Q in a typical configuration, the port must be able to send and receive untagged frames for VLAN 1 and tagged frames for the other VLANs.

If you want to use tagged frames on VLAN 1, you can change the default VLAN ID to an ID other than 1. You also can specify the VLAN on which you want the port to send and receive untagged frames (the Port Native VLAN). The Port Native VLAN ID does not need to be the same as the Default VLAN.

NOTE: Support for the IEEE 802.1Q spanning tree always uses VLAN 1, regardless of whether the devices are configured to use tagged or untagged frames on the VLAN.

Configuring PVST+ Support

PVST+ support is automatically enabled when the port receives a PVST BPDU. You can manually enable the support at any time or disable the support if desired.

If you want a tagged port to also support IEEE 802.1Q BPDUs, you need to enable the dual-mode feature on the port. The dual-mode feature is disabled by default and must be enabled manually.

A port that is in PVST+ compatibility mode due to auto-detection reverts to the default MSTP mode when one of the following events occurs:

- The link is disconnected or broken
- The link is administratively disabled
- The link is disabled by interaction with the link-keepalive protocol

This allows a port that was originally interoperating with PVST+ to revert to MSTP when connected to a Foundry device.

Enabling PVST+ Support Manually

To immediately enable PVST+ support on a port, enter commands such as the following:

```
FastIron SuperX Router(config)# interface ethernet 1/1
FastIron SuperX Router(config-if-1/1)# pvst-mode
```

Syntax: [no] pvst-mode

NOTE: If you disable PVST+ support, the software still automatically enables PVST+ support if the port receives a BPDU with PVST+ format.

NOTE: If 802.1W and pvst-mode (either by auto-detection or by explicit configuration) are enabled on a tagged VLAN port, 802.1W will treat the PVST BPDUs as legacy 802.1D BPDUs.

Enabling Dual-Mode Support

To enable the dual-mode feature on a port, enter the following command at the interface configuration level for the port:

```
FastIron SuperX Router(config-if-1/1)# dual-mode
```

1.Cisco PVST/PVST+ documentation refers to the Default VLAN as the ***Default Native VLAN***.

Syntax: [no] dual-mode [<vlan-id>]

The <vlan-id> specifies the port's Port Native VLAN. This is the VLAN on which the port will support untagged frames. By default, the Port Native VLAN is the same as the default VLAN (which is VLAN 1 by default).

For more information about the dual-mode feature, see "Dual-Mode VLAN Ports" on page 11-56.

Displaying PVST+ Support Information

To display PVST+ information for ports on a Foundry device, enter the following command at any level of the CLI:

```
FastIron SuperX Router(config)# show span pvst-mode
PVST+ Enabled on:
Port      Method
1/1       Set by configuration
1/2       Set by configuration
2/10      Set by auto-detect
3/12      Set by configuration
4/24      Set by auto-detect
```

Syntax: show span pvst-mode

This command displays the following information.

Table 7.9: CLI Display of PVST+ Information

This Field...	Displays...
Port	The Foundry port number. **Note**: The command lists information only for the ports on which PVST+ support is enabled.
Method	The method by which PVST+ support was enabled on the port. The method can be one of the following: • Set by configuration – You enabled the support. • Set by auto-detect – The support was enabled automatically when the port received a PVST+ BPDU.

Configuration Examples

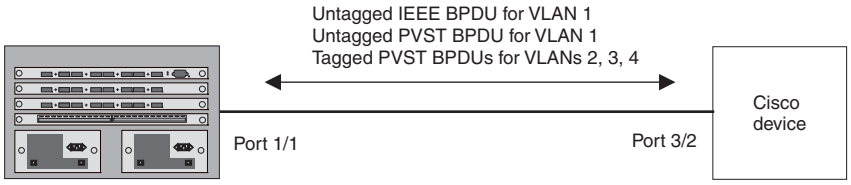
The following examples show configuration examples for two common configurations:

- Untagged IEEE 802.1Q BPDUs on VLAN 1 and tagged PVST+ BPDUs on other VLANs
- Tagged IEEE 802.1Q BPDUs on VLAN 1 and untagged BPDUs on another VLAN

Tagged Port Using Default VLAN 1 as its Port Native VLAN

Figure 7.29 shows an example of a PVST+ configuration that uses VLAN 1 as the untagged default VLAN and VLANs 2, 3, and 4 as tagged VLANs.

Figure 7.29 Default VLAN 1 for untagged BPDUs



To implement this configuration, enter the following commands.

Commands on the Foundry Device

```
FastIron SuperX Router(config)# vlan-group 1 vlan 2 to 4
FastIron SuperX Router(config-vlan-group-1)# tagged ethernet 1/1
FastIron SuperX Router(config-vlan-group-1)# exit
FastIron SuperX Router(config)# interface ethernet 1/1
FastIron SuperX Router(config-if-1/1)# dual-mode
FastIron SuperX Router(config-if-1/1)# pvst-mode
```

These commands configure a VLAN group containing VLANs 2, 3, and 4, add port 1/1 as a tagged port to the VLANs, and enable the dual-mode feature and PVST+ support on the port. The dual-mode feature allows the port to send and receive untagged frames for the default VLAN (VLAN 1 in this case) in addition to tagged frames for VLANs 2, 3, and 4. Enabling the PVST+ support ensures that the port is ready to send and receive PVST+ BPDUs. If you do not manually enable PVST+ support, the support is not enabled until the port receives a PVST+ BPDU.

The configuration leaves the default VLAN and the port's Port Native VLAN unchanged. The default VLAN is 1 and the port's Port Native VLAN also is 1. The dual-mode feature supports untagged frames on the default VLAN only. Thus, port 1/1 can send and receive untagged BPDUs for VLAN 1 and can send and receive tagged BPDUs for the other VLANs.

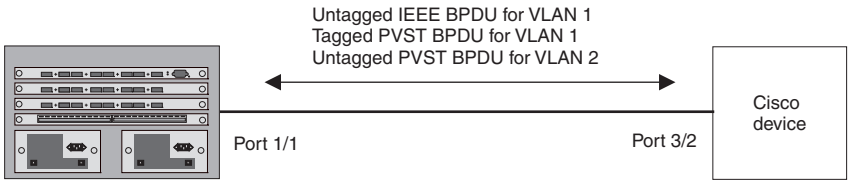
Port 1/1 will process BPDUs as follows:

- Process IEEE 802.1Q BPDUs for VLAN 1.
- Process tagged PVST BPDUs for VLANs 2, 3, and 4.
- Drop untagged PVST BPDUs for VLAN 1.

Untagged Port Using VLAN 2 as Port Native VLAN

Figure 7.30 shows an example in which a port's Port Native VLAN is not VLAN 1. In this case, VLAN 1 uses tagged frames and VLAN 2 uses untagged frames.

Figure 7.30 Port Native VLAN 2 for untagged BPDUs



To implement this configuration, enter the following commands.

Commands on the Foundry Device

```
FastIron SuperX Router(config)# default-vlan-id 4000
FastIron SuperX Router(config)# vlan 1
FastIron SuperX Router(config-vlan-1)# tagged ethernet 1/1
FastIron SuperX Router(config-vlan-1)# exit
FastIron SuperX Router(config)# vlan 2
```

```
FastIron SuperX Router(config-vlan-2)# tagged ethernet 1/1
FastIron SuperX Router(config-vlan-2)# exit
FastIron SuperX Router(config)# interface ethernet 1/1
FastIron SuperX Router(config-if-1/1)# dual-mode 2
FastIron SuperX Router(config-if-1/1)# pvst-mode
FastIron SuperX Router(config-if-1/1)# exit
```

These commands change the default VLAN ID, configure port 1/1 as a tagged member of VLANs 1 and 2, and enable the dual-mode feature and PVST+ support on port 1/1. Since VLAN 1 is tagged in this configuration, the default VLAN ID must be changed from VLAN 1 to another VLAN ID. Changing the default VLAN ID from 1 allows the port to process tagged frames for VLAN 1. VLAN 2 is specified with the **dual-mode** command, which makes VLAN 2 the port's Port Native VLAN. As a result, the port processes untagged frames and untagged PVST BPDUs on VLAN 2.

NOTE: Although VLAN 2 becomes the port's untagged VLAN, the CLI still requires that you add the port to the VLAN as a tagged port, since the port is a member of more than one VLAN.

Port 1/1 will process BPDUs as follows:

- Process IEEE 802.1Q BPDUs for VLAN 1.
- Process untagged PVST BPDUs for VLAN 2.
- Drop tagged PVST BPDUs for VLAN 1.

Note that when VLAN 1 is not the default VLAN, the ports must have the dual-mode feature enabled in order to process IEEE 802.1Q BPDUs.

For example, the following configuration is incorrect:

```
FastIron SuperX Router(config)# default-vlan-id 1000
FastIron SuperX Router(config)# vlan 1
FastIron SuperX Router(config-vlan-1)# tagged ethernet 1/1 to 1/2
FastIron SuperX Router(config-vlan-1)# exit
FastIron SuperX Router(config)# interface ethernet 1/1
FastIron SuperX Router(config-if-1/1)# pvst-mode
FastIron SuperX Router(config-if-1/1)# exit
FastIron SuperX Router(config)# interface ethernet 1/2
FastIron SuperX Router(config-if-1/2)# pvst-mode
FastIron SuperX Router(config-if-1/2)# exit
```

In the configuration above, all PVST BPDUs associated with VLAN 1 would be discarded. Since IEEE BPDUs associated with VLAN 1 are untagged, they are discarded because the ports in VLAN 1 are tagged. Effectively, the BPDUs are never processed by the Spanning Tree Protocol. STP assumes that there is no better bridge on the network and sets the ports to FORWARDING. This could cause a Layer 2 loop.

The following configuration is correct:

```
FastIron SuperX Router(config)# default-vlan-id 1000
FastIron SuperX Router(config)# vlan 1
FastIron SuperX Router(config-vlan-1)# tagged ethernet 1/1 to 1/2
FastIron SuperX Router(config-vlan-1)# exit
FastIron SuperX Router(config)# interface ethernet 1/1
FastIron SuperX Router(config-if-1/1)# pvst-mode
FastIron SuperX Router(config-if-1/1)# dual-mode
FastIron SuperX Router(config-if-1/1)# exit
FastIron SuperX Router(config)# interface ethernet 1/2
FastIron SuperX Router(config-if-1/2)# pvst-mode
FastIron SuperX Router(config-if-1/2)# dual-mode
FastIron SuperX Router(config-if-1/2)# exit
```

Setting the ports as dual-mode ensures that the untagged IEEE 802.1Q BPDUs reach the VLAN 1 instance.

Chapter 8
Configuring Metro Features

This chapter describes how to configure the Metro features listed in Table 8.1. You can use these metro features individually or in combination to provide fast, reliable, and easy to configure Layer 2 connectivity in your Metro network.

Table 8.1: Chapter Contents

Topology Groups

A topology group is a named set of VLANs that share a Layer 2 topology. Topology groups simplify configuration and enhance scalability of Layer 2 protocols by allowing you to run a single instance of a Layer 2 protocol on multiple VLANs.

You can use topology groups with the following Layer 2 protocols:

- STP
- MRP
- VSRP
- 802.1W

Topology groups simplify Layer 2 configuration and provide scalability by enabling you to use the same instance of a Layer 2 protocol for multiple VLANs. For example, if a Foundry device is deployed in a Metro network and provides forwarding for two MRP rings that each contain 128 VLANs, you can configure a topology group for each

ring. If a link failure in a ring causes a topology change, the change is applied to all the VLANs in the ring's topology group. Without topology groups, you would need to configure a separate ring for each VLAN.

NOTE: If you plan to use a configuration saved under an earlier software release and the configuration contains STP groups, the CLI converts the STP groups into topology groups when you save the configuration. For backward compatibility, you can still use the STP group commands. However, the CLI converts the commands into the topology group syntax. Likewise, the **show stp-group** command displays STP topology groups.

Master VLAN and Member VLANs

Each topology group contains a master VLAN and can contain one or more member VLANs and VLAN groups.

- ***Master VLAN*** – The master VLAN contains the configuration information for the Layer 2 protocol. For example, if you plan to use the topology group for MRP, the topology group's master VLAN contains the ring configuration information.
- ***Member VLANs*** – The member VLANs are additional VLANs that share ports with the master VLAN. The Layer 2 protocol settings for the ports in the master VLAN apply to the same ports in the member VLANs. A change to the master VLAN's Layer 2 protocol configuration or Layer 2 topology affects all the member VLANs. Member VLANs do not independently run a Layer 2 protocol.
- ***Member VLAN groups*** – A VLAN group is a named set of VLANs. The VLANs within a VLAN group have the same ports and use the same values for other VLAN parameters.

When a Layer 2 topology change occurs on a port in the master VLAN, the same change is applied to that port in all the member VLANs that contain the port. For example, if you configure a topology group whose master VLAN contains ports 1/1 and 1/2, a Layer 2 state change on port 1/1 applies to port 1/1 in all the member VLANs that contain that port. However, the state change does not affect port 1/1 in VLANs that are not members of the topology group.

Control Ports and Free Ports

A port that is in a topology group can be a control port or a free port.

- ***Control port*** – A control port is a port in the master VLAN, and is therefore controlled by the Layer 2 protocol configured in the master VLAN. The same port in all the member VLANs is controlled by the master VLAN's Layer 2 protocol. Each member VLAN must contain all of the control ports and can contain additional ports.
- ***Free port*** – A free port is not controlled by the master VLAN's Layer 2 protocol. The master VLAN can contain free ports. (In this case, the Layer 2 protocol is disabled on those ports.) In addition, any ports in the member VLANs that are not also in the master VLAN are free ports.

NOTE: Since free ports are not controlled by the master port's Layer 2 protocol, they are assumed to always be in the Forwarding state.

Configuration Considerations

- Topology groups are supported in all FESX, FSX, and FWSX devices and associated software releases.
- You must configure the master VLAN and member VLANs or member VLAN groups before you configure the topology group.
- You can configure up to 256 topology groups. Each group can control up to 4096 VLANs. A VLAN cannot be controlled by more than one topology group.
- The topology group must contain a master VLAN and can also contain individual member VLANs, VLAN groups, or a combination of individual member VLANs and VLAN groups.
- Once you add a VLAN as a member of a topology group, all the Layer 2 protocol information on the VLAN is deleted.

Configuring a Topology Group

To configure a topology group, enter commands such as the following:

```
FastIron SuperX Router(config)# topology-group 2
FastIron SuperX Router(config-topo-group-2)# master-vlan 2
FastIron SuperX Router(config-topo-group-2)# member-vlan 3
FastIron SuperX Router(config-topo-group-2)# member-vlan 4
FastIron SuperX Router(config-topo-group-2)# member-vlan 5
FastIron SuperX Router(config-topo-group-2)# member-group 2
```

These commands create topology group 2 and add the following:

- Master VLAN 2
- Member VLANs 2, 3, and 4
- Member VLAN group 2

Syntax: [no] topology-group <group-id>

The <group-id> parameter specifies the topology group ID and can be from 1 – 256.

Syntax: [no] master-vlan <vlan-id>

This command adds the master VLAN. The VLAN must already be configured. Make sure all the Layer 2 protocol settings in the VLAN are correct for your configuration before you add the VLAN to the topology group. A topology group can have only one master VLAN.

NOTE: If you remove the master VLAN (by entering **no master-vlan** <vlan-id>), the software selects the next-highest numbered member VLAN as the new master VLAN. For example, if you remove master VLAN 2 from the example above, the CLI converts member VLAN 3 into the new master VLAN. The new master VLAN inherits the Layer 2 protocol settings of the older master VLAN.

NOTE: If you add a new master VLAN to a topology group that already has a master VLAN, the new master VLAN replaces the older master VLAN. All member VLANs and VLAN groups follow the Layer 2 protocol settings of the new master VLAN.

Syntax: [no] member-vlan <vlan-id>

The <vlan-id> parameter specifies a VLAN ID. The VLAN must already be configured.

Syntax: [no] member-group <num>

The <num> specifies a VLAN group ID. The VLAN group must already be configured.

NOTE: Once you add a VLAN or VLAN group as a member of a topology group, all the Layer 2 protocol configuration information for the VLAN or group is deleted. For example, if STP is configured on a VLAN and you add the VLAN to a topology group, the STP configuration is removed from the VLAN. Once you add the VLAN to a topology group, the VLAN uses the Layer 2 protocol settings of the master VLAN.

If you remove a member VLAN or VLAN group from a topology group, you will need to reconfigure the Layer 2 protocol information in the VLAN or VLAN group.

Displaying Topology Group Information

The following sections show how to display STP information and topology group information for VLANS.

Displaying STP Information

To display STP information for a VLAN, enter a command such as the following:

```
FESX424 Router(config)# show span vlan 4
VLAN 4 BPDU cam_index is 14344 and the Master DMA Are(HEX) 18 1A
STP instance owned by VLAN 2
```

This example shows STP information for VLAN 4. The line shown in bold type indicates that the VLAN's STP configuration is controlled by VLAN 2. This information indicates that VLAN 4 is a member of a topology group and VLAN 2 is the master VLAN in that topology group.

Displaying Topology Group Information

To display topology group information, enter the following command:

```
FastIron SuperX Router(config)# show topology-group

Topology Group 3
================
 master-vlan 2
 member-vlan none

 Common control ports           L2 protocol
 ethernet 1/1                   MRP
 ethernet 1/2                   MRP
 ethernet 1/5                   VSRP
 ethernet 2/22                  VSRP
 Per vlan free ports
 ethernet 2/3                    Vlan 2
 ethernet 2/4                    Vlan 2
 ethernet 2/11                   Vlan 2
 ethernet 2/12                   Vlan 2
```

Syntax: show topology-group [<group-id>]

This display shows the following information.

Table 8.2: CLI Display of Topology Group Information

This Field...	Displays...
master-vlan	The master VLAN for the topology group. The settings for STP, MRP, or VSRP on the control ports in the master VLAN apply to all control ports in the member VLANs within the topology group.
member-vlan	The member VLANs in the topology group.
Common control ports	The master VLAN ports that are configured with Layer 2 protocol information. The Layer 2 protocol configuration and state of these ports in the master VLAN applies to the same port numbers in all the member VLANs.
L2 protocol	The Layer 2 protocol configured on the control ports. The Layer 2 protocol can be one of the following: • MRP • STP • VSRP

Table 8.2: CLI Display of Topology Group Information (Continued)

This Field...	Displays...
Per vlan free ports	The ports that are not controlled by the Layer 2 protocol information in the master VLAN.

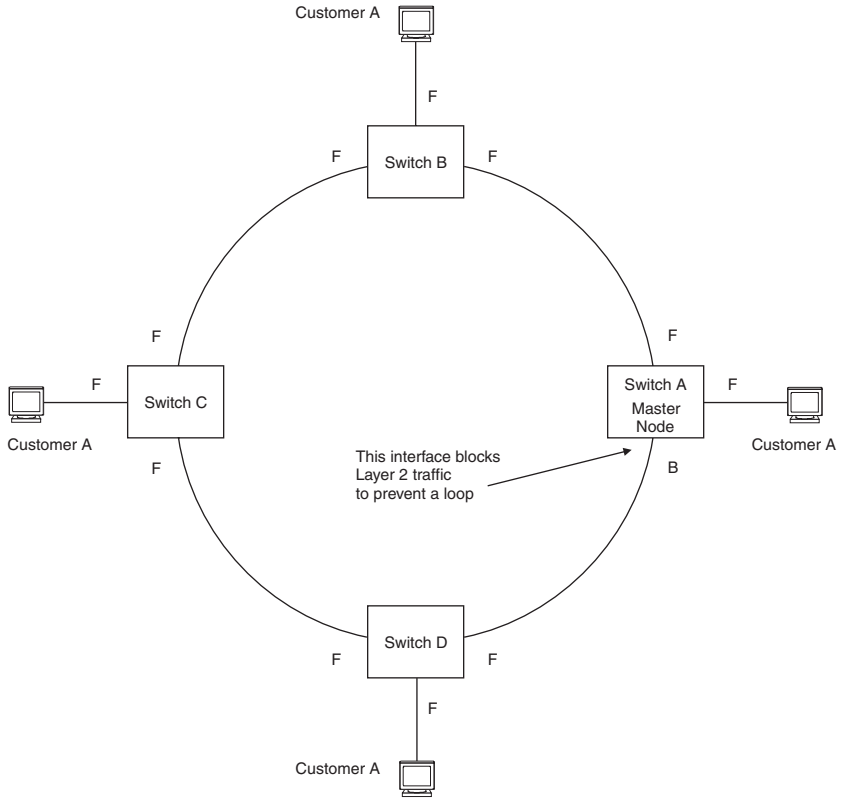
Metro Ring Protocol (MRP)

The Metro Ring Protocol (MRP) is a Foundry proprietary protocol that prevents Layer 2 loops and provides fast reconvergence in Layer 2 ring topologies. It is an alternative to STP and is especially useful in Metropolitan Area Networks (MANs) where using STP has the following drawbacks:

- STP allows a maximum of seven nodes. Metro rings can easily contain more nodes than this.
- STP has a slow reconvergence time, taking many seconds or even minutes. MRP can detect and heal a break in the ring in sub-second time.

Figure 8.1 shows an example of an MRP metro ring.

Figure 8.1 Metro ring – normal state



The ring in this example consists of four MRP nodes (Foundry switches). Each node has two interfaces with the ring. Each node also is connected to a separate customer network. The nodes forward Layer 2 traffic to and from the customer networks through the ring. The ring interfaces are all in one port-based VLAN. Each customer interface can be in the same VLAN as the ring or in a separate VLAN.

One node, is configured as the master node of the MRP ring. One of the two interfaces on the master node is configured as the primary interface; the other is the secondary interface. The primary interface originates Ring Health Packets (RHPs), which are used to monitor the health of the ring. An RHP is forwarded on the ring to the

next interface until it reaches the secondary interface of the master node. The secondary interface blocks the packet to prevent a Layer 2 loops.

Configuration Notes

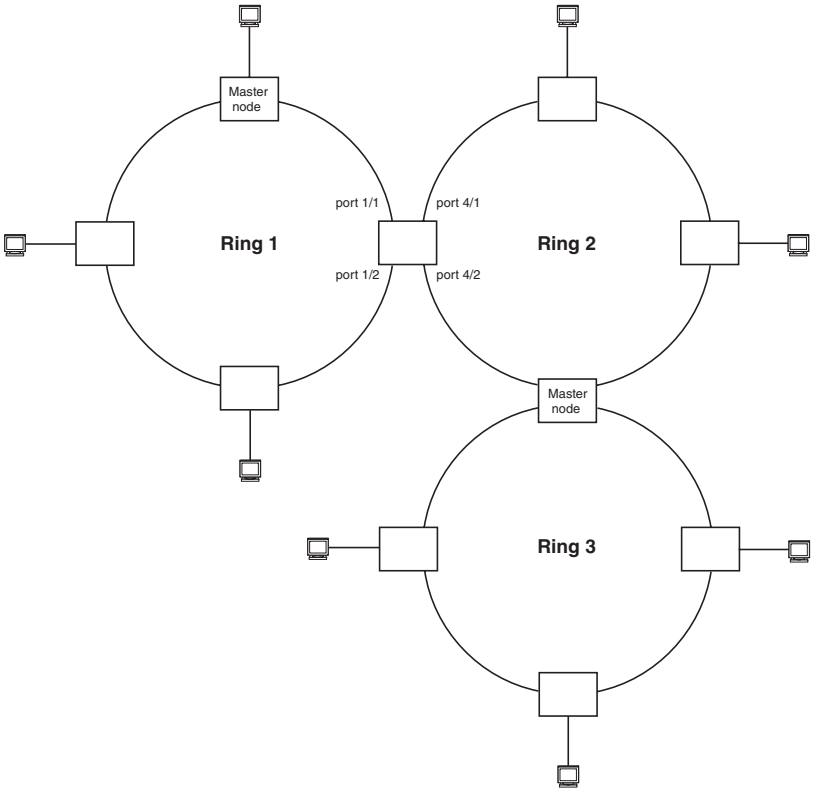
- When you configure MRP, Foundry recommends that you disable one of the ring interfaces before beginning the ring configuration. Disabling an interface prevents a Layer 2 loop from occurring while you are configuring MRP on the ring nodes. Once MRP is configured and enabled on all the nodes, you can re-enable the interface.
- MRP I is supported in all FESX, FSX, and FWSX devices and their associated software releases.
- The above configurations are capable of being configured as MRP masters or MRP members (for different rings).

MRP Rings Without Shared Interfaces (MRP Phase 1)

MRP Phase 1 allows you to configure multiple MRP rings, as shown in Figure 8.2, but the rings cannot share the same link. For example, you cannot configure ring 1 and ring 2 to each have interfaces 1/1 and 1/2.

Also, when you configure an MRP ring, any node on the ring can be designated as the master node for the ring. A master node can be the master node of more than one ring. (See Figure 8.2.) Each ring is an independent ring and RHP packets are processed within each ring.

Figure 8.2 Metro ring – multiple rings

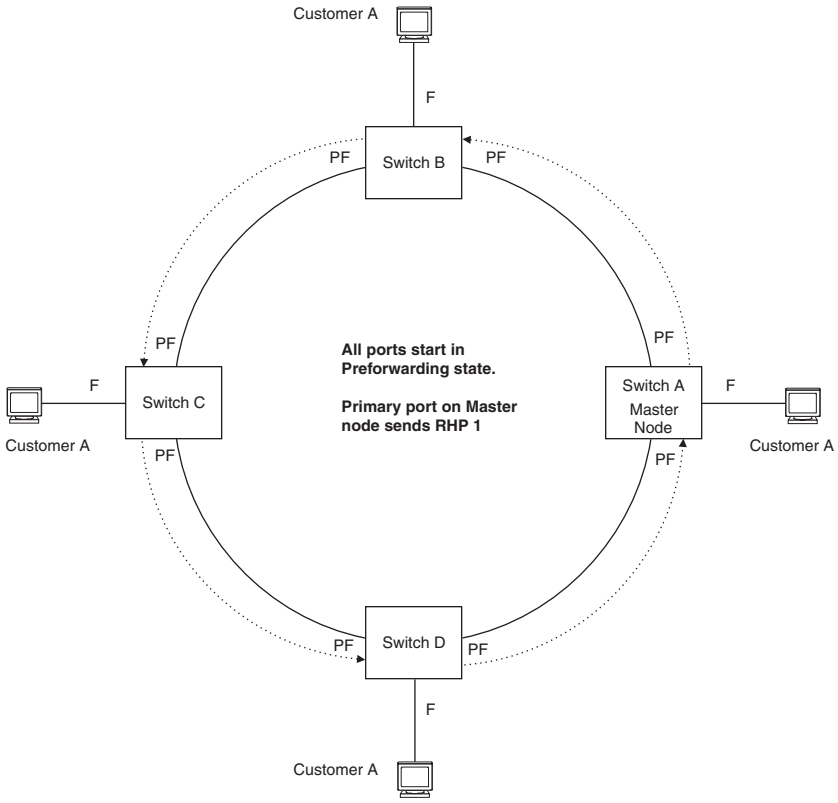


In this example, two nodes are each configured with two MRP rings. Any node in a ring can be the master for its ring. A node also can be the master for more than one ring.

Ring Initialization

The ring shown in Figure 8.1 shows the port states in a fully initialized ring without any broken links. Figure 8.3 shows the initial state of the ring, when MRP is first enabled on the ring's switches. All ring interfaces on the master node and member nodes begin in the Preforwarding state (PF).

Figure 8.3 Metro ring – initial state



MRP uses Ring Health Packets (RHPs) to monitor the health of the ring. An RHP is an MRP protocol packet. The source address is the MAC address of the master node and the destination MAC address is a protocol address for MRP. The Master node generates RHPs and sends them on the ring. The state of a ring port depends on the RHPs.

A ring interface can have one of the following MRP states:

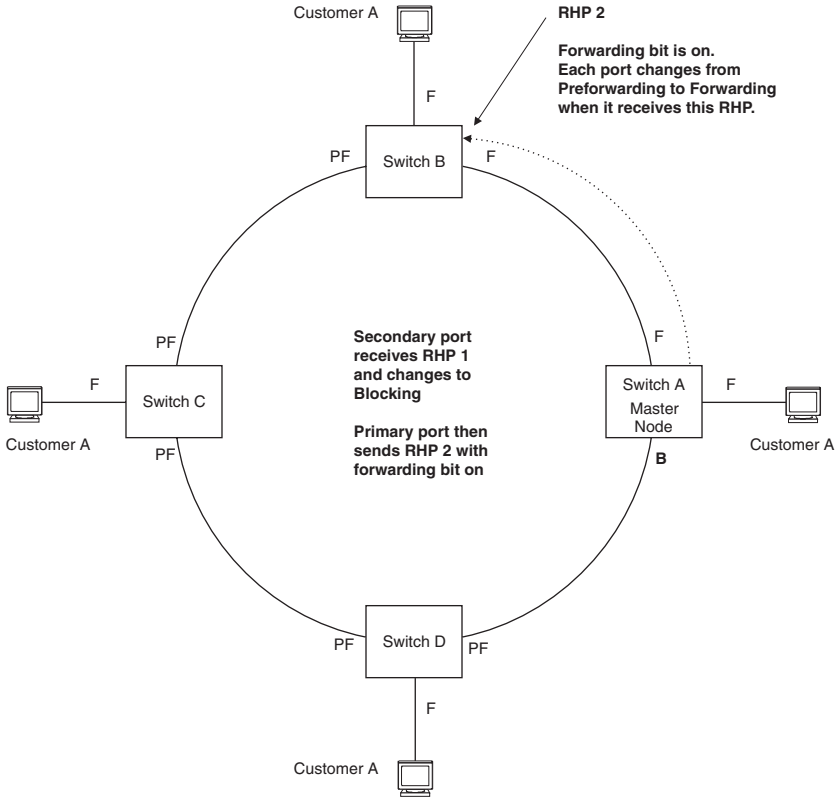
- Preforwarding (PF) – The interface can forward RHPS but cannot forward data. All ring ports being in this state when you enable MRP.
- Forwarding (F) – The interface can forward data as well as RHPs. An interface changes from Preforwarding to Forwarding when the port's preforwarding time expires. This occurs if the port does not receive an RHP from the Master, or if the forwarding bit in the RHPs received by the port is off. This indicates a break in the ring. The port heals the ring by changing its state to Forwarding. The preforwarding time is the number of milliseconds the port will remain in the Preforwarding state before changing to the Forwarding state, even without receiving an RHP.
- Blocking (B) – The interface cannot forward data. Only the secondary interface on the Master node can be Blocking.

When MRP is enabled, all ports begin in the Preforwarding state. The primary interface on the Master node, although it is in the Preforwarding state like the other ports, immediately sends an RHP onto the ring. The secondary port on the Master node listens for the RHP.

- If the secondary port receives the RHP, all links in the ring are up and the port changes its state to Blocking. The primary port then sends another MRP with its forwarding bit set on. As each of the member ports receives the RHP, the ports changes their state to Forwarding. Typically, this occurs in sub-second time. The ring very quickly enters the fully initialized state.
- If the secondary port does not receive the RHP by the time the preforwarding time expires, a break has occurred in the ring. The port changes its state to Forwarding. The member ports also change their states from Preforwarding to Forwarding as their preforwarding timers expire. The ring is not intact, but data can still travel among the nodes using the links that are up.

Figure 8.4 shows an example.

Figure 8.4 Metro ring – from Preforwarding to Forwarding

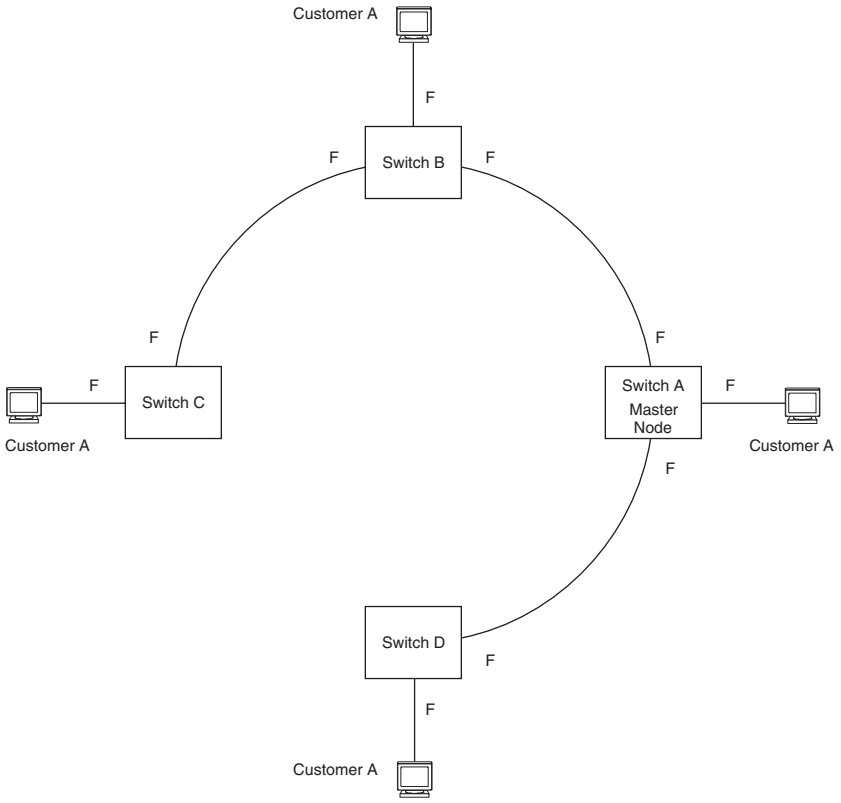


Each RHP also has a sequence number. MRP can use the sequence number to determine the round-trip time for RHPs in the ring. See “Using MRP Diagnostics” on page 8-12.

How Ring Breaks Are Detected and Healed

Figure 8.5 shows ring interface states following a link break. MRP quickly heals the ring and preserves connectivity among the customer networks.

Figure 8.5 Metro ring – ring break



If a break in the ring occurs, MRP heals the ring by changing the states of some of the ring interfaces.

- Blocking interface – The Blocking interface on the Master node has a dead timer. If the dead time expires before the interface receives one of its ring's RHPs, the interface changes state to Preforwarding. Once the secondary interface changes state to Preforwarding:
  - If the interface receives an RHP, the interface changes back to the Blocking state and resets the dead timer.
  - If the interface does not receive an RHP for its ring before the Preforwarding time expires, the interface changes to the Forwarding state, as shown in Figure 8.5.
- Forwarding interfaces – Each member interface remains in the Forwarding state.

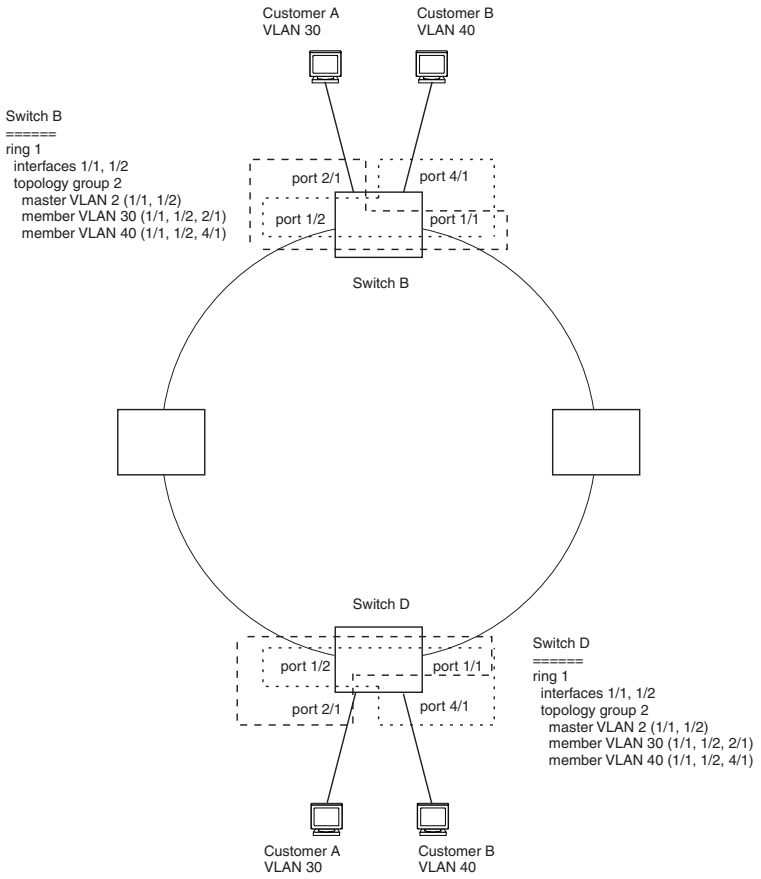
When the broken link is repaired, the link's interfaces come up in the Preforwarding state, which allows RHPs to travel through the restored interfaces and reach the secondary interface on the Master node.

- If an RHP reaches the Master node's secondary interface, the ring is intact. The secondary interface changes to Blocking. The Master node sets the forwarding bit on in the next RHP. When the restored interfaces receive this RHP, they immediately change state to Forwarding.
- If an RHP does not reach the Master node's secondary interface, the ring is still broken. The Master node does not send an RHP with the forwarding bit on. In this case, the restored interfaces remain in the Preforwarding state until the preforwarding timer expires, then change to the Forwarding state.

Master VLANs and Customer VLANs

All the ring ports must be in the same VLAN. Placing the ring ports in the same VLAN provides Layer 2 connectivity for a given customer across the ring. Figure 8.6 shows an example.

Figure 8.6 Metro ring – ring VLAN and customer VLANs



Notice that each customer has their own VLAN. Customer A has VLAN 30 and Customer B has VLAN 40. Customer A's host attached to Switch D can reach the Customer A host attached to Switch B at Layer 2 through the ring. Since Customer A and Customer B are on different VLANs, they will not receive each other's traffic.

You can configure MRP separately on each customer VLAN. However, this is impractical if you have many customers. To simplify configuration when you have a lot of customers (and therefore a lot of VLANs), you can use a topology group.

A topology group enables you to control forwarding in multiple VLANs using a single instance of a Layer 2 protocol such as MRP. A topology group contains a master VLAN and member VLANs. The master VLAN contains all the configuration parameters for the Layer 2 protocol (STP, MRP, or VSRP). The member VLANs use the Layer 2 configuration of the master VLAN.

In Figure 8.6, VLAN 2 is the master VLAN and contains the MRP configuration parameters for ring 1. VLAN 30 and VLAN 40, the customer VLANs, are member VLANs in the topology group. Since a topology group is used, a single instance of MRP provides redundancy and loop prevention for both the customer VLANs.

If you use a topology group:

- The master VLAN must contain the ring interfaces. The ports must be tagged, since they will be shared by multiple VLANs.
- The member VLAN for a customer must contain the two ring interfaces and the interfaces for the customer. Since these interfaces are shared with the master VLAN, they must be tagged. Do not add another customer's interfaces to the VLAN.

For more information about topology groups, see "Topology Groups" on page 8-1.

See "MRP CLI Example" on page 8-16 for the configuration commands required to implement the MRP configuration shown in Figure 8.6.

Configuring MRP

To configure MRP, perform the following tasks. You need to perform the first task on only one of the nodes. Perform the remaining tasks on all the nodes.

- Disable one of the ring interfaces. This prevents a Layer 2 loop from occurring while you are configuring the devices for MRP.
- Add an MRP ring to a port-based VLAN. When you add a ring, the CLI changes to the configuration level for the ring, where you can perform the following tasks.
  - Optionally, specify a name for the ring.
  - On the master node only, enable the device to be the master for the ring. Each ring can have only one master node.
  - Specify the MRP interfaces. Each device has two interfaces to an MRP ring.
  - Optionally, change the hello time and the preforwarding time. These parameters control how quickly failover occurs following a change in the state of a link in the ring.
  - Enable the ring.
- Optionally, add the ring's VLAN to a topology group to add more VLANs to the ring. If you use a topology group, make sure you configure MRP on the group's master VLAN. See "Topology Groups" on page 8-1.
- Re-enable the interface you disabled to prevent a Layer 2 loop. Once MRP is enabled, MRP will prevent the Layer 2 loop.

Adding an MRP Ring to a VLAN

To add an MRP ring to a VLAN, enter commands such as the following.

NOTE: If you plan to use a topology group to add VLANs to the ring, make sure you configure MRP on the topology group's master VLAN.

```
FastIron SuperX Router(config)# vlan 2
FastIron SuperX Router(config-vlan-2)# metro-ring 1
FastIron SuperX Router(config-vlan-2-mrp-1)# name CustomerA
FastIron SuperX Router(config-vlan-2-mrp-1)# master
FastIron SuperX Router(config-vlan-2-mrp-1)# ring-interface ethernet 1/1 ethernet 1/
2
FastIron SuperX Router(config-vlan-2-mrp-1)# enable
```

These commands configure an MRP ring on VLAN 2. The ring ID is 1, the ring name is CustomerA, and this node (this Foundry device) is the master for the ring. The ring interfaces are 1/1 and 1/2. Interface 1/1 is the primary interface and 1/2 is the secondary interface. The primary interface will initiate RHPs by default. The ring takes effect in VLAN 2.

Syntax: [no] metro-ring <ring-id>

The <ring-id> parameter specifies the ring ID and can be from 1 – 255. Configure the same ring ID on each of the nodes in the ring.

Syntax: [no] name <string>

The <string> parameter specifies a name for the ring. The name is optional, but it can be up to 20 characters long and can include blank spaces. If you use a name that has blank spaces, enclose the name in double quotation marks (for example: "Customer A").

Syntax: [no] master

Configures this node as the master node for the ring. Enter this command only on one node in the ring. The node is a member (non-master) node by default.

Syntax: [no] ring-interface ethernet <primary-if> ethernet <secondary-if>

The **ethernet** <primary-if> parameter specifies the primary interface. On the master node, the primary interface is the one that originates RHPs. Ring control traffic and Layer 2 data traffic will flow in the outward direction from this interface by default. On member nodes, the direction of traffic flow depends on the traffic direction selected by the master node. Therefore, on a member node, the order in which you enter the interfaces does not matter.

The **ethernet** <secondary-if> parameter specifies the secondary interface.

NOTE: To take advantage of every interface in a Metro network, you can configure another MRP ring and either configure a different Master node for the ring or reverse the configuration of the primary and secondary interfaces on the Master node. Configuring multiple rings enables you to use all the ports in the ring. The same port can forward traffic one ring while blocking traffic for another ring.

Syntax: [no] enable

The **enable** command enables the ring.

Changing the Hello and PreForwarding Times

You also can change the RHP hello time and preforwarding time. To do so, enter commands such as the following:

```
FastIron SuperX Router(config-vlan-2-mrp-1)# hello-time 200
FastIron SuperX Router(config-vlan-2-mrp-1)# preforwarding-time 400
```

These commands change the hello time to 200 ms and change the preforwarding time to 400 ms.

NOTE: The preforwarding time must be at least twice the value of the hello time and must be a multiple of the hello time.

Syntax: [no] hello-time <ms>

Syntax: [no] preforwarding-time <ms>

The <ms> specifies the number of milliseconds. For the hello time, you can specify from 100 – 1000 (one second). The default hello time is 100 ms. The preforwarding time can be from 200 – 5000 ms, but must be at least twice the value of the hello time and must be a multiple of the hello time. The default preforwarding time is 300 ms. A change to the hello time or preforwarding time takes effect as soon as you enter the command.

NOTE: You can use MRP ring diagnostics to determine whether you need to change the hello time and preforwarding time. See “Using MRP Diagnostics”.

Using MRP Diagnostics

The MRP diagnostics feature calculates how long it takes for RHP packets to travel through the ring. When you enable MRP diagnostics, the software tracks RHP packets according to their sequence numbers and calculates how long it takes an RHP packet to travel one time through the entire ring. When you display the diagnostics, the CLI shows the average round-trip time for the RHP packets sent since you enabled diagnostics. The calculated results have a granularity of 1 microsecond.

Enabling MRP Diagnostics

To enable MRP diagnostics for a ring, enter the following command on the Master node, at the configuration level for the ring:

```
FastIron SuperX Router(config-vlan-2-mrp-1)# diagnostics
```

Syntax: [no] diagnostics

NOTE: This command is valid only on the master node.

Displaying MRP Diagnostics

To display MRP diagnostics results, enter the following command on the Master node:

```
FastIron SuperX Router(config)# show metro 1 diag

Metro Ring 1 - CustomerA
=============
diagnostics results

Ring       Diag       RHP average     Recommended     Recommended
id         state      time(microsec)  hello time(ms)  Prefwing time(ms)
2          enabled    125             100             300

Diag frame sent     Diag frame lost
1230                0
```

Syntax: show metro <ring-id> diag

This display shows the following information.

Table 8.3: CLI Display of MRP Ring Diagnostic Information

This Field...	Displays...
Ring id	The ring ID.
Diag state	The state of ring diagnostics.
RHP average time	The average round-trip time for an RHP packet on the ring. The calculated time has a granularity of 1 microsecond.
Recommended hello time	The hello time recommended by the software based on the RHP average round-trip time.
Recommended Prefwing time	The preforwarding time recommended by the software based on the RHP average round-trip time.
Diag frame sent	The number of diagnostic RHPs sent for the test.
Diag frame lost	The number of diagnostic RHPs lost during the test.

If the recommended hello time and preforwarding time are different from the actual settings and you want to change them, see “Configuring MRP” on page 8-11.

Displaying MRP Information

You can display the following MRP information:

- Topology group configuration information
- Ring configuration information and statistics

Displaying Topology Group Information

To display topology group information, enter the following command:

Syntax: show topology-group [<group-id>]

See “Displaying Topology Group Information” on page 8-3 for more information.

Displaying Ring Information

To display ring information, enter the following command:

```
FastIron SuperX Router(config)# show metro

Metro Ring 1
=============
Ring       State     Ring       Master     Topo       Hello      Prefwing
id                   role       vlan       group      time(ms)   time(ms)
2          enabled   member     2          not conf   100        300

Ring interfaces    Interface role     Forwarding state    Active interface    Interface Type
ethernet 1/1       primary            disabled            none                Regular
ethernet 1/2       secondary          forwarding          ethernet 2          Tunnel

RHPs sent          RHPs rcvd          TC RHPs rcvd        State changes
3                    0                    0                   4
```

Syntax: show metro [<ring-id>]

This display shows the following information.

Table 8.4: CLI Display of MRP Ring Information

This Field...	Displays...
Ring id	The ring ID
State	The state of MRP. The state can be one of the following: • enabled – MRP is enabled • disabled – MRP is disabled
Ring role	Whether this node is the master for the ring. The role can be one of the following: • master • member
Master vlan	The ID of the master VLAN in the topology group used by this ring. If a topology group is used by MRP, the master VLAN controls the MRP settings for all VLANs in the topology group. **Note**: The topology group ID is 0 if the MRP VLAN is not the master VLAN in a topology group. Using a topology group for MRP configuration is optional.
Topo group	The topology group ID.
Hello time	The interval, in milliseconds, at which the Forwarding port on the ring's master node sends Ring Hello Packets (RHPs).

Table 8.4: CLI Display of MRP Ring Information (Continued)

This Field...	Displays...
Prefwing time	The number of milliseconds an MRP interface that has entered the Preforwarding state will wait before changing to the Forwarding state. If a member port in the Preforwarding state does not receive an RHP within the Preforwarding time (Prefwing time), the port assumes that a topology change has occurred and changes to the Forwarding state. The secondary port on the Master node changes to Blocking if it receives an RHP, but changes to Forwarding if the port does not receive an RHP before the preforwarding time expires. **Note**: A member node's Preforwarding interface also changes from Preforwarding to Forwarding if it receives an RHP whose forwarding bit is on.
Ring interfaces	The device's two interfaces with the ring. **Note**: If the interfaces are trunk groups, only the primary ports of the groups are listed.

Interface role	The interface role can be one of the following:
	• primary
	  • Master node – The interface generates RHPs.
	  • Member node – The interface forwards RHPs received on the other interface (the secondary interface).
	• secondary – The interface does not generate RHPs.
	  • Master node – The interface listens for RHPs.
	  • Member node – The interface receives RHPs.

Forwarding state	Whether MRP Forwarding is enabled on the interface. The forwarding state can be one of the following: • blocking – The interface is blocking Layer 2 data traffic and RHPs • disabled – The interface is down • forwarding – The interface is forwarding Layer 2 data traffic and RHPs • preforwarding – The interface is listening for RHPs but is blocking Layer 2 data traffic
Active interface	The physical interfaces that are sending and receiving RHPs. **Note**: If a port is disabled, its state is shown as "disabled". **Note**: If an interface is a trunk group, only the primary port of the group is listed.
Interface Type	Shows if the interface is a regular port or a tunnel port.
RHPs sent	The number of RHPs sent on the interface. **Note**: This field applies only to the master node. On non-master nodes, this field contains 0. This is because the RHPs are forwarded in hardware on the non-master nodes.

Table 8.4: CLI Display of MRP Ring Information (Continued)

This Field...	Displays...
RHPs rcvd	The number of RHPs received on the interface. **Note**: On most Foundry devices, this field applies only to the master node. On non-master nodes, this field contains 0. This is because the RHPs are forwarded in hardware on the non-master nodes. However, on the FastIron devices, the RHP received counter on non-master MRP nodes increment. This is because, on FastIron devices, the CPU receives a copy of the RHPs forwarded in hardware.
TC RHPs rcvd	The number of Topology Change RHPs received on the interface. A Topology Change RHP indicates that the ring topology has changed.
State changes	The number of MRP interface state changes that have occurred. The state can be one of the states listed in the Forwarding state field.

MRP CLI Example

The following examples show the CLI commands required to implement the MRP configuration shown in Figure 8.6 on page 8-10.

NOTE: For simplicity, the figure shows the VLANs on only two switches. The CLI examples implement the ring on all four switches.

Commands on Switch A (Master Node)

The following commands configure a VLAN for the ring. The ring VLAN must contain both of the node's interfaces with the ring. Add these interfaces as tagged interfaces, since the interfaces also must be in each of the customer VLANs configured on the node.

```
FastIron SuperX Router(config)# vlan 2
FastIron SuperX Router(config-vlan-2)# tag ethernet 1/1 to 1/2
FastIron SuperX Router(config-vlan-2)# metro-ring 1
FastIron SuperX Router(config-vlan-2-mrp-1)# name "Metro A"
FastIron SuperX Router(config-vlan-2-mrp-1)# master
FastIron SuperX Router(config-vlan-2-mrp-1)# ring-interface ethernet 1/1 ethernet 1/
2
FastIron SuperX Router(config-vlan-2-mrp-1)# enable
FastIron SuperX Router(config-vlan-2-mrp-1)# exit
FastIron SuperX Router(config-vlan-2)# exit
```

The following commands configure the customer VLANs. The customer VLANs must contain both the ring interfaces as well as the customer interfaces.

```
FastIron SuperX Router(config)# vlan 30
FastIron SuperX Router(config-vlan-30)# tag ethernet 1/1 to 1/2
FastIron SuperX Router(config-vlan-30)# tag ethernet 2/1
FastIron SuperX Router(config-vlan-30)# exit
FastIron SuperX Router(config)# vlan 40
FastIron SuperX Router(config-vlan-40)# tag ethernet 1/1 to 1/2
FastIron SuperX Router(config-vlan-40)# tag ethernet 4/1
FastIron SuperX Router(config-vlan-40)# exit
```

The following commands configure topology group 1 on VLAN 2. The master VLAN is the one that contains the MRP configuration. The member VLANs use the MRP parameters of the master VLAN. The control interfaces (the ones shared by the master VLAN and member VLAN) also share MRP state.

```
FastIron SuperX Router(config)# topology-group 1
FastIron SuperX Router(config-topo-group-1)# master-vlan 2
```



```
FastIron SuperX Router(config-topo-group-1)# member-vlan 30
FastIron SuperX Router(config-topo-group-1)# member-vlan 40
```

Commands on Switch B

The commands for configuring Switches B, C, and D are similar to the commands for configuring Switch A, with two differences: the nodes are not configured to be the ring master. Omitting the **master** command is required for non-master nodes.

```
FastIron SuperX Router(config)# vlan 2
FastIron SuperX Router(config-vlan-2)# tag ethernet 1/1 to 1/2
FastIron SuperX Router(config-vlan-2)# metro-ring 1
FastIron SuperX Router(config-vlan-2-mrp-1)# name "Metro A"
FastIron SuperX Router(config-vlan-2-mrp-1)# ring-interface ethernet 1/1 ethernet 1/
2
FastIron SuperX Router(config-vlan-2-mrp-1)# enable
FastIron SuperX Router(config-vlan-2)# exit

FastIron SuperX Router(config)# vlan 30
FastIron SuperX Router(config-vlan-30)# tag ethernet 1/1 to 1/2
FastIron SuperX Router(config-vlan-30)# tag ethernet 2/1
FastIron SuperX Router(config-vlan-30)# exit
FastIron SuperX Router(config)# vlan 40
FastIron SuperX Router(config-vlan-40)# tag ethernet 1/1 to 1/2
FastIron SuperX Router(config-vlan-40)# tag ethernet 4/1
FastIron SuperX Router(config-vlan-40)# exit

FastIron SuperX Router(config)# topology-group 1
FastIron SuperX Router(config-topo-group-1)# master-vlan 2
FastIron SuperX Router(config-topo-group-1)# member-vlan 30
FastIron SuperX Router(config-topo-group-1)# member-vlan 40
```

Commands on Switch C

```
FastIron SuperX Router(config)# vlan 2
FastIron SuperX Router(config-vlan-2)# tag ethernet 1/1 to 1/2
FastIron SuperX Router(config-vlan-2)# metro-ring 1
FastIron SuperX Router(config-vlan-2-mrp-1)# name "Metro A"
FastIron SuperX Router(config-vlan-2-mrp-1)# ring-interface ethernet 1/1 ethernet 1/
2
FastIron SuperX Router(config-vlan-2-mrp-1)# enable
FastIron SuperX Router(config-vlan-2)# exit

FastIron SuperX Router(config)# vlan 30
FastIron SuperX Router(config-vlan-30)# tag ethernet 1/1 to 1/2
FastIron SuperX Router(config-vlan-30)# tag ethernet 2/1
FastIron SuperX Router(config-vlan-30)# exit
FastIron SuperX Router(config)# vlan 40
FastIron SuperX Router(config-vlan-40)# tag ethernet 1/1 to 1/2
FastIron SuperX Router(config-vlan-40)# tag ethernet 4/1
FastIron SuperX Router(config-vlan-40)# exit

FastIron SuperX Router(config)# topology-group 1
FastIron SuperX Router(config-topo-group-1)# master-vlan 2
FastIron SuperX Router(config-topo-group-1)# member-vlan 30
FastIron SuperX Router(config-topo-group-1)# member-vlan 40
```

Commands on Switch D

```
FastIron SuperX Router(config)# vlan 2
FastIron SuperX Router(config-vlan-2)# tag ethernet 1/1 to 1/2
FastIron SuperX Router(config-vlan-2)# metro-ring 1
FastIron SuperX Router(config-vlan-2-mrp-1)# name "Metro A"
```

```
FastIron SuperX Router(config-vlan-2-mrp-1)# ring-interface ethernet 1/1 ethernet 1/
2
FastIron SuperX Router(config-vlan-2-mrp-1)# enable
FastIron SuperX Router(config-vlan-2)# exit

FastIron SuperX Router(config)# vlan 30
FastIron SuperX Router(config-vlan-30)# tag ethernet 1/1 to 1/2
FastIron SuperX Router(config-vlan-30)# tag ethernet 2/1
FastIron SuperX Router(config-vlan-30)# exit
FastIron SuperX Router(config)# vlan 40
FastIron SuperX Router(config-vlan-40)# tag ethernet 1/1 to 1/2
FastIron SuperX Router(config-vlan-40)# tag ethernet 4/1
FastIron SuperX Router(config-vlan-40)# exit

FastIron SuperX Router(config)# topology-group 1
FastIron SuperX Router(config-topo-group-1)# master-vlan 2
FastIron SuperX Router(config-topo-group-1)# member-vlan 30
FastIron SuperX Router(config-topo-group-1)# member-vlan 40
```

Virtual Switch Redundancy Protocol (VSRP)

Virtual Switch Redundancy Protocol (VSRP) is a Foundry proprietary protocol that provides redundancy and sub-second failover in Layer 2 and Layer 3 mesh topologies. Based on the Foundry Virtual Router Redundancy Protocol Extended (VRRPE), VSRP provides one or more backups for a Layer 2 Switch or Layer 3 Switch. If the active Layer 2 Switch or Layer 3 Switch becomes unavailable, one of the backups takes over as the active device and continues forwarding traffic for the network.

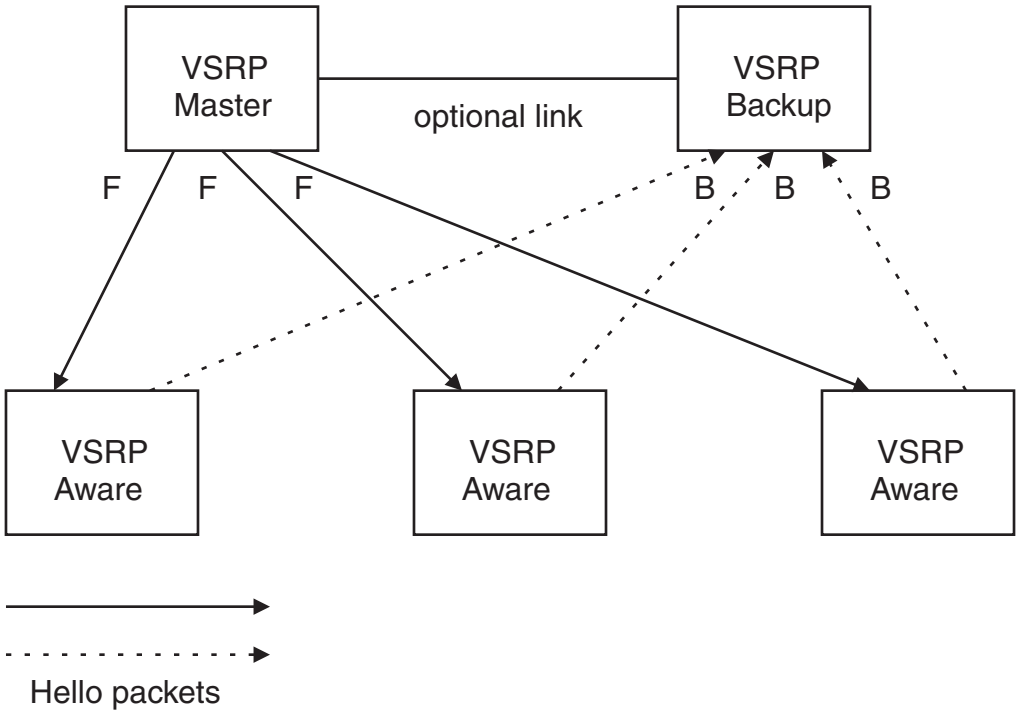
The FastIron family of switches support full VSRP as well as ***VSRP-awareness***. A Foundry device that is not itself configured for VSRP but is connected to a Foundry device that is configured for VSRP, is ***VSRP aware***.

You can use VSRP for Layer 2, Layer 3, or for both layers. On Layer 3 Switches, Layer 2 and Layer 3 share the same VSRP configuration information. On Layer 2 Switches, VSRP applies only to Layer 2.

NOTE: VSRP and 802.1Q-n-Q tagging are not supported together on the same device.

Figure 8.7 shows an example of a VSRP configuration.

Figure 8.7 VSRP mesh – redundant paths for Layer 2 and Layer 3 traffic



In this example, two Foundry devices are configured as redundant paths for VRID 1. On each of the devices, a Virtual Router ID (VRID) is configured on a port-based VLAN. Since VSRP is primarily a Layer 2 redundancy protocol, the VRID applies to the entire VLAN. However, you can selectively remove individual ports from the VRID if needed.

Following Master election (described below), one of the Foundry devices becomes the Master for the VRID and sets the state of all the VLAN's ports to Forwarding. The other device is a Backup and sets all the ports in its VRID VLAN to Blocking.

If a failover occurs, the Backup becomes the new Master and changes all its VRID ports to the Forwarding state.

Other Foundry devices can use the redundant paths provided by the VSRP devices. In this example, three Foundry devices use the redundant paths. A Foundry device that is not itself configured for VSRP but is connected to a Foundry device that is configured for VSRP, is ***VSRP aware***. In this example, the three Foundry devices connected to the VSRP devices are VSRP aware. A Foundry device that is VSRP aware can failover its link to the new Master in sub-second time, by changing the MAC address associated with the redundant path.

When you configure VSRP, make sure each of the non-VSRP Foundry devices connected to the VSRP devices has a separate link to each of the VSRP devices.

Layer 2 and Layer 3 Redundancy

You can configure VSRP to provide redundancy for Layer 2 only or also for Layer 3.

- Layer 2 only – The Layer 2 links are backup up but specific IP addresses are not backed up.
- Layer 2 and Layer 3 – The Layer 2 links are backup up and a specific IP address is also backed up. Layer 3 VSRP is the same as VRRPE. However, using VSRP provides redundancy at both layers at the same time.

Layer 2 Switches support Layer 2 VSRP only. Layer 3 Switches support Layer 2 and Layer 3 redundancy. You can configure a Layer 3 Switch for either Layer 2 only or Layer 2 and Layer 3. To configure for Layer 3, specify the IP address you are backing up.

NOTE: If you want to provide Layer 3 redundancy only, disable VSRP and use VRRPE.

Master Election and Failover

Each VSRP device advertises its VSRP priority in Hello messages. During Master election, the VSRP device with the highest priority for a given VRID becomes the Master for that VRID. After Master election, the Master sends Hello messages at regular intervals to inform the Backups that the Master is healthy.

If there is a tie for highest VSRP priority, the tie is resolved as follows:

- Layer 2 Switches – The Layer 2 Switch with the higher management IP address becomes the Master.
    - Switches with management IP addresses are preferred over switches without management IP addresses.
    - If neither of the switches has a management IP address, then the switch with the higher MAC address becomes the Master. (VSRP compares the MAC addresses of the ports configured for the VRID, not the base MAC addresses of the switches.)
- Layer 3 Switches – The Layer 3 Switch whose virtual routing interface has a higher IP address becomes the master.

VSRP Failover

Each Backup listens for Hello messages from the Master. The Hello messages indicate that the Master is still available. If the Backups stop receiving Hello messages from the Master, the election process occurs again and the Backup with the highest priority becomes the new Master.

Each Backup waits for a specific period of time, the Dead Interval, to receive a new Hello message from the Master. If the Backup does not receive a Hello message from the Master by the time the Dead Interval expires, the Backup sends a Hello message of its own, which includes the Backup's VSRP priority, to advertise the Backup's intent to become the Master. If there are multiple Backups for the VRID, each Backup sends a Hello message.

When a Backup sends a Hello message announcing its intent to become the Master, the Backup also starts a hold-down timer. During the hold-down time, the Backup listens for a Hello message with a higher priority than its own.

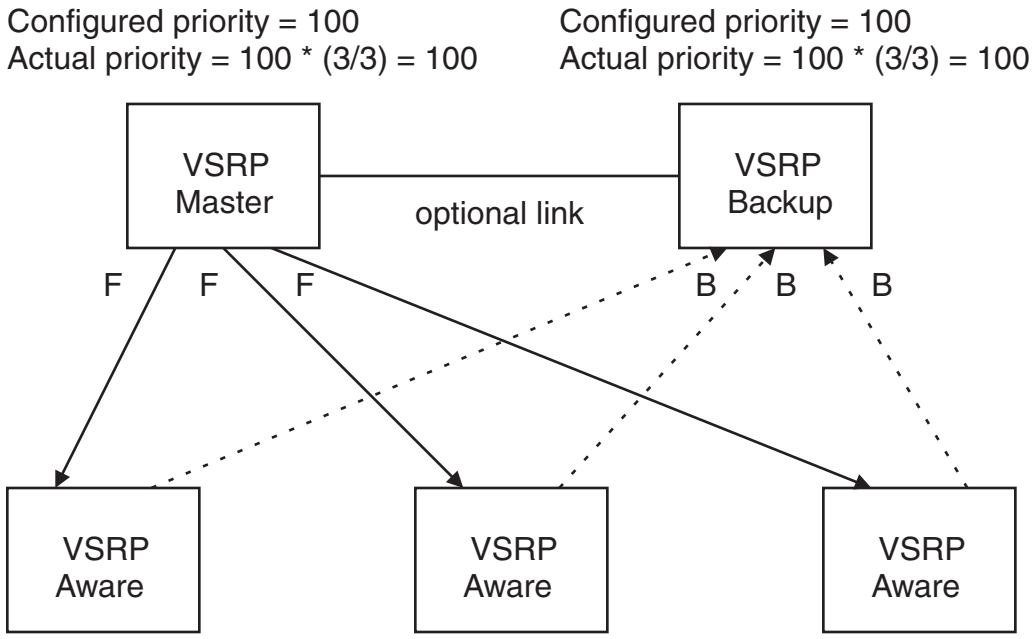
- If the Backup receives a Hello message with a higher priority than its own, the Backup resets its Dead Interval and returns to normal Backup status.
- If the Backup does not receive a Hello message with a higher priority than its own by the time the hold-down timer expires, the Backup becomes the new Master and starts forwarding Layer 2 traffic on all ports.

If you increase the timer scale value, each timer's value is divided by the scale value. To achieve sub-second failover times, you can change the scale to a value up to 10. This shortens all the VSRP timers to 10 percent of their configured values.

VSRP Priority Calculation

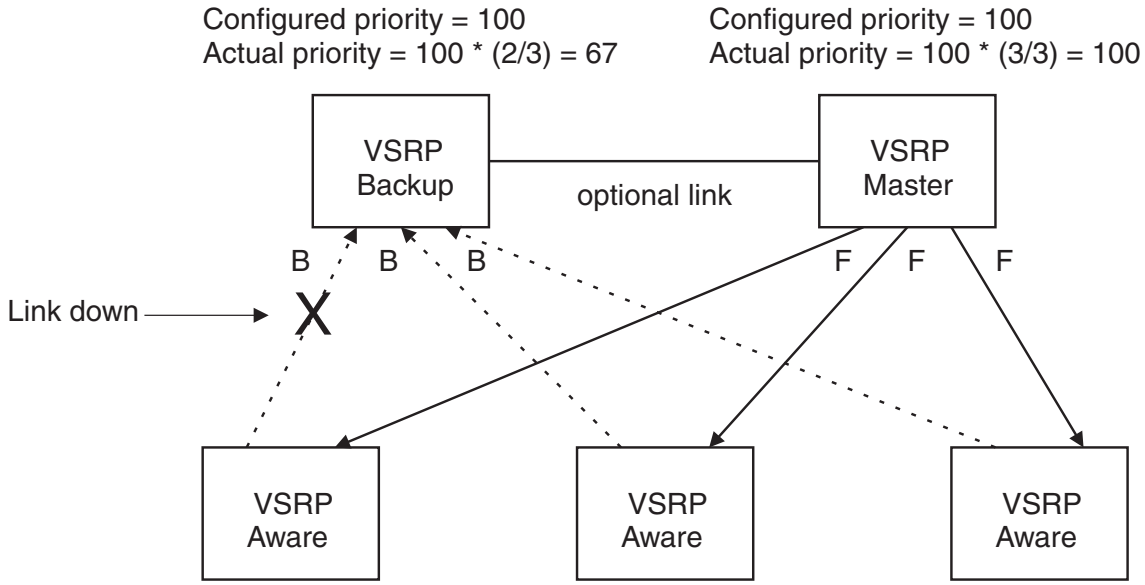
Each VSRP device has a VSRP priority for each VRID and its VLAN. The VRID is used during Master election for the VRID. By default, a device's VSRP priority is the value configured on the device (which is 100 by default). However, to ensure that a Backup with a high number of up ports for a given VRID is elected, the device reduces the priority if a port in the VRID's VLAN goes down. For example, if two Backups each have a configured priority of 100, and have three ports in VRID 1 in VLAN 10, each Backup begins with an equal priority, 100. This is shown in Figure 8.8

Figure 8.8 VSRP priority



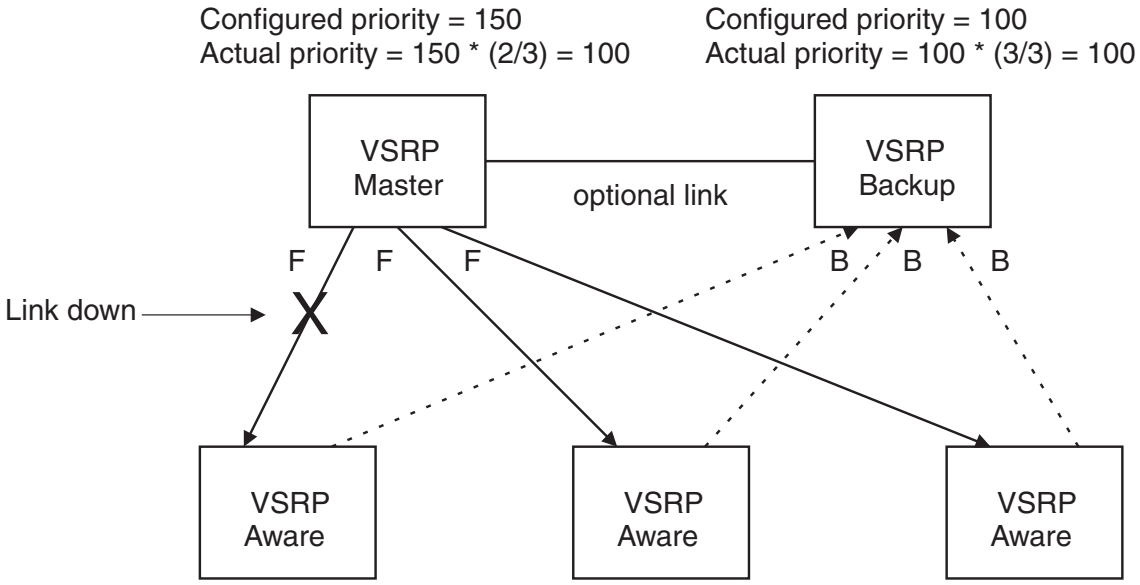
However, if one of the VRID's ports goes down on one of the Backups, that Backup's priority is reduced. If the Master's priority is reduced enough to make the priority lower than a Backup's priority, the VRID fails over to the Backup. Figure 8.9 shows an example.

Figure 8.9 VSRP priority recalculation



You can reduce the sensitivity of a VSRP device to failover by increasing its configured VSRP priority. For example, you can increase the configured priority of the VSRP device on the left in Figure 8.9 to 150. In this case, failure of a single link does not cause failover. The link failure caused the priority to be reduced to 100, which is still equal to the priority of the other device. This is shown in Figure 8.10.

Figure 8.10 VSRP priority bias

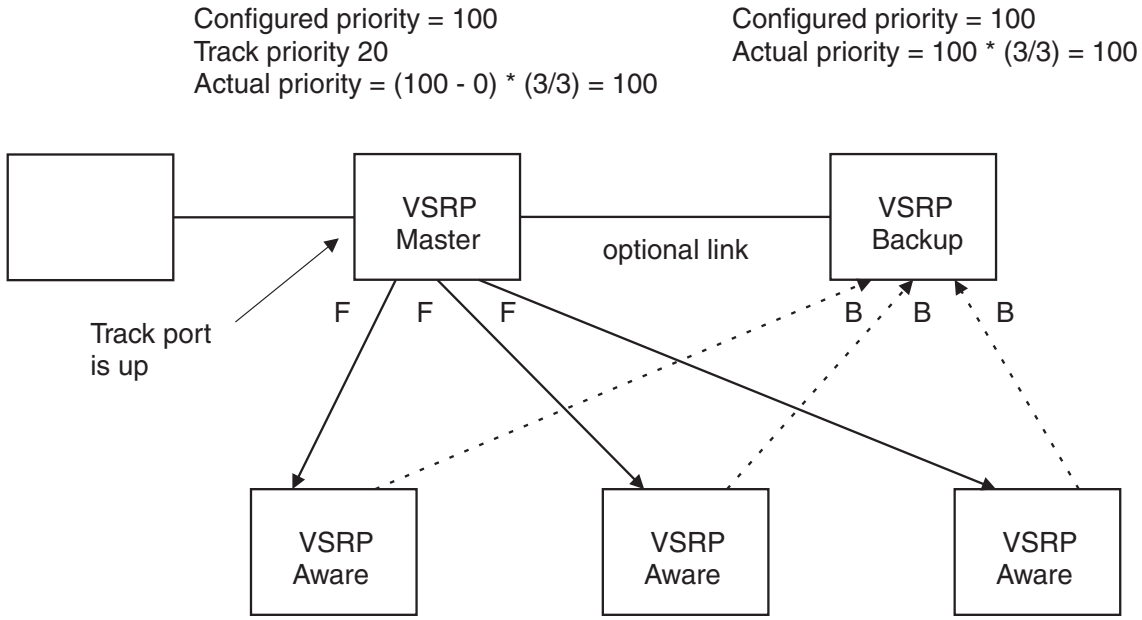


Track Ports

Optionally, you can configure track ports to be included during VSRP priority calculation. In VSRP, a ***track port*** is a port that is not a member of the VRID's VLAN, but whose state is nonetheless considered when the priority is calculated. Typically, a track port represents the exit side of traffic received on the VRID ports. By default, no track ports are configured.

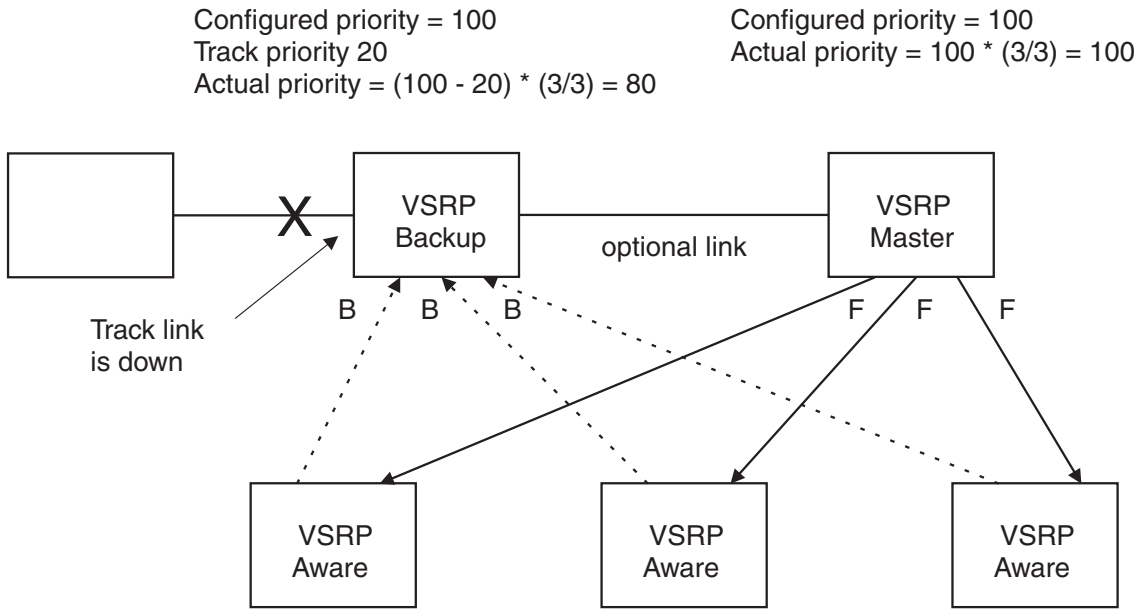
When you configure a track port, you assign a priority value to the port. If the port goes down, VSRP subtracts the track port's priority value from the configured VSRP priority. For example, if the you configure a track port with priority 20 and the configured VSRP priority is 100, the software subtracts 20 from 100 if the track port goes down, resulting in a VSRP priority of 80. The new priority value is used when calculating the VSRP priority. Figure 8.11 shows an example.

Figure 8.11 Track port priority



In Figure 8.11, the track port is up. SInce the port is up, the track priority does not affect the VSRP priority calculation. If the track port goes down, the track priority does affect VSRP priority calculation, as shown in Figure 8.12.

Figure 8.12 Track port priority subtracted during priority calculation



MAC Address Failover on VSRP-Aware Devices

VSRP-aware devices maintain a record of each VRID and its VLAN. When the device has received a Hello message for a VRID in a given VLAN, the device creates a record for that VRID and VLAN and includes the port number in the record. Each subsequent time the device receives a Hello message for the same VRID and VLAN, the device checks the port number.

- If the port number is the same as the port that previously received a Hello message, the VSRP-aware device assumes that the message came from the same VSRP Master that sent the previous message.
- If the port number does not match, the VSRP-aware device assumes that a VSRP failover has occurred to a new Master, and moves the MAC addresses learned on the previous port to the new port.

The VRID records age out if unused. This can occur if the VSRP-aware device becomes disconnected from the Master. The VSRP-aware device will wait for a Hello message for the period of time equal to the following:

VRID Age = Dead Interval + Hold-down Interval + (3 x Hello Interval)

The values for these timers are determined by the VSRP device sending the Hello messages. If the Master uses the default timer values, the age time for VRID records on the VSRP-aware devices is as follows:

3 + 2 + (3 x 1) = 8 seconds

In this case, if the VSRP-aware device does not receive a new Hello message for a VRID in a given VLAN, on any port, the device assumes the connection to the Master is unavailable and removes the VRID record.

Timer Scale

The VSRP Hello interval, Dead interval, Backup Hello interval, and Hold-down interval timers are individually configurable. You also can easily change all the timers at the same time while preserving the ratios among their values. To do so, change the timer scale. The ***timer scale*** is a value used by the software to calculate the timers. The software divides a timer's value by the timer scale value. By default, the scale is 1. This means the VSRP timer values are the same as the values in the configuration.

VSRP-Aware Security Features

Without VSRP-aware security configured, a VSRP-aware device passively learns the authentication method conveyed by the received VSRP hello packet. The VSRP-aware device then stores the authentication method until it ages out with the aware entry.

With VSRP-aware security, you can:

- Define the specific authentication parameters that a VSRP-aware device will use on a VSRP backup switch. The authentication parameters that you define will not age out.
- Define a list of ports that have authentic VSRP backup switch connections. For ports included in the list, the VSRP-aware switch will process VSRP hello packets using the VSRP-aware security configuration. Conversely, for ports not included in the list, the VSRP-aware switch will not use the VSRP-aware security configuration.

If VSRP hello packets do not meet the acceptance criteria, the VSRP-aware device forwards the packets normally, without any VSRP-aware security processing.

VSRP Parameters

Table 8.5 lists the VSRP parameters.

Table 8.5: VSRP Parameters

Parameter	Description	Default	See page...
Protocol	VSRP state **Note**: On a Layer 3 Switch, you must disable VSRP to use VRRPE or VRRP.	Enabled	8-28
Virtual Router ID (VRID)	The ID of the virtual switch you are creating by configuring multiple devices as redundant links. You must configure the same VRID on each device that you want to use to back up the links.	None	8-27
Timer scale	The value used by the software to calculate all VSRP timers. Increasing the timer scale value decreases the length of all the VSRP timers equally, without changing the ratio of one timer to another.	1	8-28
Interface Parameters			
Authentication type	The type of authentication the VSRP devices use to validate VSRP packets. On Layer 3 Switches, the authentication type must match the authentication type the VRID's port uses with other routing protocols such as OSPF. • No authentication – The interfaces do not use authentication. This is the VRRP default. • Simple – The interface uses a simple text-string as a password in packets sent on the interface. If the interface uses simple password authentication, the VRID configured on the interface must use the same authentication type and the same password. **Note**: MD5 is not supported.	No authentication	8-29

Table 8.5: VSRP Parameters (Continued)

Parameter	Description	Default	See page...
VSRP-Aware Security Parameters			
VSRP-Aware Authentication type	The type of authentication the VSRP-aware devices will use on a VSRP backup switch. • No authentication – The device does not accept incoming packets that have authentication strings. • Simple – The device uses a simple text-string as the authentication string for accepting incoming packets.	Not configured	8-29
VRID Parameters			
VSRP device type	Whether the device is a VSRP Backup for the VRID. All VSRP devices for a given VRID are Backups.	Not configured	8-27
VSRP ports	The ports in the VRID's VLAN that you want to use as VRID interfaces. You can selectively exclude individual ports from VSRP while allowing them to remain in the VLAN.	All ports in the VRID's VLAN	8-30
VRID IP address	A gateway address you are backing up. Configuring an IP address provides VRRPE Layer 3 redundancy in addition to VSRP LAyer 2 redundancy. The VRID IP address must be in the same subnet as a real IP address configured on the VSRP interface, but cannot be the same as a real IP address configured on the interface. **Note**: This parameter is valid only on Layer 3 Switches.	None	8-30
Backup priority	A numeric value that determines a Backup's preferability for becoming the Master for the VRID. During negotiation, the device with the highest priority becomes the Master. In VSRP, all devices are Backups and have the same priority by default. If two or more Backups are tied with the highest priority, the Backup with the highest IP address becomes the Master for the VRID.	100 for all Backups	8-30
Preference of timer source	When you save a Backup's configuration, the software can save the configured VSRP timer values or the VSRP timer values received from the Master. Saving the current timer values instead of the configured ones helps ensure consistent timer usage for all the VRID's devices. **Note**: The Backup always gets its timer scale value from the Master.	Configured timer values are saved	8-31

Table 8.5: VSRP Parameters (Continued)

Parameter	Description	Default	See page...
Time-to-Live (TTL)	The maximum number of hops a VSRP Hello packet can traverse before being dropped. You can specify from 1 – 255.	2	8-31
Hello interval	The amount of time between Hello messages from the Master to the Backups for a given VRID. The interval can be from 1 – 84 seconds.	One second	8-32
Dead interval	The amount of time a Backup waits for a Hello message from the Master for the VRID before determining that the Master is no longer active. If the Master does not send a Hello message before the dead interval expires, the Backups negotiate (compare priorities) to select a new Master for the VRID.	Three times the Hello Interval	8-32
Backup Hello state and interval	The amount of time between Hello messages from a Backup to the Master. The message interval can be from 60 – 3600 seconds. You must enable the Backup to send the messages. The messages are disabled by default on Backups. The current Master sends Hello messages by default.	Disabled 60 seconds when enabled	8-32
Hold-down interval	The amount of time a Backup that has sent a Hello packet announcing its intent to become Master waits before beginning to forward traffic for the VRID. The hold-down interval prevents Layer 2 loops from occurring during VSRP's rapid failover. The interval can from 1 – 84 seconds.	2 seconds	8-32
Track priority	A VSRP priority value assigned to the tracked port(s). If a tracked port's link goes down, the VRID port's VSRP priority is reduced by the amount of the tracked port's priority.	5	8-33
Track port	A track port is a port or virtual routing interface that is outside the VRID but whose link state is tracked by the VRID. Typically, the tracked interface represents the other side of VRID traffic flow through the device. If the link for a tracked interface goes down, the VSRP priority of the VRID interface is changed, causing the devices to renegotiate for Master.	None	8-33
Backup preempt mode	Prevents a Backup with a higher VSRP priority from taking control of the VRID from another Backup that has a lower priority but has already assumed control of the VRID.	Enabled	8-33
VRID active state	The active state of the VSRP VRID.	Disabled	8-27

Table 8.5: VSRP Parameters (Continued)

Parameter	Description	Default	See page...
RIP Parameters			
Suppression of RIP advertisements	A Layer 3 Switch that is running RIP normally advertises routes to a backed up VRID even when the Layer 3 Switch is not currently the active Layer 3 Switch for the VRID. Suppression of these advertisements helps ensure that other Layer 3 Switches do not receive invalid route paths for the VRID. **Note**: This parameter is valid only on Layer 3 Switches.	Disabled (routes are advertised)	8-34

Configuring Basic VSRP Parameters

To configure VSRP, perform the following required tasks:

- Configure a port-based VLAN containing the ports for which you want to provide VSRP service.

  NOTE: If you already have a port-based VLAN but only want to use VSRP on a sub-set of the VLANs ports, you can selectively remove ports from VSRP service in the VLAN. See “Removing a Port from the VRID’s VLAN” on page 8-30.

- Configure a VRID.
  - Specify that the device is a backup. Since VSRP, like VRRPE, does not have an “owner”, all VSRP devices are backups. The active device for a VRID is elected based on the VRID priority, which is configurable.
  - Activate the VRID.

The following example shows a simple VSRP configuration.

```
FastIron SuperX Router(config)# vlan 200
FastIron SuperX Router(config-vlan-200)# tag ethernet 1/1 to 1/8
FastIron SuperX Router(config-vlan-200)# vsrp vrid 1
FastIron SuperX Router(config-vlan-200-vrid-1)# backup
FastIron SuperX Router(config-vlan-200-vrid-1)# activate
```

Syntax: [no] vsrp vrid <num>

The <num> parameter specifies the VRID and can be from 1 – 255.

Syntax: [no] backup [priority <value>] [track-priority <value>]

This command is required. In VSRP, all devices on which a VRID are configured are Backups. The Master is then elected based on the VSRP priority of each device. There is no “owner” device as there is in VRRP.

For information about the command’s optional parameters, see the following:

- “Changing the Backup Priority” on page 8-30
- “Changing the Default Track Priority” on page 8-33

Syntax: [no] activate

or

Syntax: enable | disable

Configuring Optional VSRP Parameters

The following sections describe how to configure optional VSRP parameters.

Disabling or Re-Enabling VSRP

VSRP is enabled by default on Layer 2 Switches and Layer 3 Switches. On a Layer 3 Switch, if you want to use VRRP or VRRPE for Layer 3 redundancy instead of VSRP, you need to disable VSRP first. To do so, enter the following command at the global CONFIG level:

```
FastIron SuperX Router(config)# no router vsrp
router vsrp is disabled. All vsrp config data will be lost when writing to flash
```

To re-enable the protocol, enter the following command:

```
FastIron SuperX Router(config)# router vsrp
```

Syntax: [no] router vsrp

Since VRRP and VRRPE do not apply to Layer 2 Switches, there is no need to disable VSRP and there is no command to do so. The protocol is always enabled.

Changing the Timer Scale

To achieve sub-second failover times, you can shorten the duration of all VSRP timers by adjusting the timer scale. The ***timer scale*** is a value used by the software to calculate the timers. By default, the scale value is 1. If you increase the timer scale, each timer's value is divided by the scale value. Using the timer scale to adjust VSRP timer values enables you to easily change all the timers while preserving the ratios among their values. Here is an example.

Timer	Timer Scale	Timer Value
Hello interval	1	1 second
	2	0.5 seconds
Dead interval	1	3 seconds
	2	1.5 seconds
Backup Hello interval	1	60 seconds
	2	30 seconds
Hold-down interval	1	2 seconds
	2	1 second

If you configure the device to receive its timer values from the Master, the Backup also receives the timer scale value from the Master.

NOTE: The Backups always use the value of the timer scale received from the Master, regardless of whether the timer values that are saved in the configuration are the values configured on the Backup or the values received from the Master.

To change the timer scale, enter a command such as the following at the global CONFIG level of the CLI:

```
FastIron SuperX Router(config)# scale-timer 2
```

This command changes the scale to 2. All VSRP timer values will be divided by 2.

Syntax: [no] scale-timer <num>

The <num> parameter specifies the multiplier. You can specify a timer scale from 1 – 10.

Configuring Authentication

If the interfaces on which you configure the VRID use authentication, the VSRP packets on those interfaces also must use the same authentication. VSRP supports the following authentication types:

- No authentication – The interfaces do not use authentication. This is the default.
- Simple – The interfaces use a simple text-string as a password in packets sent on the interface. If the interfaces use simple password authentication, the VRID configured on the interfaces must use the same authentication type and the same password.

To configure a simple password, enter a command such as the following at the interface configuration level:

```
FastIron SuperX Router(config-if-1/6)# ip vsrp auth-type simple-text-auth ourpword
```

This command configures the simple text password “ourpword”.

Syntax: [no] ip vsrp auth-type no-auth | simple-text-auth <auth-data>

The **auth-type no-auth** parameter indicates that the VRID and the interface it is configured on do not use authentication.

The **auth-type simple-text-auth** <auth-data> parameter indicates that the VRID and the interface it is configured on use a simple text password for authentication. The <auth-data> value is the password. If you use this parameter, make sure all interfaces on all the devices supporting this VRID are configured for simple password authentication and use the same password.

Configuring Security Features on a VSRP-Aware Device

The VSRP-aware security feature enables you to:

- Define the specific authentication parameters that a VSRP-aware device will use on a VSRP backup switch. The authentication parameters that you define will not age out.
- Define a list of ports that have authentic VSRP backup switch connections. For ports included in the list, the VSRP-aware switch will process VSRP hello packets using the VSRP-aware security configuration. Conversely, for ports not included in the list, the VSRP-aware switch will not use the VSRP-aware security configuration.

If VSRP hello packets do not meet the acceptance criteria, the VSRP-aware device forwards the packets normally, without any VSRP-aware security processing.

Specifying an Authentication String for VSRP Hello Packets

The following configuration defines **pri-key** as the authentication string for accepting incoming VSRP hello packets. In this example, the VSRP-aware device will accept all incoming packets that have this authorization string.

```
FastIron SuperX Router(config)# vlan 10
FastIron SuperX Router(config-vlan-10)# vsrp-aware vrid 3 simple-text-auth pri-key
```

Syntax: vsrp-aware vrid <vrid number> simple text auth <string>

Specifying no Authentication for VSRP Hello Packets

The following configuration specifies no authentication as the preferred VSRP-aware security method. In this case, the VSRP device will not accept incoming packets that have authentication strings.

```
FastIron SuperX Router(config)# vlan 10
FastIron SuperX Router(config-vlan-10)# vsrp-aware vrid 2 no-auth
```

Syntax: vsrp-aware vrid <vrid number> no-auth

The following configuration specifies no authentication for VSRP hello packets received on ports 1/1, 1/2, 1/3, and 1/4 in VRID 4. For these ports, the VSRP device will not accept incoming packets that have authentication strings.

```
FastIron SuperX Router(config)# vlan 10
FastIron SuperX Router(config-vlan-10)# vsrp-aware vrid 4 no-auth port-list ethe 1/
1 to 1/4
```

Syntax: vsrp-aware vrid <vrid number> no-auth port-list <port range>

<vrid number> is a valid VRID (from 1 to 255).

no-auth specifies no authentication as the preferred VSRP-aware security method. The VSRP device will not accept incoming packets that have authentication strings.

simple-text-auth <string> specifies the authentication string for accepting VSRP hello packets, where <string> can be up to 8 characters.

port-list <port range> specifies the range of ports to include in the configuration.

Removing a Port from the VRID's VLAN

By default, all the ports in the VLAN on which you configure a VRID are interfaces for the VRID. You can remove a port from the VRID while allowing it to remain in the VLAN.

Removing a port is useful in the following cases:

- There is no risk of a loop occurring, such as when the port is attached directly to an end host.
- You plan to use a port in an MRP ring.

To remove a port from a VRID, enter a command such as the following at the configuration level for the VRID:

```
FastIron SuperX Router(config-vlan-200-vrid-1)# no include-port ethernet 1/2
```

Syntax: [no] include-port ethernet [<slotnum>/]<portnum>

The <slotnum> parameter is required on chassis devices.

The <portnum> parameter specifies the port you are removing from the VRID. The port remains in the VLAN but its forwarding state is not controlled by VSRP. If you are configuring a chassis device, specify the slot number as well as the port number (<slotnum>/<portnum>).

Configuring a VRID IP Address

If you are configuring a Layer 3 Switch for VSRP, you can specify an IP address to back up. When you specify an IP address, VSRP provides redundancy for the address. This is useful if you want to back up the gateway address used by hosts attached to the VSRP Backups.

VSRP does not require you to specify an IP address. If you do not specify an address, VSRP provides Layer 2 redundancy. If you do specify an address, VSRP provides Layer 2 and Layer 3 redundancy.

The Layer 3 redundancy support is the same as VRRPE support. For information, see the chapter "Configuring VRRP and VRRPE" on page 22-1.

NOTE: The VRID IP address must be in the same subnet as a real IP address configured on the VSRP interface, but cannot be the same as a real IP address configured on the interface.

NOTE: Failover applies to both Layer 2 and Layer 3.

To specify an IP address to back up, enter a command such as the following at the configuration level for the VRID:

```
FastIron SuperX Router(config-vlan-200-vrid-1)# ip-address 10.10.10.1
```

Syntax: [no] ip-address <ip-addr>

or

Syntax: [no] ip address <ip-addr>

Changing the Backup Priority

When you enter the backup command to configure the device as a VSRP Backup for the VRID, you also can change the backup priority and the track priority.

- The backup priority is used for election of the Master. The VSRP Backup with the highest priority value for the VRID is elected as the Master for that VRID. The default priority is 100. If two or more Backups are tied with

the highest priority, the Backup with the highest IP address becomes the Master for the VRID.

- The track priority is used with the track port feature. See "VSRP Priority Calculation" on page 8-20 and "Changing the Default Track Priority" on page 8-33.

To change the backup priority, enter a command such as the following at the configuration level for the VRID:

```
FastIron SuperX Router(config-vlan-200-vrid-1)# backup priority 75
```

Syntax: [no] backup [priority <value>] [track-priority <value>]

The **priority** <value> parameter specifies the VRRP priority for this interface and VRID. You can specify a value from 3 – 254. The default is 100.

For a description of the **track-priority** <value> parameter, see "Changing the Default Track Priority" on page 8-33.

Saving the Timer Values Received from the Master

The Hello messages sent by a VRID's master contain the VRID values for the following VSRP timers:

- Hello interval
- Dead interval
- Backup Hello interval
- Hold-down interval

By default, each Backup saves the configured timer values to its startup-config file when you save the device's configuration.

You can configure a Backup to instead save the current timer values received from the Master when you save the configuration. Saving the current timer values instead of the configured ones helps ensure consistent timer usage for all the VRID's devices.

NOTE: The Backups always use the value of the timer scale received from the Master, regardless of whether the timer values that are saved in the configuration are the values configured on the Backup or the values received from the Master.

To configure a Backup to save the VSRP timer values received from the Master instead of the timer values configured on the Backup, enter the following command:

```
FastIron SuperX Router(config-vlan-200-vrid-1)# save-current-values
```

Syntax: [no] save-current-values

Changing the Time-To-Live (TTL)

A VSRP Hello packet's TTL specifies how many hops the packet can traverse before being dropped. A hop can be a Layer 3 Switch or a Layer 2 Switch. You can specify from 1 – 255. The default TTL is 2. When a VSRP device (Master or Backup) sends a VSRP HEllo packet, the device subtracts one from the TTL. Thus, if the TTL is 2, the device that originates the Hello packet sends it out with a TTL of 1. Each subsequent device that receives the packet also subtracts one from the packet's TTL. When the packet has a TTL of 1, the receiving device subtracts 1 and then drops the packet because the TTL is zero.

NOTE: An MRP ring is considered to be a single hop, regardless of the number of nodes in the ring.

To change the TTL for a VRID, enter a command such as the following at the configuration level for the VRID:

```
FastIron SuperX Router(config-vlan-200-vrid-1)# initial-ttl 5
```

Syntax: [no] initial-ttl <num>

The <num> parameter specifies the TTL and can be from 1 – 255. The default TTL is 2.

Changing the Hello Interval

The Master periodically sends Hello messages to the Backups. To change the Hello interval, enter a command such as the following at the configuration level for the VRID:

```
FastIron SuperX Router(config-vlan-200-vrid-1)# hello-interval 10
```

Syntax: [no] hello-interval <num>

The <num> parameter specifies the interval and can be from 1 – 84 seconds. The default is 1 second.

NOTE: The default Dead interval is three times the Hello interval plus one-half second. Generally, if you change the Hello interval, you also should change the Dead interval on the Backups.

NOTE: If you change the timer scale, the change affects the actual number of seconds.

Changing the Dead Interval

The Dead interval is the number of seconds a Backup waits for a Hello message from the Master before determining that the Master is dead. The default is 3 seconds. This is three times the default Hello interval.

To change the Dead interval, enter a command such as the following at the configuration level for the VRID:

```
FastIron SuperX Router(config-vlan-200-vrid-1)# dead-interval 30
```

Syntax: [no] dead-interval <num>

The <num> parameter specifies the interval and can be from 1 – 84 seconds. The default is 3 seconds.

NOTE: If you change the timer scale, the change affects the actual number of seconds.

Changing the Backup Hello State and Interval

By default, Backups do not send Hello messages to advertise themselves to the Master. You can enable these messages if desired and also change the message interval.

To enable a Backup to send Hello messages to the Master, enter a command such as the following at the configuration level for the VRID:

```
FastIron SuperX Router(config-vlan-200-vrid-1)# advertise backup
```

Syntax: [no] advertise backup

When a Backup is enabled to send Hello messages, the Backup sends a Hello message to the Master every 60 seconds by default. You can change the interval to be up to 3600 seconds.

To change the Backup Hello interval, enter a command such as the following at the configuration level for the VRID:

```
FastIron SuperX Router(config-vlan-200-vrid-1)# backup-hello-interval 180
```

Syntax: [no] backup-hello-interval <num>

The <num> parameter specifies the message interval and can be from 60 – 3600 seconds. The default is 60 seconds.

NOTE: If you change the timer scale, the change affects the actual number of seconds.

Changing the Hold-Down Interval

The hold-down interval prevents Layer 2 loops from occurring during failover, by delaying the new Master from forwarding traffic long enough to ensure that the failed Master is really unavailable.

To change the Hold-down interval, enter a command such as the following at the configuration level for the VRID:

```
FastIron SuperX Router(config-vlan-200-vrid-1)# hold-down-interval 4
```


Syntax: [no] hold-down-interval <num>

The <num> parameter specifies the hold-down interval and can be from 1 – 84 seconds. The default is 2 seconds.

NOTE: If you change the timer scale, the change affects the actual number of seconds.

Changing the Default Track Priority

When you configure a VRID to track the link state of other interfaces, if one of the tracked interface goes down, the software changes the VSRP priority of the VRID interface.

The software reduces the VRID priority by the amount of the priority of the tracked interface that went down. For example, if the VSRP interface's priority is 100 and a tracked interface with track priority 60 goes down, the software changes the VSRP interface's priority to 40. If another tracked interface goes down, the software reduces the VRID's priority again, by the amount of the tracked interface's track priority.

The default track priority for all track ports is 1. You can change the default track priority or override the default for an individual track port.

- To change the default track priority, use the **backup track-priority** command, described below.
- To override the default track priority for a specific track port, use the **track-port** command. See "Specifying a Track Port" on page 8-33.

To change the track priority, enter a command such as the following at the configuration level for the VRID:

```
FastIron SuperX Router(config-vlan-200-vrid-1)# backup track-priority 2
```

Syntax: [no] backup [priority <value>] [track-priority <value>]

Specifying a Track Port

You can configure the VRID on one interface to track the link state of another interface on the device. This capability is useful for tracking the state of the exit interface for the path for which the VRID is providing redundancy. See "VSRP Priority Calculation" on page 8-20.

To configure a VRID to track an interface, enter a command such as the following at the configuration level for the VRID:

```
FastIron SuperX Router(config-vlan-200-vrid-1)# track-port e 2/4
```

Syntax: [no] track-port ethernet [<slotnum>/]<portnum> | ve <num> [priority <num>]

The **priority** <num> parameter changes the VSRP priority of the interface. If this interface goes down, the VRID's VSRP priority is reduced by the amount of the track port priority you specify here.

NOTE: The priority <num> option changes the priority of the specified interface, overriding the default track port priority. To change the default track port priority, use the **backup track-priority** <num> command.

Disabling or Re-Enabling Backup Pre-Emption

By default, a Backup that has a higher priority than another Backup that has become the Master can preempt the Master, and take over the role of Master. If you want to prevent this behavior, disable preemption.

Preemption applies only to Backups and takes effect only when the Master has failed and a Backup has assumed ownership of the VRID. The feature prevents a Backup with a higher priority from taking over as Master from another Backup that has a lower priority but has already become the Master of the VRID.

Preemption is especially useful for preventing flapping in situations where there are multiple Backups and a Backup with a lower priority than another Backup has assumed ownership, because the Backup with the higher priority was unavailable when ownership changed.

If you enable the non-preempt mode (thus disabling the preemption feature) on all the Backups, the Backup that becomes the Master following the disappearance of the Master continues to be the Master. The new Master is not preempted.

To disable preemption on a Backup, enter a command such as the following at the configuration level for the VRID:

```
FastIron SuperX Router(config-vlan-200-vrid-1)# non-preempt-mode
```

Syntax: [no] non-preempt-mode

Suppressing RIP Advertisement from Backups

Normally, for Layer 3 a VSRP Backup includes route information for a backed up IP address in RIP advertisements. As a result, other Layer 3 Switches receive multiple paths for the backed up interface and might sometimes unsuccessfully use the path to the Backup rather than the path to the Master.

You can prevent the Backups from advertising route information for the backed up interface by enabling suppression of the advertisements.

NOTE: This parameter applies only if you specified an IP address to back up and is valid only on Layer 3 Switches.

To suppress RIP advertisements, enter the following commands:

```
Router2(config)# router rip
Router2(config-rip-router)# use-vrrp-path
```

Syntax: [no] use-vrrp-path

Displaying VSRP Information

You can display the following VSRP information:

- Configuration information and current parameter values for a VRID or VLAN
- The interfaces on a VSRP-aware device that are active for the VRID

Displaying VRID Information

To display VSRP information, enter the following command:

```
FastIron SuperX Router(config-vlan-200-vrid-1)# show vsrp vrid 1
Total number of VSRP routers defined: 2
VLAN 200
 auth-type no authentication
 VRID 1
  State        Administrative-status Advertise-backup Preempt-mode save-current
  standby      enabled               disabled         true         false

  Parameter        Configured Current    Unit
  priority               100     80     (100-0)*(4.0/5.0)
  hello-interval           1       1    sec/1
  dead-interval            3       3    sec/1
  hold-interval            3       3    sec/1
  initial-ttl              2       2    hops

  next hello sent in 00:00:00.8
  Member ports:      ethe 1/1 to 1/5
  Operational ports: ethe 1/1 to 1/4
  Forwarding ports:  ethe 1/1 to 1/4
```

Syntax: show vsrp [vrid <num> | vlan <vlan-id>]

This display shows the following information when you use the **vrid** <num> or **vlan** <vlan-id> parameter. For information about the display when you use the **aware** parameter, see “Displaying the Active Interfaces for a VRID” on page 8-37.

Table 8.6: CLI Display of VSRP VRID or VLAN Information

This Field...	Displays...
Total number of VSRP routers defined	The total number of VRIDs configured on this device.
VLAN	The VLAN on which VSRP is configured.
auth-type	The authentication type in effect on the ports in the VSRP VLAN.
VRID parameters	
VRID	The VRID for which the following information is displayed.
state	This device’s VSRP state for the VRID. The state can be one of the following: • initialize – The VRID is not enabled (activated). If the state remains “initialize” after you activate the VRID, make sure that the VRID is also configured on the other routers and that the routers can communicate with each other. **Note**: If the state is “initialize” and the mode is incomplete, make sure you have specified the IP address for the VRID. • standby – This device is a Backup for the VRID. • master – This device is the Master for the VRID.
Administrative-status	The administrative status of the VRID. The administrative status can be one of the following: • disabled – The VRID is configured on the interface but VSRP or VRRPE has not been activated on the interface. • enabled – VSRP has been activated on the interface.
Advertise-backup	Whether the device is enabled to send VSRP Hello messages when it is a Backup. This field can have one of the following values: • disabled – The device does not send Hello messages when it is a Backup. • enabled – The device does send Hello messages when it is a Backup.
Preempt-mode	Whether the device can be pre-empted by a device with a higher VSRP priority after this device becomes the Master. This field can have one of the following values: • disabled – The device cannot be pre-empted. • enabled – The device can be pre-empted.

Table 8.6: CLI Display of VSRP VRID or VLAN Information (Continued)

This Field...	Displays...
save-current	The source of VSRP timer values preferred when you save the configuration. This field can have one of the following values: • false – The timer values configured on this device are saved. • true – The timer values most recently received from the Master are saved instead of the locally configured values.
Note: For the following fields: • Configured – indicates the parameter value configured on this device. • Current – indicates the parameter value received from the Master. • Unit – indicates the formula used tor calculating the VSRP priority and the timer scales in effect for the VSRP timers. A timer's true value is the value listed in the Configured or Current field divided by the scale value.	
priority	The device's preferability for becoming the Master for the VRID. During negotiation, the Backup with the highest priority becomes the Master. If two or more Backups are tied with the highest priority, the Backup interface with the highest IP address becomes the Master for the VRID.
hello-interval	The number of seconds between Hello messages from the Master to the Backups for a given VRID.
dead-interval	The configured value for the dead interval. The dead interval is the number of seconds a Backup waits for a Hello message from the Master for the VRID before determining that the Master is no longer active. If the Master does not send a Hello message before the dead interval expires, the Backups negotiate (compare priorities) to select a new Master for the VRID. **Note**: If the value is 0, then you have not configured this parameter.
hold-interval	The number of seconds a Backup that intends to become the Master will wait before actually beginning to forward Layer 2 traffic for the VRID. If the Backup receives a Hello message with a higher priority than its own before the hold-down interval expires, the Backup remains in the Backup state and does not become the new Master.
initial-ttl	The number of hops a Hello message can traverse after leaving the device before the Hello message is dropped. **Note**: An MRP ring counts as one hop, regardless of the number of nodes in the ring.
next hello sent in	The amount of time until the Master's dead interval expires. If the Backup does not receive a Hello message from the Master by the time the interval expires, either the IP address listed for the Master will change to the IP address of the new Master, or this Layer 3 Switch itself will become the Master. **Note**: This field applies only when this device is a Backup.

Table 8.6: CLI Display of VSRP VRID or VLAN Information (Continued)

This Field...	Displays...
Member ports	The ports in the VRID.
Operational ports	The member ports that are currently up.
Forwarding ports	The member ports that are currently in the Forwarding state. Ports that are forwarding on the Master are listed. Ports on the Standby, which are in the Blocking state, are not listed.

Displaying the Active Interfaces for a VRID

On a VSRP-aware device, you can display VLAN and port information for the connections to the VSRP devices (Master and Backups).

To display the active VRID interfaces, enter the following command on the VSRP-aware device:

```
FastIron SuperX Router(config-vlan-200-vrid-1)# show vsrp aware

Aware port listing
VLAN ID  VRID  Last Port
100      1     3/2
200      2     4/1
```

Syntax: show vsrp aware

This display shows the following information when you use the **aware** parameter. For information about the display when you use the **vrid** <num> or **vlan** <vlan-id> parameter, see “Displaying VRID Information” on page 8-34.

Table 8.7: CLI Display of VSRP-Aware Information

This Field...	Displays...
VLAN ID	The VLAN that contains the VSRP-aware device’s connection with the VSRP Master and Backups.
VRID	The VRID.
Last Port	The most recent active port connection to the VRID. This is the port connected to the current Master. If a failover occurs, the VSRP-aware device changes the port to the port connected to the new Master. The VSRP-aware device uses this port to send and receive data through the backed up node.

VSRP Fast Start

Software releases 02.4.00 and later support the VSRP fast start feature. This feature allows non-Foundry or non-VSRP aware devices that are connected to a Foundry device that is the VSRP Master to quickly switchover to the new Master when a VSRP failover occurs

This feature causes the port on a VSRP Master to restart when a VSRP failover occurs. When the port shuts down at the start of the restart, ports on the non-VSRP aware devices that are connected to the VSRP Master flush the MAC address they have learned for the VSRP master. After a specified time, the port on the previous VSRP Master (which now becomes the Backup) returns back online. Ports on the non-VSRP aware devices switch over to the new Master and learn its MAC address.

Configuring VSRP Fast Start

The VSRP fast start feature can be enabled on a VSRP-configured Foundry device, either on the VLAN to which the VRID of the VSRP-configured device belongs (globally) or on a port that belongs to the VRID.

To globally configure a VSRP-configured device to shut down its ports when a failover occurs, then restart after five seconds, enter the following command:

```
FastIron SuperX Switch(configure)# vlan 100
FastIron SuperX Switch(configure-vlan-100)# vsrp vrid 1
FastIron SuperX Switch(configure-vlan-100-vrid-1)# restart-ports 5
```

Syntax: [no] restart-ports <seconds>

This command shuts down all the ports that belong to the VLAN when a failover occurs. All the ports will have the specified VRID.

To configure a single port on a VSRP-configured device to shut down when a failover occurs, then restart after a period of time, enter the following command:

```
FastIron SuperX Switch(configure)# interface ethernet 1/1
FastIron SuperX Switch(configure-if-1/1)# vsrp restart-port 5
```

Syntax: [no] vsrp restart-port <seconds>

In both commands, the <seconds> parameter instructs the VSRP Master to shut down its port for the specified number of seconds before it starts back up. Enter a value between 1 – 120 seconds. The default is 1 second.

Displaying Ports that Have the VSRP Fast Start Feature Enabled

The **show vsrp vrid** command shows the ports on which the VSRP fast start feature is enabled.

```
FastIron SuperX Switch(config-vlan-100-vrid-100)#show vsrp vrid 100

VLAN 100
  auth-type no authentication
  VRID 100
  ========
   State      Administrative-status Advertise-backup Preempt-mode save-current
   master     enabled               disabled         true         false
   Parameter        Configured Current    Unit/Formula
   priority         100        50         (100-0)*(2.0/4.0)
   hello-interval   1          1          sec/1
   dead-interval    3          3          sec/1
   hold-interval    3          3          sec/1
   initial-ttl      2          2          hops

   next hello sent in 00:00:00.3
   Member ports:      ethe 2/5 to 2/8
   Operational ports: ethe 2/5 ethe 2/8
   Forwarding ports:  ethe 2/5 ethe 2/8
   Restart ports:     2/5(1) 2/6(1) 2/7(1) 2/8(1)
```

The "Restart ports:" line lists the ports that have the VSRP fast start enabled, and the downtime for each port. See Table 8.6 on page 8-35 to interpret the remaining information on the display.

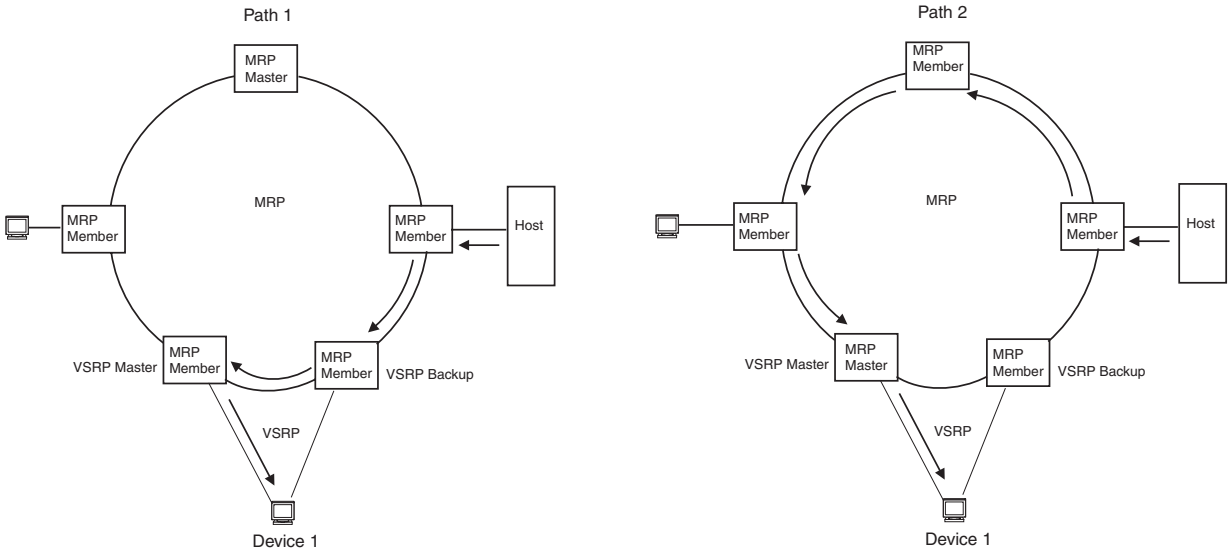
VSRP and MRP Signaling

NOTE: This feature is supported in software releases 02.4.00 and later.

A device may connect to an MRP ring via VSRP to provide a redundant path between the device and the MRP ring. VSRP and MRP signaling ensures rapid failover by flushing MAC addresses appropriately. The host on the

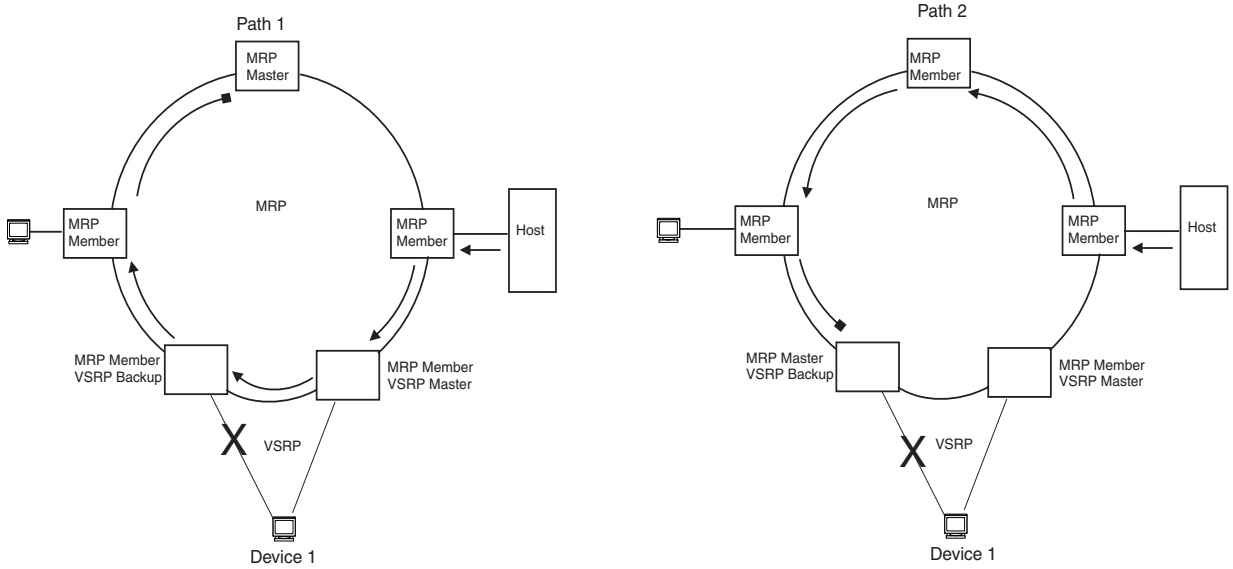
MRP ring learns the MAC addresses of all devices on the MRP ring and VSRP link. From these MAC addresses, the host creates a MAC database (table), which is used to establish a data path from the host to a VSRP-linked device. Figure 8.13 below shows two possible data paths from the host to Device 1.

Figure 8.13 Two data paths from host on an MRP ring to a VSRP-linked device



If a VSRP failover from master to backup occurs, VSRP needs to inform MRP of the topology change; otherwise, data from the host continues along the obsolete learned path and never reach the VSRP-linked device, as shown in Figure 8.14.

Figure 8.14 VSRP on MRP rings that failed over



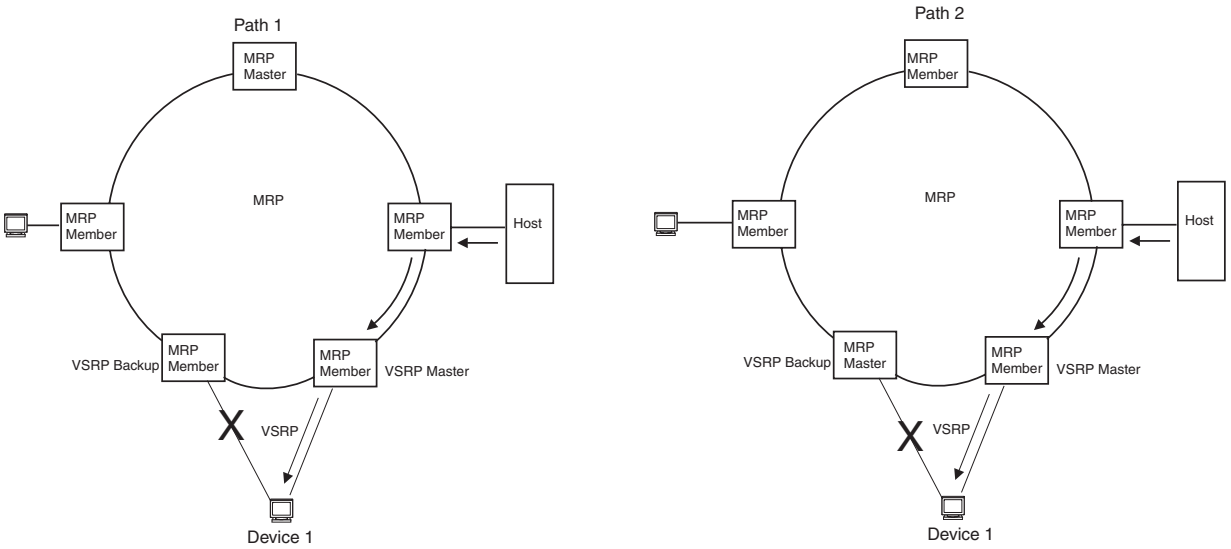
To ensure that MRP is informed of the topology change and to achieve convergence rapidly, this release provides a new signaling process for the interaction between VSRP and MRP. When a VSRP node fails, a new VSRP master is selected. The new VSRP master finds all MRP instances impacted by the failover. Then each MRP instance does the following:

- The MRP node sends out an MRP PDU with the mac-flush flag set three times on the MRP ring.

- The MRP node that receives this MRP PDU empties all the MAC entries from its interfaces that participate on the MRP ring.
- The MRP node then forwards the MRP PDU with the mac-flush flag set to the next MRP node that is in forwarding state.

The process continues until the Master MRP node's secondary (blocking) interface blocks the packet. Once the MAC address entries have been flushed, the MAC table can be rebuilt for the new path from the host to the VSRP-linked device (Figure 8.15).

Figure 8.15 New path established



There are no used CLI commands to configure this process.

Chapter 9
Configuring Uni-Directional Link Detection (UDLD)

This chapter describes how to configure Uni-directional Link Detection (UDLD) on a Foundry FastIron switch using the CLI.

This chapter contains the topics listed in Table 9.1.

Table 9.1: Chapter Contents

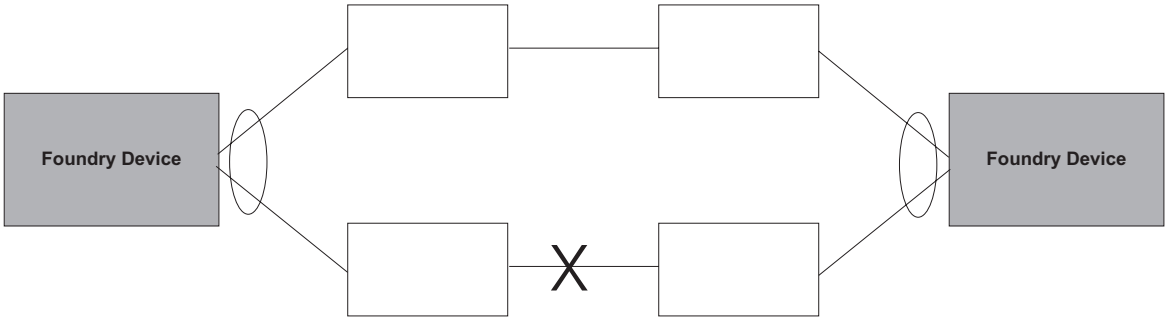
UDLD Overview

Uni-directional Link Detection (UDLD) monitors a link between two Foundry devices and brings the ports on both ends of the link down if the link goes down at any point between the two devices. This feature is useful for links that are individual ports and for trunk links. Figure 9.1 shows an example.

Figure 9.1 UDLD example

Without link keepalive, the Foundry ports remain enabled. Traffic continues to be load balanced to the ports connected to the failed link.

When link keepalive is enabled, the feature brings down the Foundry ports connected to the failed link.



Normally, a Foundry device load balances traffic across the ports in a trunk group. In this example, each Foundry device load balances traffic across two ports. Without the UDLD feature, a link failure on a link that is not directly attached to one of the Foundry devices is undetected by the Foundry devices. As a result, the Foundry devices continue to send traffic on the ports connected to the failed link.

When UDLD is enabled on the trunk ports on each Foundry device, the devices detect the failed link, disable the ports connected to the failed link, and use the remaining ports in the trunk group to forward the traffic.

Ports enabled for UDLD exchange proprietary health-check packets once every second (the keepalive interval). If a port does not receive a health-check packet from the port at the other end of the link within the keepalive interval, the port waits for two more intervals. If the port still does not receive a health-check packet after waiting for three intervals, the port concludes that the link has failed and takes the port down.

Configuration Considerations

- This feature is supported only on Ethernet ports.
- To configure UDLD on a trunk group, you must enable and configure the feature on each port of the group individually. Configuring UDLD on a trunk group's primary port enables the feature on that port only.
- Dynamic trunking is not supported. If you want to configure a trunk group that contains ports on which UDLD is enabled, you must remove the UDLD configuration from the ports. After you create the trunk group, you can re-add the UDLD configuration.

Enabling UDLD

To enable UDLD on a port, enter a command such as the following at the global CONFIG level of the CLI:

```
FastIron SuperX Router(config)# link-keepalive ethernet 1/1
```

To enable the feature on a trunk group, enter commands such as the following:

```
FastIron SuperX Router(config)# link-keepalive ethernet 1/1 ethernet 1/2
FastIron SuperX Router(config)# link-keepalive ethernet 1/3 ethernet 1/4
```

These commands enable UDLD on ports 1/1 – 1/4. You can specify up to two ports on the same command line.

Syntax: [no] link-keepalive ethernet [<slotnum>/]<portnum> [ethernet [<slotnum>/]<portnum>]

The <slotnum> parameter is required on chassis devices.

Changing the Keepalive Interval

By default, ports enabled for UDLD send a link health-check packet once every 500 ms. You can change the interval to a value from 1 – 60, where 1 is 100 ms, 2 is 200 ms, and so on. To change the interval, enter a command such as the following:

```
FastIron SuperX Router(config)# link-keepalive interval 3
```

Syntax: [no] link-keepalive interval <num>

The <num> parameter specifies how often the ports send a UDLD packet. You can specify from 1 – 60, in 100 ms increments. The default is 5 (500 ms).

Changing the Keepalive Retries

By default, a port waits one second to receive a health-check reply packet from the port at the other end of the link. If the port does not receive a reply, the port tries four more times by sending up to four more health-check packets. If the port still does not receive a reply after the maximum number of retries, the port goes down.

You can change the maximum number of keepalive attempts to a value from 3 – 10. To change the maximum number of attempts, enter a command such as the following:

```
FastIron SuperX Router(config)# link-keepalive retries 4
```

Syntax: [no] link-keepalive retries <num>

The <num> parameter specifies the maximum number of times the port will try the health check. You can specify a value from 3 – 10. The default is 5.

UDLD for Tagged Ports

The default implementation of UDLD sends the packets untagged, even across tagged ports. If the untagged UDLD packet is received by a third-party switch, that switch may reject the packet. As a result, UDLD may be limited only to Foundry devices, since UDLD may not function on third-party switches.

You can configure ports to send out UDLD control packets that are tagged with a specific VLAN ID as tagged UDLD control packets. This feature also enables third party switches to receive the control packets that are tagged with the specified VLAN.

To enable ports to receive and send UDLD control packets tagged with a specific VLAN ID, enter commands such as the following:

```
FastIron SuperX Router(config)# link-keepalive ethernet 1/18 vlan 22
```

This command enables UDLD on port 1/18 and allows UDLD control packet tagged with VLAN 22 to be received and sent on port 1/18.

Syntax: [no] link-keepalive ethernet [<slotnum>/]<portnum> [vlan <vlan-ID>]

The <slotnum> parameter is required on chassis devices.

Enter the ID of the VLAN that the UDLD control packets can contain to be received and sent on the port. If a VLAN ID is not specified, then UDLD control packets are sent out of the port as untagged packets.

NOTE: You must configure the same VLANs that will be used for UDLD on all devices across the network; otherwise, the UDLD link cannot be maintained.

Displaying UDLD Information

Displaying Information for All Ports

To display UDLD information for all ports, enter the following command:

```
FastIron SuperX Router(config)# show link-keepalive
Total link-keepalive enabled ports: 4
Keepalive Retries: 3    Keepalive Interval: 1 Sec.

Port    Physical Link   Logical Link    State
4/1     up              up              FORWARDING
4/2     up              up              FORWARDING
4/3     down            down            DISABLED
4/4     up              down            DISABLED
```

Syntax: show link-keepalive [ethernet [<slotnum>/]<portnum>]

Table 9.2: CLI Display of UDLD Information

This Field...	Displays...
Total link-keepalive enabled ports	The total number of ports on which UDLD is enabled.
Keepalive Retries	The number of times a port will attempt the health check before concluding that the link is down.
Keepalive Interval	The number of seconds between health check packets.
Port	The port number.
Physical Link	The state of the physical link. This is the link between the Foundry port and the directly connected device.
Logical Link	The state of the logical link. This is the state of the link between this Foundry port and the Foundry port on the other end of the link.
State	The traffic state of the port.

If a port is disabled by UDLD, the change also is indicated in the output of the **show interfaces brief** command. Here is an example:

```
FastIron SuperX Router(config)# show interface brief

Port  Link State      Dupl Speed Trunk Tag Priori MAC            Name
1/1   Up   LK-DISABLENone None  None  No  level0 00e0.52a9.bb00
1/2   Down None       None None  None  No  level0 00e0.52a9.bb01
1/3   Down None       None None  None  No  level0 00e0.52a9.bb02
1/4   Down None       None None  None  No  level0 00e0.52a9.bb03
```

If the port was already down before you enabled UDLD for the port, the port's state is listed as None.

Syntax: show interface brief

The **show link-keepalive** command shows the following:

```
FastIron SuperX Router(config)# show link-keepalive ethernet
Current State    : down          Remote MAC Addr   : 0000.0000.0000
Local Port       : 1/1           Remote Port       : n/a
Local System ID  : e0eb8e00      Remote System ID  : 00000000
Packets sent     : 0             Packets received  : 0
Transitions      : 0             Link-vlan         : 100
Port blocking    : No            BM disabled       : Yes
```

The Link-vlan entry shows the ID of the tagged VLAN in the UDLD packet.

Syntax: show link-keepalive ethernet

Displaying Information for a Single Port

To display detailed UDLD information for a specific port, enter a command such as the following:

```
FastIron SuperX Router(config)# show link-keepalive ethernet 4/1

Current State    : up            Remote MAC Addr   : 00e0.52d2.5100
Local Port       : 4/1           Remote Port       : 2/1
Local System ID  : e0927400      Remote System ID  : e0d25100
Packets sent     : 254           Packets received  : 255
Transitions      : 1

Port blocking    : No            BM disabled       : No
```

Table 9.3: CLI Display of Detailed UDLD Information

This Field...	Displays...
Current State	The state of the logical link. This is the link between this Foundry port and the Foundry port on the other end of the link.
Remote MAC Addr	The MAC address of the port or device at the remote end of the logical link.
Local Port	The port number on this Foundry device.
Remote Port	The port number on the Foundry device at the remote end of the link.
Local System ID	A unique value that identifies this Foundry device. The ID can be used by Foundry technical support for troubleshooting.
Remote System ID	A unique value that identifies the Foundry device at the remote end of the link.
Packets sent	The number of UDLD health-check packets sent on this port.
Packets received	The number of UDLD health-check packets received on this port.
Transitions	The number of times the logical link state has changed between up and down.
Port blocking	Information used by Foundry technical support for troubleshooting.

Table 9.3: CLI Display of Detailed UDLD Information (Continued)

This Field...	Displays...
BM disabled	Information used by Foundry technical support for troubleshooting.

The **show interface ethernet** [<slotnum>/]<portnum> command also displays the UDLD state for an individual port. In addition, the line protocol state listed in the first line will say “down” if UDLD has brought the port down. Here is an example:

```
FastIron SuperX Router(config)# show interface ethernet 1/1
FastEthernet1/1 is down, line protocol is down, link keepalive is enabled
  Hardware is FastEthernet, address is 00e0.52a9.bbca (bia 00e0.52a9.bbca)
  Configured speed auto, actual unknown, configured duplex fdx, actual unknown
  Member of L2 VLAN ID 1, port is untagged, port state is DISABLED
  STP configured to ON, priority is level0, flow control enabled
  mirror disabled, monitor disabled
  Not member of any active trunks
  Not member of any configured trunks
  No port name
  300 second input rate: 0 bits/sec, 0 packets/sec, 0.00% utilization
  300 second output rate: 0 bits/sec, 0 packets/sec, 0.00% utilization
  0 packets input, 0 bytes, 0 no buffer
  Received 0 broadcasts, 0 multicasts, 0 unicasts
  0 input errors, 0 CRC, 0 frame, 0 ignored
  0 runts, 0 giants, DMA received 0 packets
  19 packets output, 1216 bytes, 0 underruns
  Transmitted 0 broadcasts, 19 multicasts, 0 unicasts
  0 output errors, 0 collisions, DMA transmitted 19 packets
```

In this example, the port has been brought down by UDLD. Notice that in addition to the information in the first line, the port state on the fourth line of the display is listed as DISABLED.

Clearing UDLD Statistics

To clear UDLD statistics, enter the following command:

```
FastIron SuperX Router# clear link-keepalive statistics
```

Syntax: clear link-keepalive statistics

This command clears the Packets sent, Packets received, and Transitions counters in the **show link keepalive ethernet** [<slotnum>/]<portnum> display.

Chapter 10

Configuring Trunk Groups and Dynamic Link Aggregation

This chapter describes how to configure trunk groups and 802.3ad link aggregation.

- ***Trunk groups*** are manually-configured aggregate links containing multiple ports.
- ***802.3ad link aggregation*** is a protocol that dynamically creates and manages trunk groups.

NOTE: You can use both types of trunking on the same device. However, you can use only one type of trunking for a given port. For example, you can configure port 1/1 as a member of a static trunk group or you can enable 802.3ad link aggregation on the port, but you cannot do both.

This chapter contains the following information:

Table 10.1: Chapter Contents

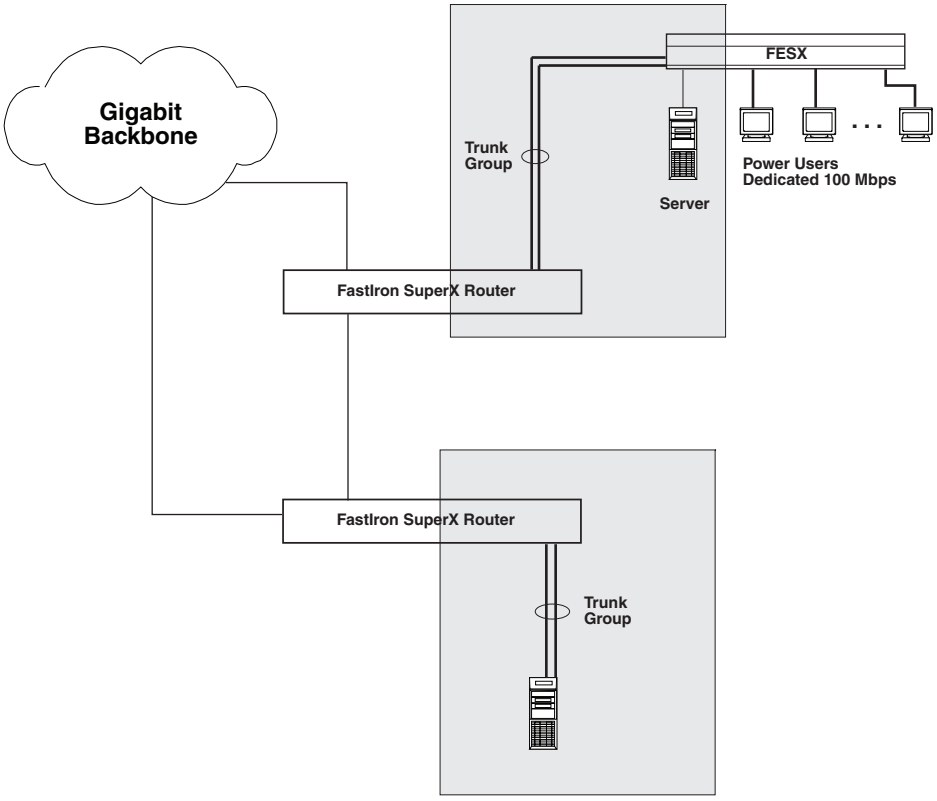
Trunk Group Overview

The Trunk Group feature allows you to manually configure multiple high-speed load-sharing links between two Foundry Layer 2 Switches or Layer 3 Switches or between a Foundry Layer 2 Switch and Layer 3 Switch and a server. You can configure up to 4 ports as a trunk group, supporting transfer rates of up to 8 Gbps of bi-directional traffic.

In addition to enabling load sharing of traffic, trunk groups provide redundant, alternate paths for traffic if any of the segments fail.

Figure 10.1 shows an example of a configuration that uses trunk groups.

Figure 10.1 Trunk Group application within a FastIron network



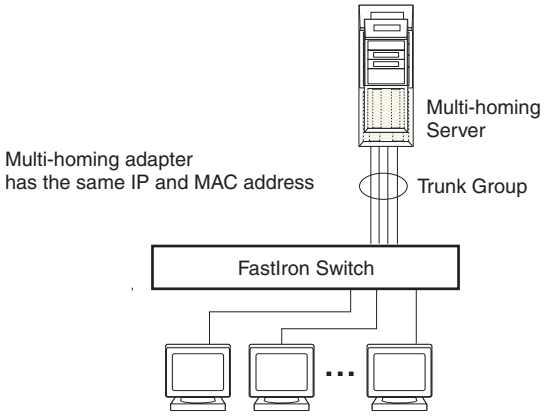
NOTE: The ports in a trunk group make a single logical link. Therefore, all the ports in a trunk group must be connected to the same device at the other end.

Trunk Group Connectivity to a Server

To support termination of a trunk group, the server must have either multiple network interface cards (NICs) or either a dual or quad interface card installed. The trunk server is designated as a server with multiple adapters or a single adapter with multiple ports that share the same MAC and IP address.

Figure 10.2 shows an example of a trunk group between a server and a Foundry device.

Figure 10.2 Trunk group between a server and a Foundry Stackable Layer 2 Switch or Layer 3 Switch



Trunk Group Rules

- Table 10.2 lists the maximum number of trunk groups you can configure on a Foundry FastIron device and the valid number of ports in a trunk group.

Table 10.2: Trunk Group Support

Model	Maximum Number of Gigabit Trunk Groups	Valid Number of Ports in a Group
FESX424 and FWSX424	13	2, 3, or 4
FESX448 and FWSX448	25	2, 3, or 4
FSX	31	2, 3, or 4

- You cannot configure a port as a member of a trunk group if 802.3ad link aggregation is enabled on the port.
- Unlike the FES and other Foundry devices, trunk groups on the FESX, FSX, and FWSX are not classified as switch trunk groups or server trunk groups.
- Table 10.2 lists the maximum number of trunk groups you can configure on a FESX, FSX, and FWSX, and the valid number of ports in a trunk group.
- Multi-slot trunk groups are supported only on FSX devices.
- Although the FESX, FSX, and FWSX devices have port ranges, they do not apply to trunk groups.
- You can select any port to be the primary port of the trunk group.
- You cannot combine Gigabit and 10-Gigabit ports in the same trunk group.
- Port assignment on a module must be contiguous. The port range on the module cannot contain gaps. For example, you can configure ports 1, 2, 3, and 4 on a module together as a trunk group but not ports 1, 3, and 4 (excluding 2).

- Make sure the device on the other end of the trunk link can support the same number of ports in the link. For example, if you configure a three-port trunk group on the FESX and the other end is a different type of switch, make sure the other switch can support a three-port trunk group.
- All the ports must be connected to the same device at the other end.
- All trunk group member properties must match the lead port of the trunk group with respect to the following parameters:
  - port tag type (untagged or tagged port)
  - statically configured port speed and duplex
  - QoS priority

  To change port parameters, you must change them on the primary port. The software automatically applies the changes to the other ports in the trunk group.

Trunk Group Configuration Examples

Figure 10.3 shows some examples of valid 2-port and 3-port trunk group links between devices. The trunk groups in this example are switch trunk groups, between two Foundry devices. Ports in a valid 2-port trunk group on one device are connected to two ports in a valid 2-port trunk group on another device. The same rules apply to 3-port, 4-port, etc., trunk groups.

Figure 10.3 Examples of 2-port and 3-port trunk groups

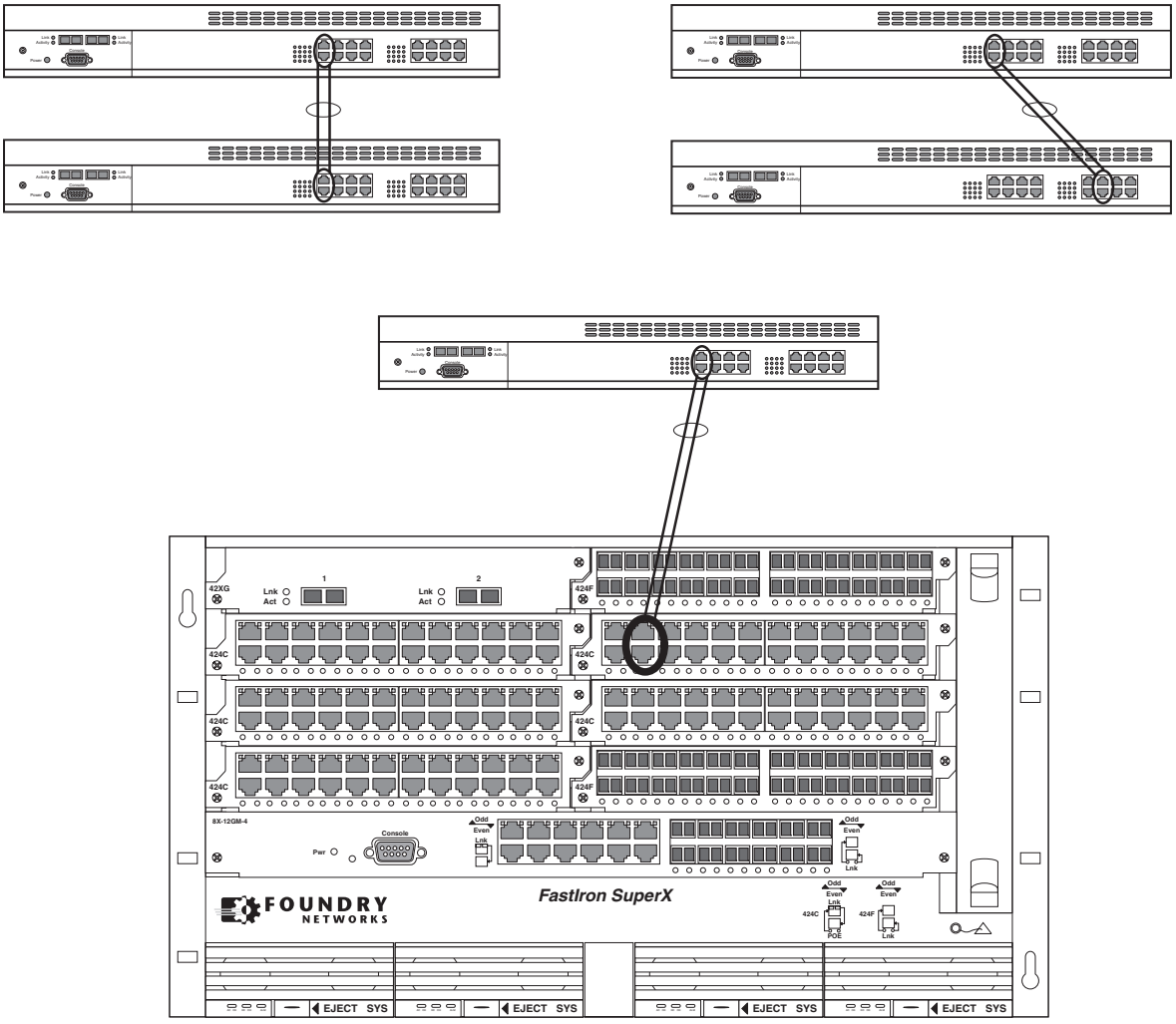
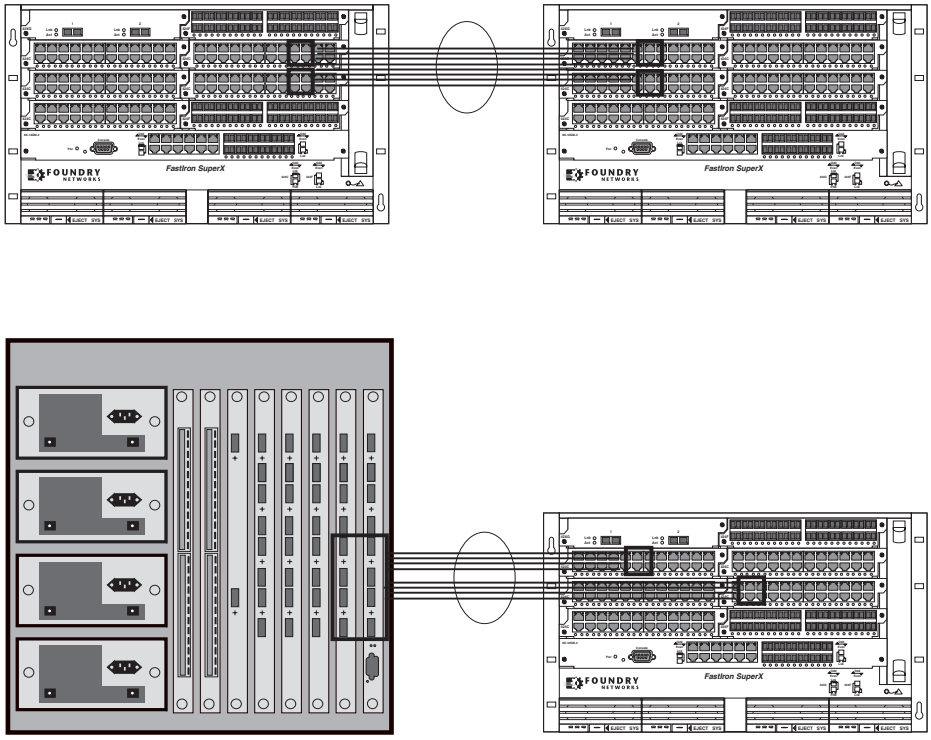


Figure 10.4 shows examples of two Chassis devices connected by multi-slot trunk groups.

Figure 10.4 Examples of multi-slot trunk groups



Trunk Group Load Sharing

Unlike the FES and other Foundry devices, trunk groups on the FESX, FSX, and FWSX devices are not classified as switch trunk groups or server trunk groups.

The Foundry device load shares across the ports in the trunk group. The method used for the load sharing depends on the following:

- Device type – Chassis device or Stackable device
- Traffic type – Layer 2 or Layer 3
- Software version your device is running

NOTE: Layer 2 and Layer 3 AppleTalk traffic is not load-balanced. Layer 3 routed IP or IPX traffic also is not load balanced. These traffic types will however still be forwarded on the trunk ports.

Note Regarding IPv6

Foundry devices that support IPv6 take a packet's IPv6 address into account when sharing traffic across a trunk group. The load sharing is performed in the same way it is for IPv4 addresses; that is; trunk types whose traffic load is shared based on IPv4 address information can now use IPv6 addresses to make the load sharing decision.

Load sharing occurs as described in Table 10.4 or Table 10.3.

How Trunk Load Sharing Works

Load balancing procedures differ depending on the software version your device is running. In releases prior to 02.2.00 for the FESX/FWSX, and release 02.1.00 for the FSX, the Foundry device load balances all bridged traffic based on source and destination MAC addresses. In subsequent releases, the load balancing method for bridged traffic varies depending on the traffic type.

Table 10.3 shows how the FESX, FWSX, and FSX load balance traffic across the ports in a trunk group, if the device is running FESX/FWSX software release 02.2.00 or later or FSX software release 02.1.00 or later. Likewise, Table 10.4 shows how a FESX and FSX load balance traffic across the ports in a trunk group, if the device is running a FESX/FWSX software release prior to 02.2.00 or a FSX software release prior to 02.1.00.

Note that load balancing on the FESX and FSX is hardware-based.

NOTE: There is no change in load balancing for routed traffic, which is always based on the source and destination IP addresses and protocol field.

.

Table 10.3: Trunk Group Load Sharing on FSX Devices (Release 02.1.00 and Later) and FESX/FWSX Devices (Release 02.2.00 and Later)

Traffic Type	Load Balancing Method
Layer 2 bridged non-IP	Source and destination MAC addresses
Layer 2 bridged TCP/UDP	Source and destination IP addresses and Source and Destination TCP/UDP ports
Layer 2 bridged IP (non-TCP/UDP)	Source and destination IP addresses
Layer 3 routed traffic	Source and destination IP addresses and protocol field

Table 10.4: Trunk Group Load Sharing on FSX Devices (Pre-Release 02.1.00) and FESX/FWSX Devices (Pre-Release 02.2.00)

Traffic Layer	Traffic Type	Load-Sharing Basis
Layer 2	All traffic types	Destination MAC address Source MAC address
Layer 3	IP	Destination IP address Source MAC address Protocol
	All other traffic types	Destination MAC address Source MAC address

Configuring a Trunk Group

To configure a trunk group, do the following:

1. Disconnect the cables from those ports on both systems that will be connected by the trunk group. Do not configure the trunk groups with the cables connected.

NOTE: If you connect the cables before configuring the trunk groups and then rebooting, the traffic on the ports can create a spanning tree loop.

2. Configure the trunk group on one of the two Layer 2 Switches or Layer 3 Switches involved in the configuration.
3. Save the configuration changes to the startup-config file.
4. Dynamically place the new trunk configuration into effect by entering the **trunk deploy** command at the global CONFIG level of the CLI.
5. If the device at the other end of the trunk group is another Layer 2 Switch or Layer 3 Switch, repeat Steps 2 – 4 for the other device.
6. When the trunk groups on both devices are operational, reconnect the cables to those ports that are now configured as trunk groups, starting with the first port (lead port) of each trunk group.
7. To verify the link is operational, use the **show trunk** command.

Example 1: Configuring the Trunk Groups Shown in Figure 10.1

To configure the trunk groups shown in Figure 10.1, enter the following commands. Notice that the commands are entered on multiple devices.

To configure the trunk group link between FSX1 and the FESX:

NOTE: The text shown in italics in the CLI example below shows messages echoed to the screen in answer to the CLI commands entered.

```
FastIron SuperX Router(config)# trunk e 1/5 to 1/8
Trunk 2 is created for next power cycle.
Please save configuration to flash and reboot.
FastIron SuperX Router(config)# write memory
Write startup-config in progress.
.Write startup-config done.
FastIron SuperX Router(config)# exit
FastIron SuperX Router# reload
```

To configure the trunk group link between FSX2 and the server:

```
FastIron SuperX Router(config)# trunk e 1/2 to 1/4
Trunk 0 is created for next power cycle.
Please save configuration to flash and reboot.
FastIron SuperX Router(config)# write memory
Write startup-config in progress.
.Write startup-config done.
FastIron SuperX Router(config)# exit
FastIron SuperX Router# reload
```

You then configure the trunk group on the FESX.

```
FESX424 Switch(config)# trunk ethernet 17 to 18
FESX424 Switch(config)# write memory
Write startup-config in progress.
.Write startup-config done.
FESX424 Switch(config)# exit
FESX424 Switch# reload
```

Example 2: Configuring a Trunk Group That Spans Multiple Gigabit Ethernet Modules in a Chassis Device

This section shows how to configure a trunk group that spans two modules in a Chassis device.

For example, to configure a trunk group consisting of two groups of ports, 1/1 – 1/4 on module 1 and 4/5 – 4/8 on module 4, enter the following commands:

```
FastIron SuperX Router(config)# trunk ethernet 1/1 to 1/4 ethernet 4/5 to 4/8
FastIron SuperX Router(config-trunk-1/1-4/8)# write memory
FastIron SuperX Router(config-trunk-1/1-4/8)# exit
FastIron SuperX Router(config)# trunk deploy
```

NOTE: The **trunk deploy** command dynamically places trunk configuration changes into effect, without a software reload.

CLI Syntax

Syntax: [no] trunk ethernet [<slotnum>/]<primary-portnum> to [<slotnum>/]<portnum> [ethernet [<slotnum>/]<primary-portnum>

Syntax: trunk deploy

Each **ethernet** parameter introduces a port group. Enter the slot number (if applicable) and the port number of the Ethernet port.

The <slotnum> parameter is required on chassis devices.

The <primary-portnum> **to** <portnum> parameters specify a port group. Notice that each port group must begin with a primary port. After you enter this command, the primary port of the first port group specified (which must be the group with the lower port numbers) becomes the primary port for the entire trunk group.

To configure a trunk group consisting of two groups of two ports each, enter commands such as the following:

```
FastIron SuperX Router(config)# trunk ethernet 1/1 to 1/2 ethernet 3/3 to 3/4
FastIron SuperX Router(config)# write memory
FastIron SuperX Router(config)# trunk deploy
```

Notice that the groups of ports meet the criteria for a multi-slot trunk group. Each group contains the same number of ports (two) and begins on a primary port (1/1 and 3/3).

Additional Trunking Options

The CLI contains commands for doing the following:

- Naming a trunk port
- Disabling or re-enabling a trunk port
- Deleting a trunk group

Naming a Trunk Port

To name an individual port in a trunk group, enter a command such as the following at the trunk group configuration level:

```
FastIron SuperX Router(config-trunk-4/1-4/4)# port-name customer1 ethernet 4/2
```

Syntax: [no] port-name <text> ethernet [<slotnum>/]<portnum>

The <text> parameter specifies the port name. The name can be up to 50 characters long.

The <slotnum> parameter is required for chassis devices.

This command assigns the name “customer1” to port 4/2 in the trunk group consisting of ports 4/1 – 4/4.

Disabling or Re-Enabling a Trunk Port

You can disable or re-enable individual ports in a trunk group. To disable an individual port in a trunk group, enter commands such as the following at the trunk group configuration level:

```
FastIron SuperX Router(config-trunk-4/1-4/4)# config-trunk-ind
```

```
FastIron SuperX Router(config-trunk-4/1-4/4)# disable ethernet 4/2
```

Syntax: [no] config-trunk-ind

Syntax: [no] disable ethernet [<slotnum>/]<portnum>

The **config-trunk-ind** command enables configuration of individual ports in the trunk group. If you do not use this command, the **disable** command will be valid only for the primary port in the trunk group and will disable all ports in the trunk group. You need to enter the **config-trunk-ind** command only once in a trunk group. After you enter the command, all applicable port configuration commands apply to individual ports only.

NOTE: If you enter **no config-trunk-ind**, all port configuration commands are removed from the individual ports and the configuration of the primary port is applied to all the ports. Also, once you enter the **no config-trunk-ind** command, the **enable**, **disable**, and **monitor** commands are valid only on the primary port and apply to the entire trunk group.

The **disable** command disables the port. The states of other ports in the trunk group are not affected.

If you have configured a name for the trunk port, you can specify the port name, as shown in the following example:

```
FastIron SuperX Router(config-trunk-4/1-4/4)# config-trunk-ind
FastIron SuperX Router(config-trunk-4/1-4/4)# disable customer1
```

Syntax: disable <portname>

To enable an individual port in a trunk group, enter commands such as the following at the trunk group configuration level:

```
FastIron SuperX Router(config-trunk-4/1-4/4)# config-trunk-ind
FastIron SuperX Router(config-trunk-4/1-4/4)# enable ethernet 4/2
```

Syntax: enable ethernet [<slotnum>/]<portnum>

Syntax: enable <portname>

Disabling or Re-Enabling a Range or List of Trunk Ports

To disable a range of ports in a trunk group, enter commands such as the following:

```
FastIron SuperX Router(config)# trunk switch ethernet 2/1 to 2/8
FastIron SuperX Router(config-trunk-2/1-2/8)# config-trunk-ind
FastIron SuperX Router(config-trunk-2/1-2/8)# disable ethernet 2/2 to 2/5
```

This command disables ports 2/2 – 2/5 in trunk group 2/1 – 2/8.

To disable a list of ports, enter a command such as the following:

```
FastIron SuperX Router(config-trunk-2/1-2/8)# disable ethernet 2/2 ethernet 2/4
ethernet 2/7
```

This command disables ports 2/2, 2/4, and 2/7 in the trunk group.

You can specify a range and a list on the same command line. For example, to re-enable some trunk ports, enter a command such as the following:

```
FastIron SuperX Router(config-trunk-2/1-2/8)# enable ethernet 2/2 to 2/5 ethernet 2/
7
```

Syntax: [no] disable ethernet [<slotnum>/]<portnum> [to [<slotnum>/]<portnum> | ethernet [<slotnum>/]<portnum>]

Syntax: [no] enable ethernet [<slotnum>/]<portnum> [to [<slotnum>/]<portnum> | ethernet [<slotnum>/]<portnum>]

The <slotnum> parameter is required on chassis devices.

The **to** <portnum> parameter indicates that you are specifying a range. Specify the lower port number in the range first, then **to**, then the higher port number in the range.

The <portnum> parameter specifies an individual port. You can enter this parameter multiple times to specify a list, as shown in the examples above.

Deleting a Trunk Group

To delete a trunk group, use "**no**" in front of the command you used to create the trunk group. For example, to remove one of the trunk groups configured in the examples above, enter the following command:

```
FastIron SuperX Router(config)# no trunk ethernet 1/1 to 1/2 ethernet 3/3 to 3/4
```

Syntax: no trunk ethernet [<slotnum>/]<portnum> to [<slotnum>/]<portnum>

The <slotnum> parameter is required on chassis devices.

Displaying Trunk Group Configuration Information

To display configuration information for the trunk groups, use the **show trunk** command. This command displays information for configured trunk groups and operational trunk groups. A configured trunk group is one that has been configured in the software but has not been placed into operation by a reset or reboot. An operational trunk group is one that has been placed into operation by a reset or reboot.

Enter the following command at any CLI level:

```
FastIron SuperX Router(config)# show trunk

Configured trunks:

Trunk ID: 1
HW Trunk ID: 1
Ports_Configured: 8
Primary Port Monitored: Jointly

Ports        1/1      1/2      1/3      1/4      1/5      1/6      1/7      1/8
Port Names   none     none     none     none     none     longna   test     none
Port_Status  enable   enable   enable   enable   disable  disable  enable   enable
Monitor      on       on       off      on       off      off      off      off
Mirror Port  3/3      3/4      N/A      3/5      N/A      N/A      N/A      N/A
Monitor Dir  both     in       N/A      out      N/A      N/A      N/A      N/A

Operational trunks:

Trunk ID: 1
HW Trunk ID: 1
Duplex: Full
Speed: 1G
Tag: No
Priority: level0
Active Ports: 6

Ports           1/1      1/2      1/3      1/4      1/5    1/6      1/7      1/8
Link_Status     active   active   active   active   down   down     active   active
LACP_Status     ready    ready    ready    expired  down   down     ready    ready
Load Sharing
 Mac Address    3        2        2        2        0      0        6        1
 IP             0        0        0        0        0      0        0        0
 Multicast      4        2        5        2        0      0        2        3
```

Syntax: show trunk [ethernet [<slotnum>/]<portnum> to [<slotnum>/]<portnum>]

The [<slotnum/> applies to chassis devices only.

Table 10.5 describes the information displayed by the **show trunk** command.

Table 10.5: CLI Trunk Group Information

This Field...	Displays...
Trunk ID	The trunk group number. The software numbers the groups in the display to make the display easy to use.
HW Trunk ID	The trunk ID.
Duplex	The mode of the port, which can be one of the following: • None – The link on the primary trunk port is down. • Full – The primary port is running in full-duplex. • Half – The primary port is running in half-duplex. **Note**: This field and the following fields apply only to operational trunk groups.
Speed	The speed set for the port. The value can be one of the following: • None – The link on the primary trunk port is down. • 10 – The port speed is 10 Mbps. • 100 – The port speed is 100 Mbps. • IG – The port speed is 1000 Mbps.
Tag	Indicates whether the ports have 802.1Q VLAN tagging. The value can be Yes or No.
Priority	Indicates the Quality of Service (QoS) priority of the ports. The priority can be a value from 0 – 7.
Active Ports	The number of ports in the trunk group that are currently active.
Ports	The ports in the trunk group.
Link_Status	The link status or each port in the trunk group.
LACP_Status	For more information about this feature, see the section “Displaying and Determining the Status of Aggregate Links” on page 10-22. • Ready - The port is functioning normally in the trunk group and is able to transmit and receive LACP packets. • Expired - The time has expired (as determined by timeout values) and the port has shut down because the port on the other side of the link has stopped transmitting packets. • Down - The port’s physical link is down.
Load Sharing	The number of traffic flows currently being load balanced on the trunk ports. All traffic exchanged within the flow is forwarded on the same trunk port. For information about trunk load sharing, see “Trunk Group Load Sharing” on page 10-6.

Dynamic Link Aggregation

Foundry software supports the IEEE 802.3ad standard for link aggregation. This standard describes the Link Aggregation Control Protocol (LACP), a mechanism for allowing ports on both sides of a redundant link to configure themselves into a trunk link (aggregate link), without the need for manual configuration of the ports into trunk groups.

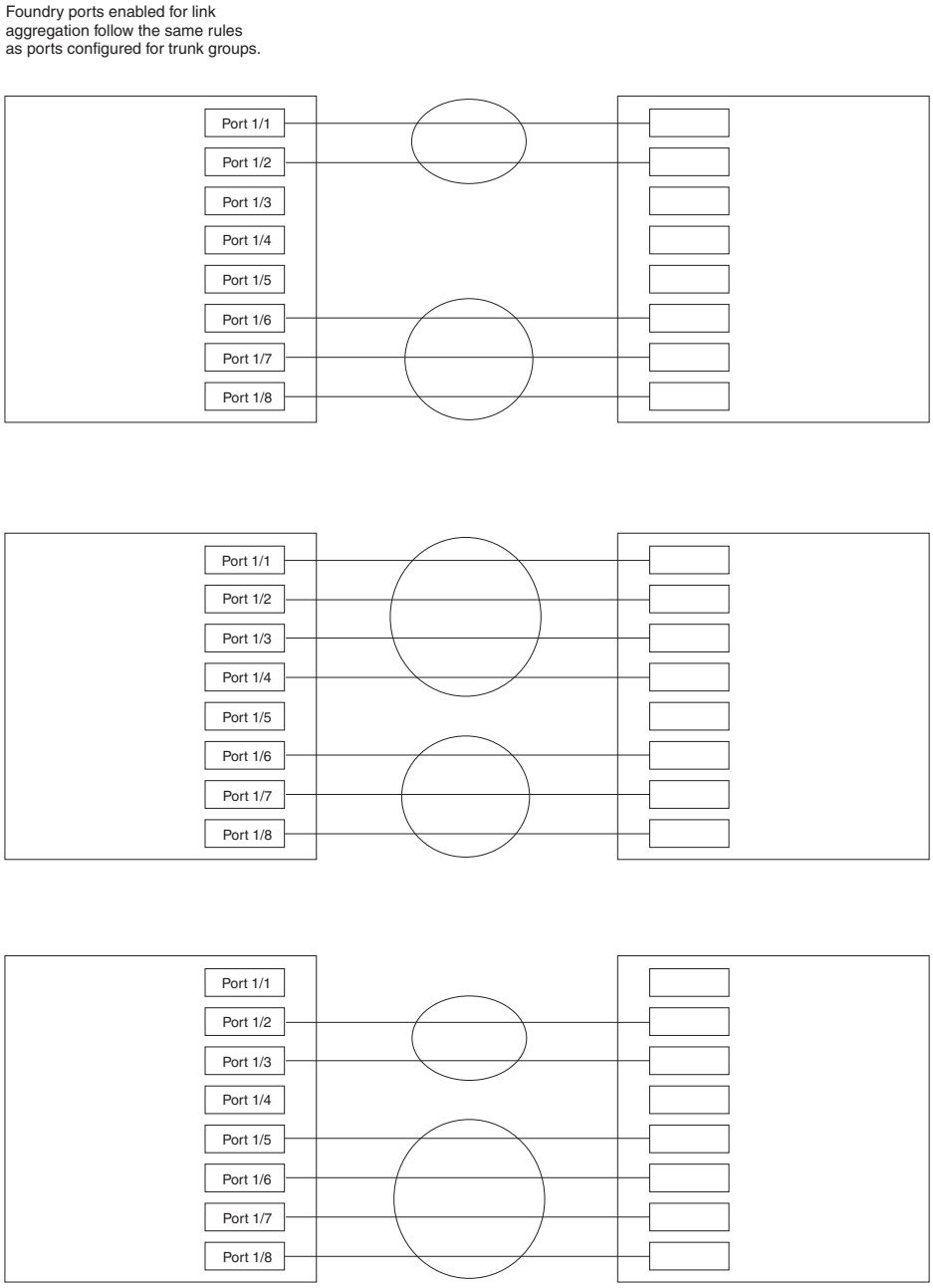
When you enable link aggregation on a group of Foundry ports, the Foundry ports can negotiate with the ports at the remote ends of the links to establish trunk groups.

Configuration Example

Foundry ports follow the same configuration rules for dynamically created aggregate links as they do for statically configured trunk groups. See “Trunk Group Rules” on page 10-3 and “Trunk Group Load Sharing” on page 10-6.

Figure 10.5 on page 10-14 shows some examples of valid aggregate links.

Figure 10.5 Examples of valid aggregate links



In this example, assume that link aggregation is enabled on all of the links between the Foundry device on the left and the device on the right (which can be either a Foundry device or another vendor's device). The ports that are members of aggregate links in this example are following the configuration rules for trunk links on Foundry devices.

The Foundry rules apply to a Foundry device even if the device at the other end is from another vendor and uses different rules. See "Trunk Group Rules" on page 10-3.

The link aggregation feature automates trunk configuration but can coexist with Foundry's trunk group feature. Link aggregation parameters do not interfere with trunk group parameters.

NOTE: Use the link aggregation feature only if the device at the other end of the link you want to aggregate also supports IEEE 802.3ad link aggregation. Otherwise, you need to manually configure the trunk links.

Link aggregation support is disabled by default. You can enable the feature on an individual port basis, in active or passive mode.

- Active mode – When you enable a port for active link aggregation, the Foundry port can exchange standard LACP Protocol Data Unit (LACPDU) messages to negotiate trunk group configuration with the port on the other side of the link. In addition, the Foundry port actively sends LACPDU messages on the link to search for a link aggregation partner at the other end of the link, and can initiate an LACPDU exchange to negotiate link aggregation parameters with an appropriately configured remote port.
- Passive mode – When you enable a port for passive link aggregation, the Foundry port can exchange LACPDU messages with the port at the remote end of the link, but the Foundry port cannot search for a link aggregation port or initiate negotiation of an aggregate link. Thus, the port at the remote end of the link must initiate the LACPDU exchange.

NOTE: Foundry recommends that you disable or remove the cables from the ports you plan to enable for dynamic link aggregation. Doing so prevents the possibility that LACP will use a partial configuration to talk to the other side of a link. A partial configuration does not cause errors, but does sometimes require LACP to be disabled and re-enabled on both sides of the link to ensure that a full configuration is used. It's easier to disable a port or remove its cable first. This applies both for active link aggregation and passive link aggregation.

Configuration Notes

- You cannot use 802.3ad link aggregation on a port configured as a member of a static trunk group.
- This feature is supported only on Gigabit Ethernet ports.
- The dynamic link aggregation (802.3ad) implementation on the FESX, FSX, and FWSX allow any number of ports up to four to be aggregated into a link. The feature does not require the aggregate link to consist of exactly two or four ports.
- When the feature dynamically adds or changes a trunk group, the **show trunk** command displays the trunk as both configured and active. However, the **show running-config** or **write terminal** command does not contain a trunk command defining the new or changed trunk group.
- If the feature places a port into a trunk group as a secondary port, all configuration information except information related to link aggregation is removed from the port. For example, if port 1/3 has an IP interface, and the link aggregation feature places port 1/3 into a trunk group consisting of ports 1/1 – 1/4, the IP interface is removed from the port.
- If you use this feature on a Layer 3 Switch that is running OSPF or BGP4, the feature causes these protocols to reset when a dynamic link change occurs. The reset includes ending and restarting neighbor sessions with OSPF and BGP4 peers, and clearing and relearning dynamic route entries and forwarding cache entries. Although the reset causes a brief interruption, the protocols automatically resume normal operation.
- Starting with software release 02.4.00, you can enable link aggregation on 802.1Q tagged ports (ports that belong to more than one port-based VLAN). In releases prior to 02.4.00, link aggregation works with untagged ports only.
- Dynamic Operation of Allocation Keys (section 43.6.2 in the 802.3ad specification) is supported.
- The trunks that will be formed by link aggregation will strictly adhere to the static trunking rules on the FastIron family of switches. Be careful in selecting keys if you are manually configuring link aggregation keys. Make sure that the possible trunks that you expect to be formed conform to the static trunking rules.

Adaptation to Trunk Disappearance

The Foundry device will tear down an aggregate link if the device at the other end of the link reboots or brings all the links down. Tearing the aggregate link down prevents a mismatch if the other device has a different trunk

configuration following the reboot or re-establishment of the links. Once the other device recovers, 802.3 can renegotiate the link without a mismatch.

Flexible Trunk Eligibility

The criteria for being eligible to be in an aggregate link are flexible. A range of ports can contain down ports and still be eligible to become an aggregate link.

The device groups the device's ports into 2-port groups consisting of an odd-numbered port and the next even-numbered port. For example, ports 1/1 and 1/2 are a two-port group, as are ports 1/3 and 1/4, 9/1 and 9/10, and so on. If either of the ports in a two-port group is up, the device considers both ports to be eligible to be in an aggregate link.

Figure 10.6 shows an example of 2-port groups in a range of four ports on which link aggregation is enabled. Based on the states of the ports, some or all of them will be eligible to be used in an aggregate link.

Figure 10.6 Two-port groups used to determine aggregation eligibility

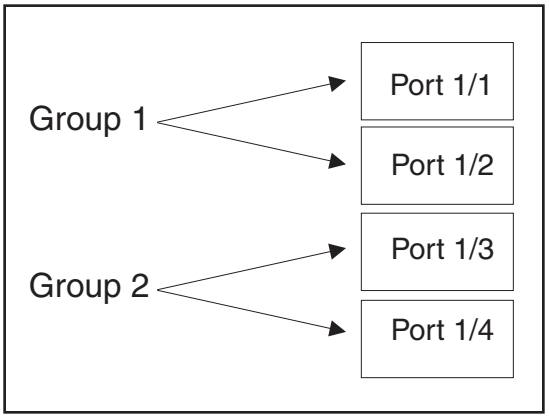


Table 10.6 shows examples of the ports from Figure 10.6 that will be eligible for an aggregate link based on individual port states.

Table 10.6: Port Eligibility for Link Aggregation

	Port Group 1		Port Group 2		Trunk Eligibility
	1/1	1/2	1/3	1/4	
Link State	Up	Up	Up	Up	4-port 1/1 – 1/4
	Up	Up	Up	Down	4-port 1/1 – 1/4
	Up	Down	Up	Down	4-port 1/1 – 1/4
	Up	Up	Down	Up	4-port 1/1 – 1/4
	Down	Down	Down	Up	2-port 1/3 – 1/4
	Up	Down	Down	Down	2-port 1/1 – 1/2

As shown in these examples, all or a subset of the ports within a port range will be eligible for formation into an aggregate link based on port states. Notice that the sets of ports that are eligible for the aggregate link must be valid static trunk configurations.

Command Syntax

By default, link aggregation is disabled on all ports. To enable link aggregation on a set of ports, enter commands such as the following at the interface configuration level of the CLI.

NOTE: Configuration commands for link aggregation differ depending on whether you are using the default link aggregation key automatically assigned by the software, or if you are assigning a different, unique key. Follow the commands below, according to the type of key you are using. For more information about keys, see “Key” on page 10-18.

Using the Default Key Assigned by the Software

```
FastIron SuperX Router(config)# interface ethernet 1/1
FastIron SuperX Router(config-if-e1000-1/1)# link-aggregate active
FastIron SuperX Router(config)# interface ethernet 1/2
FastIron SuperX Router(config-if-e1000-1/2)# link-aggregate active
```

The commands in this example enable the active mode of link aggregation on ports 1/1 and 1/2. The ports can send and receive LACPDU messages. Note that these ports will use the default key, since one has not been explicitly configured.

NOTE: In conformance with the 802.3ad specification, the default key assigned to an aggregate link is based on the port type (1-Gigabit port or 10-Gigabit port). The Foundry device assigns different keys to 10-Gigabit ports than 1-Gigabit ports, so that ports with different physical capabilities will not be able to form a trunk.

Assigning a Unique Key

```
FastIron SuperX Router(config)# interface ethernet 1/1
FastIron SuperX Router(config-if-e1000-1/1)# link-aggregate configure key 10000
FastIron SuperX Router(config-if-e1000-1/1)# link-aggregate active
FastIron SuperX Router(config)# interface ethernet 1/2
FastIron SuperX Router(config-if-e1000-1/2)# link-aggregate configure key 10000
FastIron SuperX Router(config-if-e1000-1/2)# link-aggregate active
```

The commands in this example assign the key 10000 and enable the active mode of link aggregation on ports 1/1 and 1/2. The ports can send and receive LACPDU messages.

NOTE: As shown in this example, when configuring a key, it is pertinent that you assign the key prior to enabling link aggregation.

The following commands enable passive link aggregation on ports 1/5 – 1/8:

```
FastIron SuperX Router(config)# interface ethernet 1/5 to 1/8
FastIron SuperX Router(config-mif-1/5-1/8)# link-aggregate passive
```

The commands in this example enable the passive mode of link aggregation on ports 1/5 – 1/8. These ports wait for the other end of the link to contact them. After this occurs, the ports can send and receive LACPDU messages.

To disable link aggregation on a port, enter a command such as the following:

```
FastIron SuperX Router(config-if-e1000-1/8)# link-aggregate off
```

Syntax: [no] link-aggregate active | passive | off

Syntax: [no] link-aggregate configure [system-priority <num>] | [port-priority <num>] | [key <num>] | [type server | switch]

NOTE: For more information about keys, including details about the syntax shown above, see “Key” on page 10-18.

Link Aggregation Parameters

You can change the settings on individual ports for the following link aggregation parameters:

- System priority
- Port priority
- Link type
- Key

System Priority

The system priority parameter specifies the Foundry device’s link aggregation priority relative to the devices at the other ends of the links on which link aggregation is enabled. A higher value indicates a lower priority. You can specify a priority from 0 – 65535. The default is 1.

NOTE: If you are connecting the Foundry device to another vendor’s device and the link aggregation feature is not working, set the system priority on the Foundry device to a lower priority (a higher priority value). In some cases, this change allows the link aggregation feature to operate successfully between the two devices.

Port Priority

The port priority parameter determines the active and standby links. When a group of ports is negotiating with a group of ports on another device to establish a trunk group, the Foundry port with the highest priority becomes the default active port. The other ports (with lower priorities) become standby ports in the trunk group. You can specify a priority from 0 – 65535. A higher value indicates a lower priority. The default is 1.

NOTE: This parameter is not supported in the current software release. The primary port in the port group becomes the default active port. The primary port is the lowest-numbered port in a valid trunk-port group.

Link Type

The link type parameter specifies whether the trunk is connecting to a server (server link) or to another networking device (switch link). The default link type is switch.

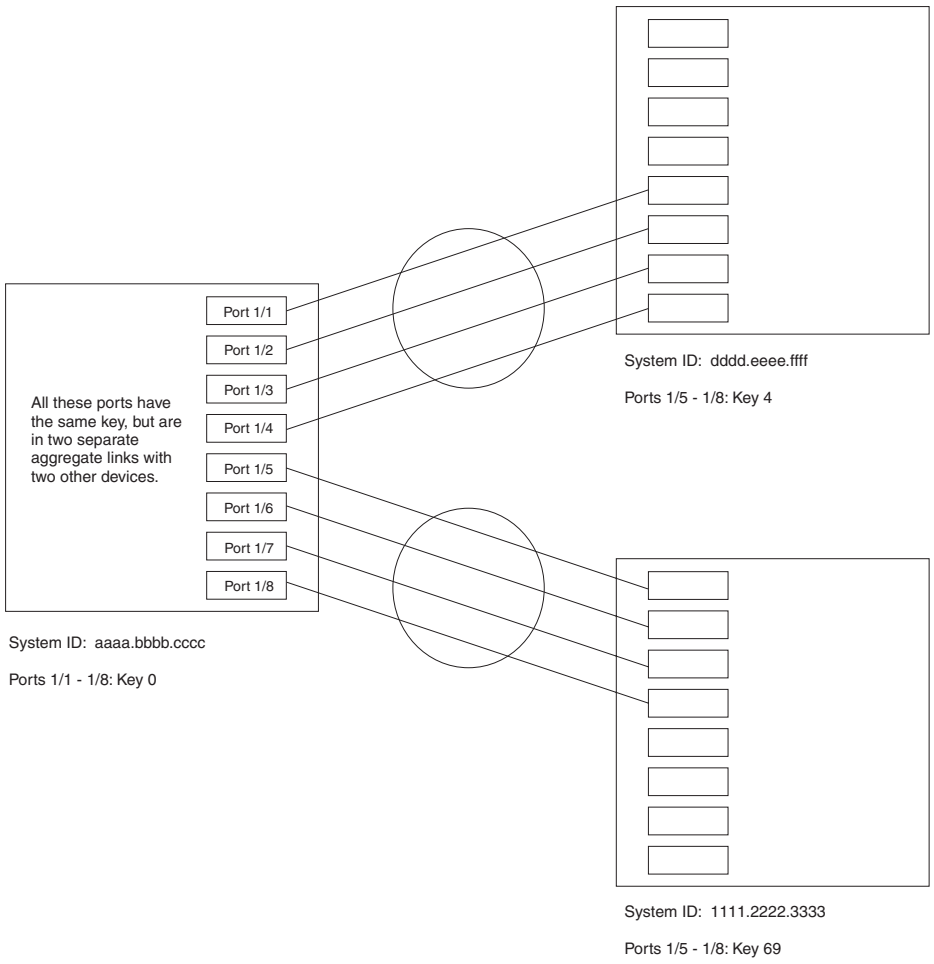
Key

Every port that is 802.3ad-enabled has a key. The key identifies the group of potential trunk ports to which the port belongs. Ports with the same key are called a key group and are eligible to be in the same trunk group.

When you enable link-aggregation on a tagged or untagged port, Foundry’s software assigns a default key to the port. The default key is based on the position of the port within an eight-port group (the maximum number of ports in a trunk group on a chassis device). The software assigns the keys in ascending numerical order, beginning with key 0 for the first group of eight ports. For example, a 24-port module in chassis slot 1 contains keys 0, 1, and 2 by default. Ports 1/1 – 1/8 have key 0, ports 1/9 – 1/16 have key 1, and so on.

All ports within an aggregate link must have the same key. However, if the device has ports that are connected to two different devices, and the port groups allow the ports to form into separate aggregate links with the two devices, then each group of ports can have the same key while belonging to separate aggregate links with different devices. Figure 10.7 on page 10-19 shows an example.

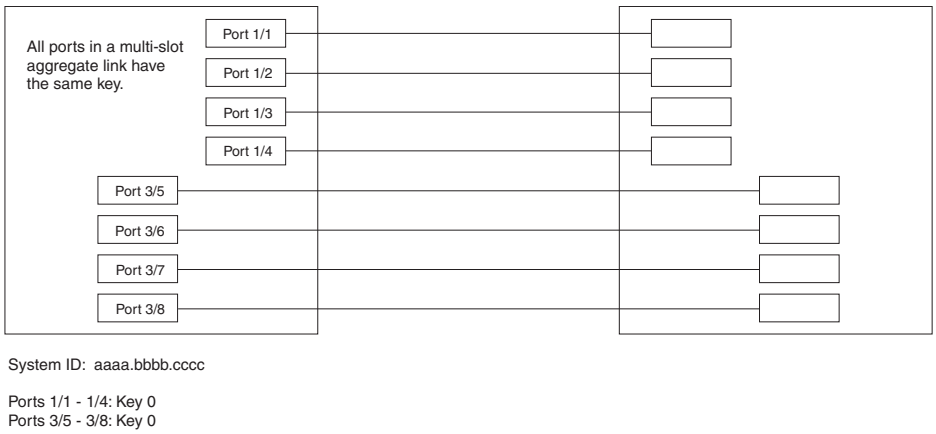
Figure 10.7 Ports with the same key in different aggregate links



Notice that the keys between one device and another do not need to match. The only requirement for key matching is that all the ports within an aggregate link on a given device must have the same key.

Devices that support multi-slot trunk groups can form multi-slot aggregate links using link aggregation. However, the link aggregation keys for the groups of ports on each module must match. For example, if you want to allow link aggregation to form an aggregate link containing ports 1/1 – 1/4 and 3/5 – 3/8, you must change the link aggregation key on one or both groups of ports so that the key is the same on all eight ports. Figure 10.8 on page 10-20 shows an example.

Figure 10.8 Multi-slot aggregate link



By default, the device's ports are divided into 4-port groups. The software dynamically assigns a unique key to each 4-port group. If you need to divide a 4-port group into two 2-port groups, change the key in one of the groups so that the two 2-port groups have different keys. For example, if you plan to use ports 1/1 and 1/2 in VLAN 1, and ports 1/3 and 1/4 in VLAN 2, change the key for ports 1/3 and 1/4.

NOTE: If you change the key for a port group, Foundry recommends that you use the value 10000 or higher, to avoid potential conflicts with dynamically created keys.

Dynamic Operation of Allocation Keys

The Foundry device dynamically changes a port's key based on changes to the port's VLAN membership.

When you change a port's VLAN membership, the device searches through existing key groups for a port with matching port properties. Specifically, it searches for a match on all three of the following properties:

- VLAN ID
- default key
- port tag type (tagged or untagged)

If it finds a match, the port (whose VLAN membership you are changing) gets the matching port's key. If it does not find a match, the port gets a new key.

NOTE: For multi-slot trunk groups, you must manually configure the keys in the trunk group(s) to match. For instructions on configuring keys manually, see "Configuring Keys For Ports with Link Aggregation Enabled" on page 10-22.

How Changing a Port's VLAN Membership Affects Trunk Groups and Dynamic Keys

When you change a port's VLAN membership and the port is currently a member of a trunk group, the following changes occur:

- The Foundry device tears down the existing trunk group.
- All ports in the trunk group get a new key.
- The new key group aggregates into a new trunk group.

When you change a port's VLAN membership, and the port is not a member of a trunk group, the following changes occur:

- The port gets a new key depending on changes to the port's VLAN tag type, as follows:
  - Tagged to Tagged VLAN – The primary port of the trunk group gets a new key.
  - Tagged to Untagged VLAN –The port gets the default key for untagged ports.

  - Untagged to Tagged VLAN – If the Foundry device finds a port with matching port properties, the port gets that port's key. If it doesn't find one, the port gets a new key.
  - Untagged to Untagged VLAN – The port gets a new key depending on whether it's in the default VLAN or not. If there is a trunk group associated with the key, it is not affected.
- All other ports keep their existing key.
- The new key groups try to aggregate into trunk groups.

Viewing Keys for Tagged Ports

To display link aggregation information, including the key for a specific port, enter a command such as the following at any level of the CLI:

```
FastIron SuperX Router# show link-aggregation ethernet 1/1

System ID: 00e0.52a9.bb00
Port   [Sys P] [Port P] [ Key ] [Act][Tio][Agg][Syn][Col][Dis][Def][Exp]
1/1         0         0       0   No    L    No   No   No   No   No   No
```

The command in this example shows the key and other link aggregation information for port 1/1.

To display link aggregation information, including the key for all ports on which link aggregation is enabled, enter the following command at any level of the CLI:

```
FastIron SuperX Router# sh link-agg

System ID: 0004.8055.b200
Long  timeout: 90, default: 90
Short timeout: 3, default: 3

Port   [Sys P] [Port P] [ Key ] [Act][Tio][Agg][Syn][Col][Dis][Def][Exp][Ope]
1/1         1         1   10000   Yes   S    Agg  Syn  Col  Dis  Def  No   Dwn
1/2         1         1   10000   Yes   S    Agg  Syn  Col  Dis  Def  No   Dwn
2/1         1         1   10000   Yes   S    Agg  Syn  Col  Dis  Def  No   Dwn
2/2         1         1   10000   Yes   S    Agg  Syn  Col  Dis  Def  No   Dwn
4/1         1         1     480   Yes   S    Agg  Syn  Col  Dis  Def  No   Dwn
4/2         1         1     480   Yes   S    Agg  Syn  Col  Dis  Def  No   Dwn
4/3         1         1     480   Yes   S    Agg  Syn  Col  Dis  Def  No   Dwn
4/4         1         1     480   Yes   S    Agg  Syn  Col  Dis  Def  No   Dwn
4/17        1         1     481   Yes   S    Agg  Syn  Col  Dis  Def  No   Ope
4/18        1         1     481   Yes   S    Agg  Syn  Col  Dis  Def  No   Ope
4/19        1         1     481   Yes   S    Agg  Syn  Col  Dis  Def  No   Ope
4/20        1         1     481   Yes   S    Agg  Syn  Col  Dis  Def  No   Ope
```

Syntax: show link-aggregation [ethernet [<slotnum>/]<portnum>]

Possible values: N/A

Default value: N/A

Configuring Link Aggregation Parameters

You can configure one or more parameters on the same command line, and you can enter the parameters in any order.

NOTE: For key configuration only, configuration commands differ depending on whether or not link aggregation is enabled on the port(s). Follow the appropriate set of commands below, according to your system's configuration.

For example, to change a port group's key from the one assigned by the software to another value, enter commands such as the following:

NOTE: Use this command sequence to change the key for ports that do not have link aggregation enabled, and for all other link aggregation parameters (i.e., system priority, port priority, and link type).

```
FastIron SuperX Router(config)# interface ethernet 1/1 to 1/4
FastIron SuperX Router(config-mif-1/1-1/4)# link-aggregate configure key 10000
FastIron SuperX Router(config-mif-1/1-1/4)# interface ethernet 3/5 to 3/8
FastIron SuperX Router(config-mif-3/5-3/8)# link-aggregate configure key 10000
```

Configuring Keys For Ports with Link Aggregation Enabled

NOTE: As shown in this command sequence, to change the key on ports that already have link aggregation enabled, you must first turn OFF link aggregation, configure the new key, then re-enable link aggregation.

```
FastIron SuperX Router(config)# interface ethernet 1/1 to 1/4
FastIron SuperX Router(config-mif-1/1-1/4)# link-aggregate off
FastIron SuperX Router(config-mif-1/1-1/4)# link-aggregate configure key 10000
FastIron SuperX Router(config-mif-1/1-1/4)# link-aggregate active
FastIron SuperX Router(config-mif-1/1-1/4)# interface ethernet 3/5 to 3/8
FastIron SuperX Router(config-mif-3/5-3/8)# link-aggregate off
FastIron SuperX Router(config-mif-3/5-3/8)# link-aggregate configure key 10000
FastIron SuperX Router(config-mif-3/5-3/8)# link-aggregate active
```

These commands change the key for ports 1/1 – 1/4 and 3/5 – 3/8 to 10000. Since all ports in an aggregate link must have the same key, the command in this example enables ports 1/1 – 1/4 and 3/5 – 3/8 to form a multi-slot aggregate link.

Syntax: [no] link-aggregate configure [system-priority <num>] | [port-priority <num>] | [key <num>] | [type server | switch]

The **system-priority** <num> parameter specifies the Foundry device's link aggregation priority. A higher value indicates a lower priority. You can specify a priority from 0 – 65535. The default is 1.

The **port-priority** <num> parameter specifies an individual port's priority within the port group. A higher value indicates a lower priority. You can specify a priority from 0 – 65535. The default is 1.

The **key** <num> parameter identifies the group of ports that are eligible to be aggregated into a trunk group. The software automatically assigns a key to each group of ports. The software assigns the keys in ascending numerical order, beginning with 0. You can change a port group's key to a value from 0 – 65535.

NOTE: If you change the key for a port group, Foundry recommends that you use the value 10000 or higher, to avoid potential conflicts with dynamically created keys.

The **type server | switch** parameter specifies whether the port group is connected to a server (**server**) or to another networking device (**switch**). The default is **switch**.

You can enter one or more of the command's parameters on the same command line, in any order.

Displaying and Determining the Status of Aggregate Links

The **show link-aggregation** command provides the ability to view the status of dynamic links. You can determine the status of ports that are members of an aggregate link, and tell whether or not LACPCU messages are being transmitted between the ports.

The following section provides details about the events that can affect the status of ports in an aggregate link and the status of LACP messages exchanged between the ports. Later sections provide instructions for viewing these status reports.

About Blocked Ports

Foundry devices can block traffic on a port or shut down a port that is part of a trunk group or aggregate link, when a port joins a trunk group and the port on the other end of the link shuts down or stops transmitting LACP packets. Depending on the timeout value set on the port, the link aggregation information expires. If this occurs, the Foundry device shuts down the port and notifies all the upper layer protocols that the port is down.

Foundry devices can also block traffic on a port that is initially configured with link aggregation. The port is blocked until it joins a trunk group. In this case, traffic is blocked, but the port is still operational.

A port remains blocked until one of the following events occur:

- Both ports in the aggregate link have the same key
- LACP brings the port back up
- The port joins a trunk group

Displaying Link Aggregation and Port Status Information

Use the **show link-aggregation** command to determine the operational status of ports associated with aggregate links.

To display the link aggregation information for a specific port, enter a command such as the following at any level of the CLI:

```
FastIron SuperX Router(config-mif-1/1-1/8)# show link-aggregation ethernet 1/1
System ID: 00e0.52a9.bb00
Port  [Sys P] [Port P] [ Key ] [Act][Tio][Agg][Syn][Col][Dis][Def][Exp] [Ope]
1/1        0        0       0   No    L   No   No   No   No   No   No    Ope
```

The command in this example shows the link aggregation information for port 1/1.

To display the link aggregation information for all ports on which link aggregation is enabled, enter the following command at any level of the CLI:

```
FastIron SuperX Router(config)# show link-aggregation

System ID: 00e0.52a9.bb00
Port  [Sys P] [Port P] [ Key ] [Act][Tio][Agg][Syn][Col][Dis][Def][Exp][Ope]
1/1        1        1       0   No    L   Agg  Syn  No   No   Def  Exp  Ope
1/2        1        1       0   No    L   Agg  Syn  No   No   Def  Exp  Ina
1/3        1        1       0   No    L   Agg  Syn  No   No   Def  Exp  Ina
1/4        1        1       0   No    L   Agg  Syn  No   No   Def  Exp  Blo
1/5        1        1       1   No    L   Agg  No   No   No   Def  Exp  Ope
1/6        1        1       1   No    L   Agg  No   No   No   Def  Exp  Ope
1/7        1        1       1   No    L   Agg  No   No   No   Def  Exp  Dwn
1/8        1        1       1   No    L   Agg  No   No   No   Def  Exp  Dwn
```

Syntax: show link-aggregation [ethernet [<slotnum>/]<portnum>]

The <slotnum> parameter is required on chassis devices.

Use **ethernet <portnum>** to display link-aggregation information for a specific port.

NOTE: Ports that are configured as part of an aggregate link must also have the same key. For more information about assigning keys, see the section “Link Aggregation Parameters” on page 10-18.

The **show link aggregation** command shows the following information.

Table 10.7: CLI Display of Link Aggregation Information

This Field...	Displays...
System ID	Lists the base MAC address of the device. This is also the MAC address of port 1 (or 1/1).
Port	Lists the port number.
Sys P	Lists the system priority configured for this port.
Port P	Lists the port's link aggregation priority.
Key	Lists the link aggregation key.
Act	Indicates the link aggregation mode, which can be one of the following: • No – The mode is passive or link aggregation is disabled (off) on the port. If link aggregation is enabled (and the mode is passive), the port can send and receive LACPDU messages to participate in negotiation of an aggregate link initiated by another port, but cannot search for a link aggregation port or initiate negotiation of an aggregate link. • Yes – The mode is active. The port can send and receive LACPDU messages.
Tio	Indicates the timeout value of the port. The timeout value can be one of the following: • L – Long. The trunk group has already been formed and the port is therefore using a longer message timeout for the LACPDU messages exchanged with the remote port. Typically, these messages are used as confirmation of the health of the aggregate link. • S – Short. The port has just started the LACPDU message exchange process with the port at the other end of the link. The S timeout value also can mean that the link aggregation information received from the remote port has expired and the ports are starting a new information exchange.
Agg	Indicates the link aggregation state of the port. The state can be one of the following: • Agg – Link aggregation is enabled on the port. • No – Link aggregation is disabled on the port.

Table 10.7: CLI Display of Link Aggregation Information (Continued)

This Field...	Displays...
Syn	Indicates the synchronization state of the port. The state can be one of the following: • No – The port is out of sync with the remote port. The port does not understand the status of the LACPDU process and is not prepared to enter a trunk link. • Syn – The port is in sync with the remote port. The port understands the status of the LACPDU message exchange process, and therefore knows the trunk group to which it belongs, the link aggregation state of the remote port, and so on.
Col	Indicates the collection state of the port, which determines whether the port is ready to send traffic over the trunk link. • Col – The port is ready to send traffic over the trunk link. • No – The port is not ready to send traffic over the trunk link.
Dis	Indicates the distribution state of the port, which determines whether the port is ready to receive traffic over the trunk link. • Dis – The port is ready to receive traffic over the trunk link. • No – The port is not ready to receive traffic over the trunk link.
Def	Indicates whether the port is using default link aggregation values. The port uses default values if it has not received link aggregation information through LACP from the port at the remote end of the link. This field can have one of the following values: • Def – The port has not received link aggregation values from the port at the other end of the link and is therefore using its default link aggregation LACP settings. • No – The port has received link aggregation information from the port at the other end of the link and is using the settings negotiated with that port.
Exp	Indicates whether the negotiated link aggregation settings have expired. The settings expire if the port does not receive an LACPDU message from the port at the other end of the link before the message timer expires. This field can have one of the following values: • Exp – The link aggregation settings this port negotiated with the port at the other end of the link have expired. The port is now using its default link aggregation settings. • No – The link aggregation values that this port negotiated with the port at the other end of the link have not expired, so the port is still using the negotiated settings.

Table 10.7: CLI Display of Link Aggregation Information (Continued)

This Field...	Displays...
Ope	• Ope (operational) - The port is operating normally. • Ina (inactive) - The port is inactive because the port on the other side of the link is down or has stopped transmitting LACP packets. • Blo (blocked) - The port is blocked because the adjacent port is not configured with link aggregation or because it is not able to join a trunk group. To unblock the port and bring it to an operational state, enable link aggregation on the adjacent port and ensure that the ports have the same key.

Displaying Trunk Group and LACP Status Information

Use the **show trunk** command to determine the status of LACP. See "Displaying Trunk Group Configuration Information" on page 10-11.

Clearing the Negotiated Aggregate Links Table

When a group of ports negotiates a trunk group configuration, the software stores the negotiated configuration in a table. You can clear the negotiated link aggregation configurations from the software. When you clear the information, the software does not remove link aggregation parameter settings you have configured. Only the configuration information negotiated using LACP is removed.

NOTE: The software automatically updates the link aggregation configuration based on LACPDU messages. However, clearing the link aggregation information can be useful if you are troubleshooting a configuration.

To clear the link aggregation information, enter the following command at the Privileged EXEC level of the CLI:

```
FastIron SuperX Router# clear link-aggregate
```

Syntax: clear link-aggregate

Chapter 11
Configuring Virtual LANs (VLANs)

This chapter describes how to configure Virtual LANs (VLANs) on Foundry Layer 2 Switches and Layer 3 Switches using the CLI.

This chapter contains information and configuration procedures and examples for the following:

Table 11.1: Chapter Contents

VLAN Overview

The following sections provide details about the VLAN types and features supported on the FastIron family of switches.

Types of VLANs

You can configure the following types of VLANs on Foundry devices.

- Layer 2 port-based VLAN – a set of physical ports that share a common, exclusive Layer 2 broadcast domain
- Layer 3 protocol VLANs – a subset of ports within a port-based VLAN that share a common, exclusive broadcast domain for Layer 3 broadcasts of the specified protocol type
- IP sub-net VLANs – a subset of ports in a port-based VLAN that share a common, exclusive sub-net broadcast domain for a specified IP sub-net
- IPv6 VLANs – a subset of ports in a port-based VLAN that share a common, exclusive network broadcast domain for IPv6 packets
- IPX network VLANs – a subset of ports in a port-based VLAN that share a common, exclusive network broadcast domain for a specified IPX network
- AppleTalk cable VLANs – a subset of ports in a port-based VLAN that share a common, exclusive network broadcast domain for a specified AppleTalk cable range

When a Foundry device receives a packet on a port that is a member of a VLAN, the device forwards the packet based on the following VLAN hierarchy:

- If the port belongs to an IP sub-net VLAN, IPX network VLAN, or AppleTalk cable VLAN and the packet belongs to the corresponding IP sub-net, IPX network, or AppleTalk cable range, the device forwards the packet to all the ports within that VLAN.
- If the packet is a Layer 3 packet but cannot be forwarded as described above, but the port is a member of a Layer 3 protocol VLAN for the packet's protocol, the device forwards the packet on all the Layer 3 protocol VLAN's ports.
- If the packet cannot be forwarded based on either of the VLAN membership types listed above, but the packet can be forwarded at Layer 2, the device forwards the packet on all the ports within the receiving port's port-based VLAN.

Protocol VLANs differ from IP sub-net, IPX network, and AppleTalk VLANs in an important way. Protocol VLANs accept any broadcast of the specified protocol type. An IP sub-net, IPx network, or AppleTalk VLAN accepts only broadcasts for the specified IP sub-net, IPX network, or AppleTalk cable range.

NOTE: Protocol VLANs are different from IP sub-net, IPX network, and AppleTalk cable VLANs. A port-based VLAN cannot contain both an IP sub-net, IPX network, or AppleTalk cable VLAN and a protocol VLAN for the same protocol. For example, a port-based VLAN cannot contain both an IP protocol VLAN and an IP sub-net VLAN.

Layer 2 Port-Based VLANs

On all Foundry devices, you can configure port-based VLANs. A port-based VLAN is a subset of ports on a Foundry device that constitutes a Layer 2 broadcast domain.

By default, all the ports on a Foundry device are members of the default VLAN. Thus, all the ports on the device constitute a single Layer 2 broadcast domain. You can configure multiple port-based VLANs. When you configure a port-based VLAN, the device automatically removes the ports you add to the VLAN from the default VLAN.

You can configure up to 4094 port-based VLANs on a Layer 2 Switch or Layer 3 Switch. On both device types, valid VLAN IDs are 1 – 4095. You can configure up to the maximum number of VLANs within that ID range.

NOTE: VLAN ID 4094 is reserved for use by Single STP. VLAN IDs 4091 and 4092 are reserved for use in the Layer 3 Switch and Base Layer 3 images. You can configure these VLAN IDs in the Layer 2 Switch image.

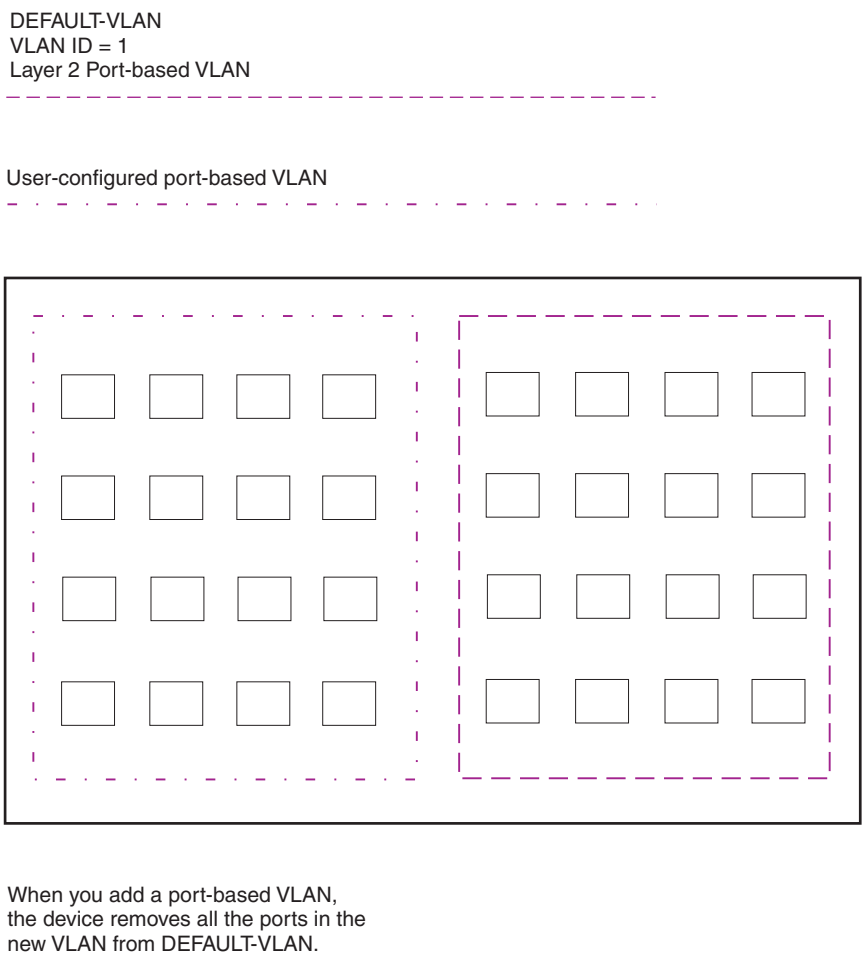
Each port-based VLAN can contain either tagged or untagged ports. A port cannot be a member of more than one port-based VLAN unless the port is tagged. ***802.1Q tagging*** allows the port to add a four-byte tag field, which contains the VLAN ID, to each packet sent on the port. You also can configure port-based VLANs that span multiple devices by tagging the ports within the VLAN. The tag enables each device that receives the packet to determine the VLAN the packet belongs to. 802.1Q tagging applies only to Layer 2 VLANs, not to Layer 3 VLANs.

Since each port-based VLAN is a separate Layer 2 broadcast domain, by default each VLAN runs a separate instance of the Spanning Tree Protocol (STP).

Layer 2 traffic is bridged within a port-based VLAN and Layer 2 broadcasts are sent to all the ports within the VLAN.

Figure 11.1 shows an example of a Foundry device on which a Layer 2 port-based VLAN has been configured.

Figure 11.1 Foundry device containing user-defined Layer 2 port-based VLAN



Layer 3 Protocol-Based VLANs

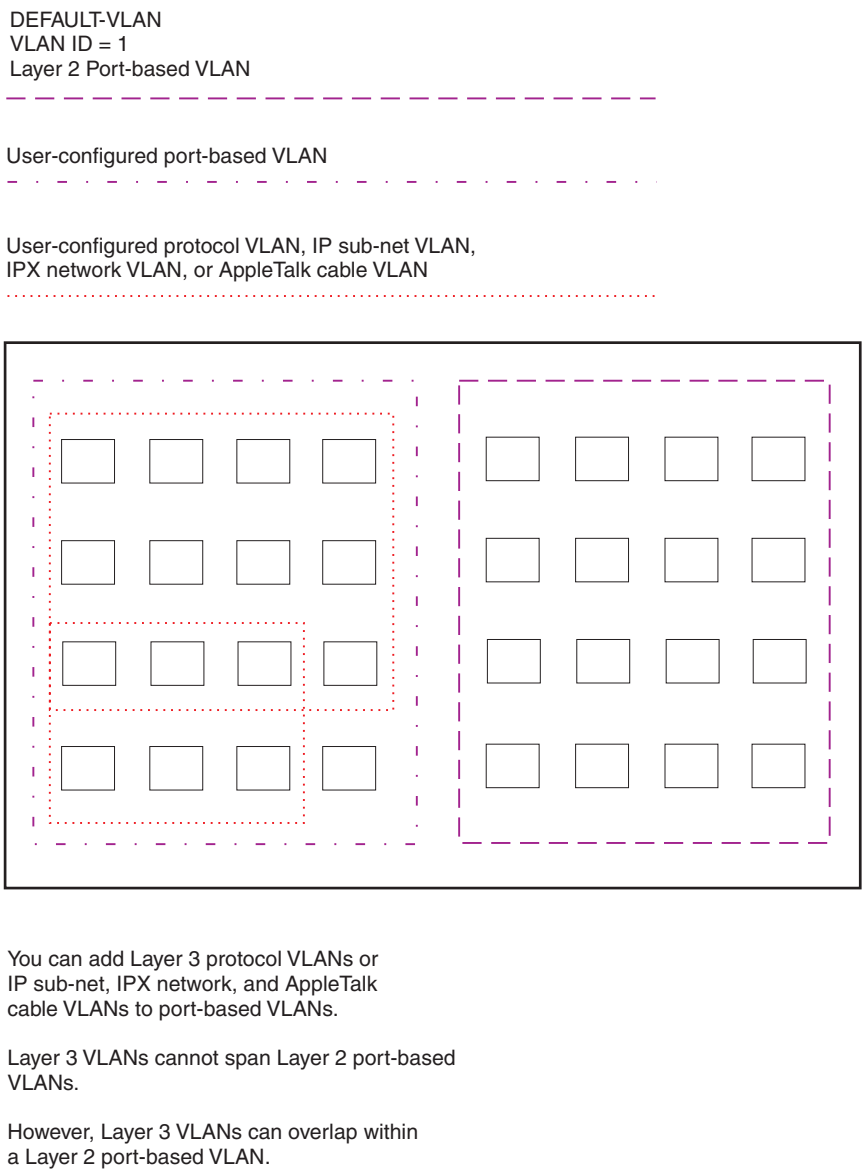
If you want some or all of the ports within a port-based VLAN to be organized according to Layer 3 protocol, you must configure a Layer 3 protocol-based VLAN within the port-based VLAN.

You can configure each of the following types of protocol-based VLAN within a port-based VLAN. All the ports in the Layer 3 VLAN must be in the same Layer 2 VLAN.

- AppleTalk – The device sends AppleTalk broadcasts to all ports within the AppleTalk protocol VLAN.
- IP – The device sends IP broadcasts to all ports within the IP protocol VLAN.
- IPv6 – The device sends IPv6 broadcasts to all ports within the IPv6 protocol VLAN.
- IPX – The device sends IPX broadcasts to all ports within the IPX protocol VLAN.
- DECnet – The device sends DECnet broadcasts to all ports within the DECnet protocol VLAN.
- NetBIOS – The device sends NetBIOS broadcasts to all ports within the NetBIOS protocol VLAN.
- Other – The device sends broadcasts for all protocol types other than those listed above to all ports within the VLAN.

Figure 11.2 shows an example of Layer 3 protocol VLANs configured within a Layer 2 port-based VLAN.

Figure 11.2 Layer 3 protocol VLANs within a Layer 2 port-based VLAN



Integrated Switch Routing (ISR)

Foundry Networks' ***Integrated Switch Routing (ISR)*** feature enables VLANs configured on Layer 3 Switches to route Layer 3 traffic from one protocol VLAN or IP sub-net, IPX network, or AppleTalk cable VLAN to another. Normally, to route traffic from one IP sub-net, IPX network, or AppleTalk cable VLAN to another, you would need to forward the traffic to an external router. The VLANs provide Layer 3 broadcast domains for these protocols but do not in themselves provide routing services for these protocols. This is true even if the source and destination IP sub-nets, IPX networks, or AppleTalk cable ranges are on the same device.

ISR eliminates the need for an external router by allowing you to route between VLANs using virtual routing interfaces (ves). A ***virtual routing interface*** is a logical port on which you can configure Layer 3 routing parameters. You configure a separate virtual routing interface on each VLAN that you want to be able to route from or to. For example, if you configure two IP sub-net VLANs on a Layer 3 Switch, you can configure a virtual routing interface on each VLAN, then configure IP routing parameters for the sub-nets. Thus, the Layer 3 Switch forwards IP sub-net broadcasts within each VLAN at Layer 2 but routes Layer 3 traffic between the VLANs using the virtual routing interfaces.

NOTE: The Layer 3 Switch uses the lowest MAC address on the device (the MAC address of port 1 or 1/1) as the MAC address for all ports within all virtual routing interfaces you configure on the device.

The routing parameters and the syntax for configuring them are the same as when you configure a physical interface for routing. The logical interface allows the Layer 3 Switch to internally route traffic between the protocol-based VLANs without using physical interfaces.

All the ports within a protocol-based VLAN must be in the same port-based VLAN. The protocol-based VLAN cannot have ports in multiple port-based VLANs, unless the ports in the port-based VLAN to which you add the protocol-based VLAN are 802.1Q tagged.

You can configure multiple protocol-based VLANs within the same port-based VLAN. In addition, a port within a port-based VLAN can belong to multiple protocol-based VLANs of the same type or different types. For example, if you have a port-based VLAN that contains ports 1 – 10, you can configure port 5 as a member of an AppleTalk protocol VLAN, an IP protocol VLAN, and an IPX protocol VLAN, and so on.

IP Sub-Net, IPX Network, and AppleTalk Cable VLANs

The protocol-based VLANs described in the previous section provide separate protocol broadcast domains for specific protocols. For IP, IPX, and AppleTalk, you can provide more granular broadcast control by instead creating the following types of VLAN:

- ***IP sub-net VLAN*** – An IP sub-net broadcast domain for a specific IP sub-net.
- ***IPX network VLAN*** – An IPX network broadcast domain for a specific IPX network.
- ***AppleTalk cable VLAN*** – An AppleTalk broadcast domain for a specific cable range.

You can configure these types of VLANs on Layer 3 Switches only. The Layer 3 Switch sends broadcasts for the IP sub-net, IPX network, or AppleTalk cable range to all ports within the IP sub-net, IPX network, or AppleTalk cable VLAN at Layer 2.

The Layer 3 Switch routes packets between VLANs at Layer 3. To configure an IP sub-net, IPX network, or AppleTalk cable VLAN to route, you must add a virtual routing interface to the VLAN, then configure the appropriate routing parameters on the virtual routing interface.

NOTE: The Layer 3 Switch routes packets between VLANs of the same protocol. The Layer 3 Switch cannot route from one protocol to another.

NOTE: IP sub-net VLANs are not the same thing as IP protocol VLANs. An IP protocol VLAN sends all IP broadcasts on the ports within the IP protocol VLAN. An IP sub-net VLAN sends only the IP sub-net broadcasts for the sub-net of the VLAN. You cannot configure an IP protocol VLAN and an IP sub-net VLAN within the same port-based VLAN.

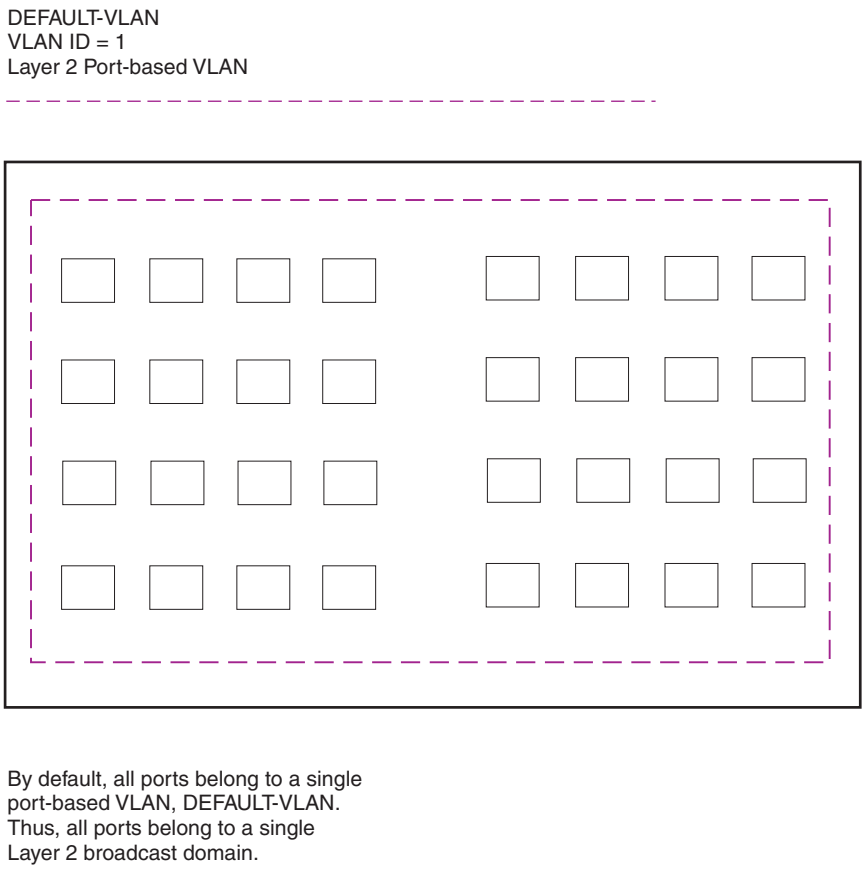
This note also applies to IPX protocol VLANs and IPX network VLANs, and to AppleTalk protocol VLANs and AppleTalk cable VLANs.

Default VLAN

By default, all the ports on a Foundry device are in a single port-based VLAN. This VLAN is called DEFAULT-VLAN and is VLAN number 1. Foundry devices do not contain any protocol VLANs or IP sub-net, IPX network, or AppleTalk cable VLANs by default.

Figure 11.3 shows an example of the default Layer 2 port-based VLAN.

Figure 11.3 Default Layer 2 port-based VLAN



When you configure a port-based VLAN, one of the configuration items you provide is the ports that are in the VLAN. When you configure the VLAN, the Foundry device automatically removes the ports that you place in the VLAN from DEFAULT-VLAN. By removing the ports from the default VLAN, the Foundry device ensures that each port resides in only one Layer 2 broadcast domain.

NOTE: Information for the default VLAN is available only after you define another VLAN.

Some network configurations may require that a port be able to reside in two or more Layer 2 broadcast domains (port-based VLANs). In this case, you can enable a port to reside in multiple port-based VLANs by tagging the port. See the following section.

If your network requires that you use VLAN ID 1 for a user-configured VLAN, you can reassign the default VLAN to another valid VLAN ID. See “Assigning a Different VLAN ID to the Default VLAN” on page 11-15.

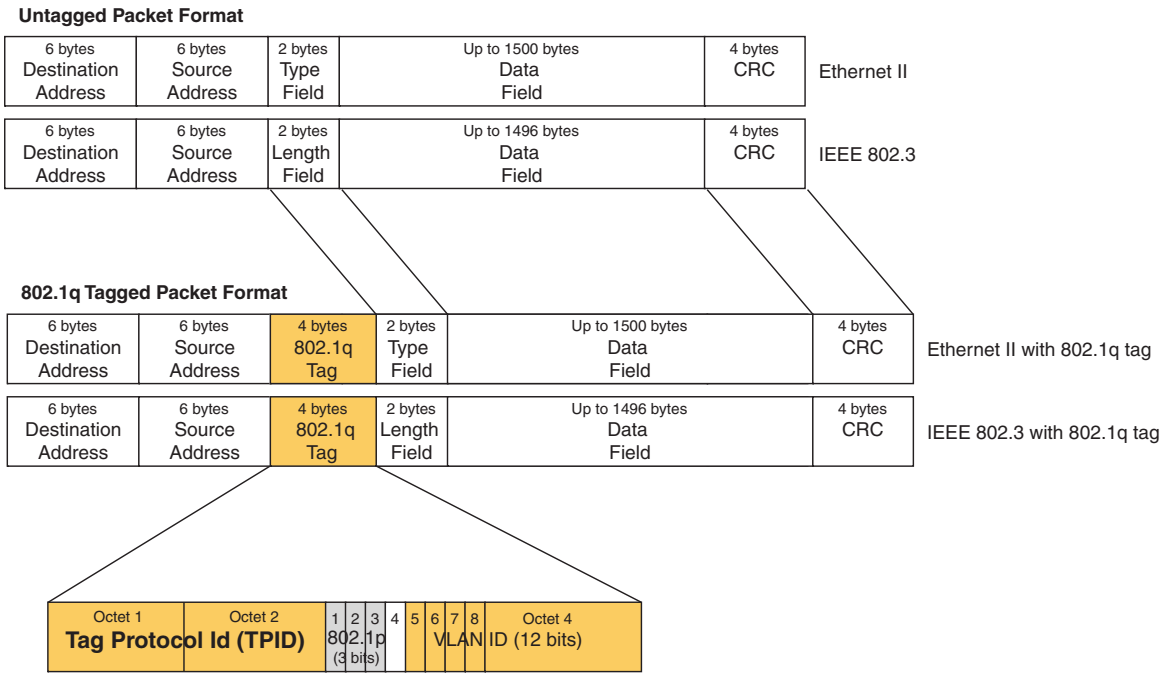
802.1Q Tagging

802.1Q tagging is an IEEE standard that allows a networking device to add information to a Layer 2 packet in order to identify the VLAN membership of the packet. Foundry devices tag a packet by adding a four-byte tag to the packet. The tag contains the tag value, which identifies the data as a tag, and also contains the VLAN ID of the VLAN from which the packet is sent.

- The default tag value is 8100 (hexadecimal). This value comes from the 802.1Q specification. You can change this tag value on a global basis on Foundry devices if needed to be compatible with other vendors’ equipment.
- The VLAN ID is determined by the VLAN on which the packet is being forwarded.

Figure 11.4 shows the format of packets with and without the 802.1Q tag. The tag format is vendor-specific. To use the tag for VLANs configured across multiple devices, make sure all the devices support the same tag format.

Figure 11.4 Packet containing Foundry’s 802.1QVLAN tag

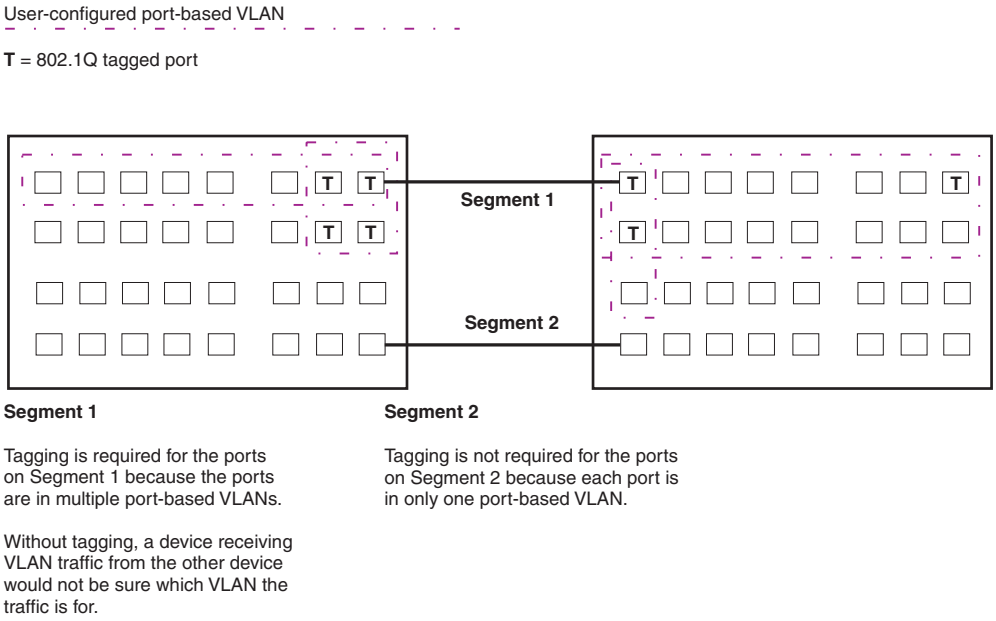


If you configure a VLAN that spans multiple devices, you need to use tagging only if a port connecting one of the devices to the other is a member of more than one port-based VLAN. If a port connecting one device to the other is a member of only a single port-based VLAN, tagging is not required.

If you use tagging on multiple devices, each device must be configured for tagging and must use the same tag value. In addition, the implementation of tagging must be compatible on the devices. The tagging on all Foundry devices is compatible with other Foundry devices.

Figure 11.5 shows an example of two devices that have the same Layer 2 port-based VLANs configured across them. Notice that only one of the VLANs requires tagging.

Figure 11.5 VLANs configured across multiple devices



802.1Q-in-Q Tagging

Foundry devices provide finer granularity for configuring 802.1Q tagging, enabling you to configure 802.1Q tag-types on a group of ports, thereby enabling the creation of two identical 802.1Q tags (802.1Q-in-Q tagging) on a single device. This enhancement improves SAV interoperability between Foundry devices and other vendors' devices that support the 802.1Q tag-types, but are not very flexible with the tag-types they accept.

For example applications and configuration details, see "Configuring 802.1Q-in-Q Tagging" on page 11-49.

Spanning Tree Protocol (STP)

The default state of STP depends on the device type:

- STP is disabled by default on Foundry Layer 3 Switches.
- STP is enabled by default on Foundry Layer 2 Switches.

Also by default, each port-based VLAN has a separate instance of STP. Thus, when STP is globally enabled, each port-based VLAN on the device runs a separate spanning tree.

You can enable or disable STP on the following levels:

- Globally – Affects all ports on the device.

  NOTE: If you configure a port-based VLAN on the device, the VLAN has the same STP state as the default STP state on the device. Thus, on Layer 2 Switches, new VLANs have STP enabled by default. On Layer 3 Switches, new VLANs have STP disabled by default. You can enable or disable STP in each VLAN separately. In addition, you can enable or disable STP on individual ports.

- Port-based VLAN – Affects all ports within the specified port-based VLAN.

STP is a Layer 2 protocol. Thus, you cannot enable or disable STP for individual protocol VLANs or for IP sub-net, IPX network, or AppleTalk cable VLANs. The STP state of a port-based VLAN containing these other types of

VLANs determines the STP state for all the Layer 2 broadcasts within the port-based VLAN. This is true even though Layer 3 protocol broadcasts are sent on Layer 2 within the VLAN.

It is possible that STP will block one or more ports in a protocol VLAN that uses a virtual routing interface to route to other VLANs. For IP protocol and IP sub-net VLANs, even though some of the physical ports of the virtual routing interface are blocked, the virtual routing interface can still route so long as at least one port in the virtual routing interface's protocol VLAN is not blocked by STP.

If you enable Single STP (SSTP) on the device, the ports in all VLANs on which STP is enabled become members of a single spanning tree. The ports in VLANs on which STP is disabled are excluded from the single spanning tree.

For more information, see "Configuring Spanning Tree Protocol (STP) and IronSpan Features" on page 7-1.

Virtual Routing Interfaces

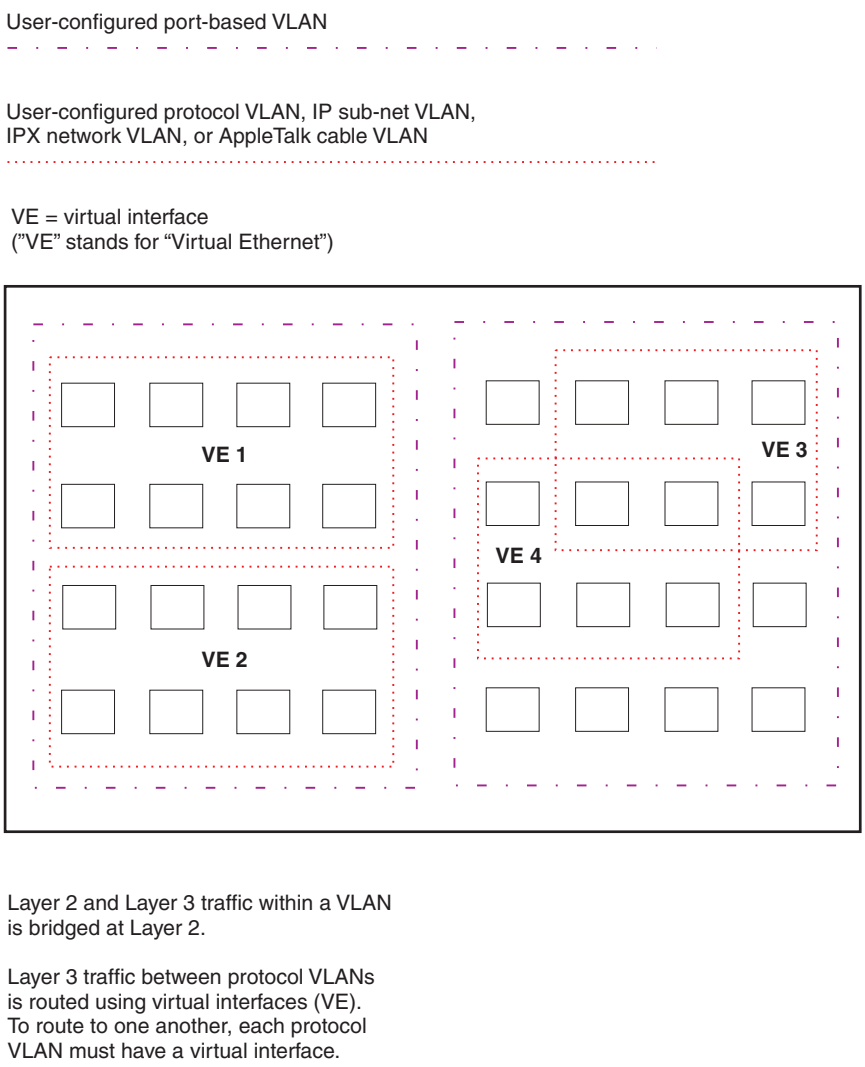
A virtual routing interface is a logical routing interface that Foundry Layer 3 Switches use to route Layer 3 protocol traffic between protocol VLANs.

Foundry devices send Layer 3 traffic at Layer 2 within a protocol VLAN. However, Layer 3 traffic from one protocol VLAN to another must be routed.

If you want the device to be able to send Layer 3 traffic from one protocol VLAN to another, you must configure a virtual routing interface on each protocol VLAN, then configure routing parameters on the virtual routing interfaces. For example, to enable a FastIron Layer 3 Switch to route IP traffic from one IP sub-net VLAN to another, you must configure a virtual routing interface on each IP sub-net VLAN, then configure the appropriate IP routing parameters on each of the virtual routing interfaces.

Figure 11.6 shows an example of Layer 3 protocol VLANs that use virtual routing interfaces for routing.

Figure 11.6 Use virtual routing interfaces for routing between Layer 3 protocol VLANs



VLAN and Virtual Routing Interface Groups

To simplify configuration, you can configure VLAN groups and virtual routing interface groups. When you create a VLAN group, the VLAN parameters you configure for the group apply to all the VLANs within the group. Additionally, you can easily associate the same IP sub-net interface with all the VLANs in a group by configuring a virtual routing interface group with the same ID as the VLAN group.

For configuration information, see "Configuring VLAN Groups and Virtual Routing Interface Groups" on page 11-40.

Dynamic, Static, and Excluded Port Membership

When you add ports to a protocol VLAN, IP sub-net VLAN, IPX network VLAN, or AppleTalk cable VLAN, you can add them dynamically or statically:

- Dynamic ports
- Static ports

You also can explicitly exclude ports.

Dynamic Ports

Dynamic ports are added to a VLAN when you create the VLAN. However, if a dynamically added port does not receive any traffic for the VLAN's protocol within ten minutes, the port is removed from the VLAN. However, the port remains a candidate for port membership. Thus, if the port receives traffic for the VLAN's protocol, the device adds the port back to the VLAN.

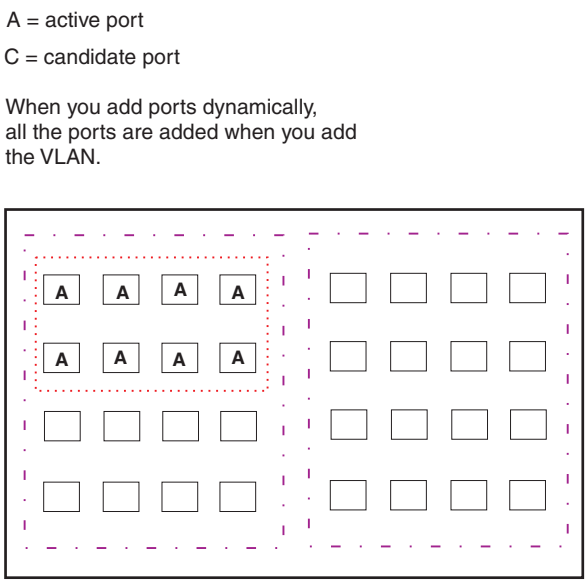
After the port is added back to the VLAN, the port can remain an active member of the VLAN up to 20 minutes without receiving traffic for the VLAN's protocol. If the port ages out, it remains a candidate for VLAN membership and is added back to the VLAN when the VLAN receives protocol traffic. At this point, the port can remain in the VLAN up to 20 minutes without receiving traffic for the VLAN's protocol, and so on.

Unless you explicitly add a port statically or exclude a port, the port is a dynamic port and thus can be an active member of the VLAN, depending on the traffic it receives.

NOTE: You cannot configure dynamic ports in an AppleTalk cable VLAN. The ports in an AppleTalk cable VLAN must be static. However, ports in an AppleTalk protocol VLAN can be dynamic or static.

Figure 11.7 shows an example of a VLAN with dynamic ports. Dynamic ports not only join and leave the VLAN according to traffic, but also allow some broadcast packets of the specific protocol to "leak" through the VLAN. See "Broadcast Leaks" on page 11-12.

Figure 11.7 VLAN with dynamic ports—all ports are active when you create the VLAN

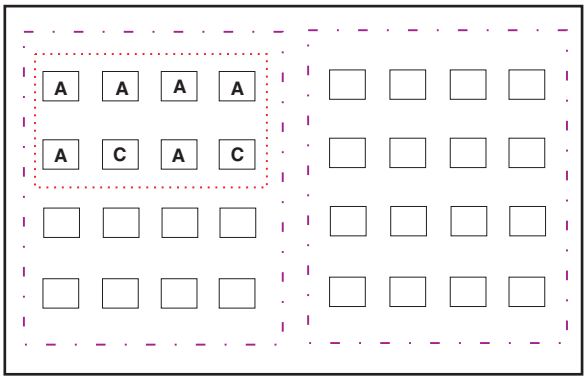


Ports in a new protocol VLAN that do not receive traffic for the VLAN's protocol age out after 10 minutes and become candidate ports. Figure 11.8 shows what happens if a candidate port receives traffic for the VLAN's protocol.

Figure 11.8 VLAN with dynamic ports—candidate ports become active again if they receive protocol traffic

Ports that time out remain candidates for membership in the VLAN and become active again if they receive traffic for the VLAN's protocol, IP sub-net, IPX network, or AppleTalk cable range.

When a candidate port rejoins a VLAN, the timeout for that port becomes 20 minutes. Thus, the port remains an active member of the VLAN even if it does not receive traffic for 20 minutes. After that, the port becomes a candidate port again.



Static Ports

Static ports are permanent members of the protocol VLAN. The ports remain active members of the VLAN regardless of whether the ports receive traffic for the VLAN's protocol. You must explicitly identify the port as a static port when you add it to the VLAN. Otherwise, the port is dynamic and is subject to aging out.

Excluded Ports

If you want to prevent a port in a port-based VLAN from ever becoming a member of a protocol, IP sub-net, IPX network, or AppleTalk cable VLAN configured in the port-based VLAN, you can explicitly exclude the port. You exclude the port when you configure the protocol, IP sub-net, IPX network, or AppleTalk cable VLAN.

Excluded ports do not leak broadcast packets. See "Broadcast Leaks" on page 11-12.

Broadcast Leaks

A dynamic port becomes a member of a Layer 3 protocol VLAN when traffic from the VLAN's protocol is received on the port. After this point, the port remains an active member of the protocol VLAN, unless the port does not receive traffic from the VLAN's protocol for 20 minutes. If the port does not receive traffic for the VLAN's protocol for 20 minutes, the port ages out and is no longer an active member of the VLAN.

To enable a host that has been silent for awhile to send and receive packets, the dynamic ports that are currently members of the Layer 3 protocol VLAN "leak" Layer 3 broadcast packets to the ports that have aged out. When a host connected to one of the aged out ports responds to a leaked broadcast, the port is added to the protocol VLAN again.

To "leak" Layer 3 broadcast traffic, an active port sends 1/8th of the Layer 3 broadcast traffic to the inactive (aged out) ports.

Static ports do not age out and do not leak broadcast packets.

Super Aggregated VLANs

You can aggregate multiple VLANs within another VLAN. This feature allows you to construct Layer 2 paths and channels. This feature is particularly useful for Virtual Private Network (VPN) applications in which you need to provide a private, dedicated Ethernet connection for an individual client to transparently reach its sub-net across multiple networks.

For an application example and configuration information, see “Configuring Super Aggregated VLANs” on page 11-43.

Trunk Group Ports and VLAN Membership

A trunk group is a set of physical ports that are configured to act as a single physical interface. Each trunk group’s port configuration is based on the configuration of the lead port, which is the lowest numbered port in the group.

If you add a trunk group’s lead port to a VLAN, all of the ports in the trunk group become members of that VLAN.

Summary of VLAN Configuration Rules

A hierarchy of VLANs exists between the Layer 2 and Layer 3 protocol-based VLANs:

- Port-based VLANs are at the lowest level of the hierarchy.
- Layer 3 protocol-based VLANs, IP, IPv6, IPX, AppleTalk, Decnet, and NetBIOS are at the middle level of the hierarchy.
- IP sub-net, IPX network, and AppleTalk cable VLANs are at the top of the hierarchy.

NOTE: You cannot have a protocol-based VLAN and a sub-net or network VLAN of the same protocol type in the same port-based VLAN. For example, you can have an IPX protocol VLAN and IP sub-net VLAN in the same port-based VLAN, but you cannot have an IP protocol VLAN and an IP sub-net VLAN in the same port-based VLAN, nor can you have an IPX protocol VLAN and an IPX network VLAN in the same port-based VLAN.

As a Foundry device receives packets, the VLAN classification starts from the highest level VLAN first. Therefore, if an interface is configured as a member of both a port-based VLAN and an IP protocol VLAN, IP packets coming into the interface are classified as members of the IP protocol VLAN because that VLAN is higher in the VLAN hierarchy.

Multiple VLAN Membership Rules

- A port can belong to multiple, unique, overlapping Layer 3 protocol-based VLANs without VLAN tagging.
- A port can belong to multiple, overlapping Layer 2 port-based VLANs only if the port is a tagged port. Packets sent out of a tagged port use an 802.1Q-tagged frame.
- When both port and protocol-based VLANs are configured on a given device, all protocol VLANs must be strictly contained within a port-based VLAN. A protocol VLAN cannot include ports from multiple port-based VLANs. This rule is required to ensure that port-based VLANs remain loop-free Layer 2 broadcast domains.
- IP protocol VLANs and IP sub-net VLANs cannot operate concurrently on the system or within the same port-based VLAN.
- IPX protocol VLANs and IPX network VLANs cannot operate concurrently on the system or within the same port-based VLAN.
- If you first configure IP and IPX protocol VLANs before deciding to partition the network by IP sub-net and IPX network VLANs, then you need to delete those VLANs before creating the IP sub-net and IPX network VLANs.
- One of each type of protocol VLAN is configurable within each port-based VLAN on the Layer 2 Switch.
- Multiple IP sub-net and IPX network VLANs are configurable within each port-based VLAN on the Layer 2 Switch.

- Removing a configured port-based VLAN from a Foundry Networks Layer 2 Switch or Layer 3 Switch automatically removes any protocol-based VLAN, IP sub-net VLAN, AppleTalk cable VLAN, or IPX network VLAN, or any Virtual Ethernet router interfaces defined within the Port-based VLAN.

Routing Between VLANs

Foundry Layer 3 Switches can locally route IP, IPX, and Appletalk between VLANs defined within a single router. All other routable protocols or protocol VLANs (for example, DecNet) must be routed by another external router capable of routing the protocol.

Virtual Routing Interfaces (Layer 3 Switches Only)

You need to configure virtual routing interfaces if an IP, IPX, or Appletalk protocol VLAN, IP sub-net VLAN, AppleTalk cable VLAN, or IPX network VLAN needs to route protocols to another port-based VLAN on the same router. A virtual routing interface can be associated with the ports in only a single port-based VLAN. Virtual router interfaces must be defined at the highest level of the VLAN hierarchy.

If you do not need to further partition the port-based VLAN by defining separate Layer 3 VLANs, you can define a single virtual routing interface at the port-based VLAN level and enable IP, IPX, and Appletalk routing on a single virtual routing interface.

Bridging and Routing the Same Protocol Simultaneously on the Same Device (Layer 3 Switches Only)

Some configurations may require simultaneous switching and routing of the same single protocol across different sets of ports on the same router. When IP, IPX, or Appletalk routing is enabled on a Foundry Layer 3 Switch, you can route these protocols on specific interfaces while bridging them on other interfaces. In this scenario, you can create two separate backbones for the same protocol, one bridged and one routed.

To bridge IP, IPX, or Appletalk at the same time these protocols are being routed, you need to configure an IP protocol, IP sub-net, IPX protocol, IPX network, or Appletalk protocol VLAN and not assign a virtual routing interface to the VLAN. Packets for these protocols are bridged or switched at Layer 2 across ports on the router that are included in the Layer 3 VLAN. If these VLANs are built within port-based VLANs, they can be tagged across a single set of backbone fibers to create separate Layer 2 switched and Layer 3 routed backbones for the same protocol on a single physical backbone.

Routing Between VLANs Using Virtual Routing Interfaces (Layer 3 Switches Only)

Foundry calls the ability to route between VLANs with virtual routing interfaces ***Integrated Switch Routing (ISR)***. There are some important concepts to understand before designing an ISR backbone.

Virtual router interfaces can be defined on port-based, IP protocol, IP sub-net, IPX protocol, IPX network, AppleTalk protocol, and AppleTalk cable VLANs.

To create any type of VLAN on a Foundry Layer 3 Switch, Layer 2 forwarding must be enabled. When Layer 2 forwarding is enabled, the Layer 3 Switch becomes a Switch on all ports for all non-routable protocols.

If the router interfaces for IP, IPX, or AppleTalk are configured on physical ports, then routing occurs independent of the Spanning Tree Protocol (STP). However, if the router interfaces are defined for any type VLAN, they are virtual routing interfaces and are subject to the rules of STP.

If your backbone consists of virtual routing interfaces all within the same STP domain, it is a bridged backbone, not a routed one. This means that the set of backbone interfaces that are blocked by STP will be blocked for routed protocols as well. The routed protocols will be able to cross these paths only when the STP state of the link is FORWARDING. This problem is easily avoided by proper network design.

When designing an ISR network, pay attention to your use of virtual routing interfaces and the spanning-tree domain. If Layer 2 switching of your routed protocols (IP, IPX, AppleTalk) is not required across the backbone, then the use of virtual routing interfaces can be limited to edge switch ports within each router. Full backbone routing can be achieved by configuring routing on each physical interface that connects to the backbone. Routing is independent of STP when configured on a physical interface.

If your ISR design requires that you switch IP, IPX, or Appletalk at Layer 2 while simultaneously routing the same protocols over a single backbone, then create multiple port-based VLANs and use VLAN tagging on the backbone links to separate your Layer 2 switched and Layer 3 routed networks.

There is a separate STP domain for each port-based VLAN. Routing occurs independently across port-based VLANs or STP domains. You can define each end of each backbone link as a separate tagged port-based VLAN. Routing will occur independently across the port-based VLANs. Because each port-based VLAN's STP domain is a single point-to-point backbone connection, you are guaranteed to never have an STP loop. STP will never block the virtual router interfaces within the tagged port-based VLAN, and you will have a fully routed backbone.

Dynamic Port Assignment (Layer 2 Switches and Layer 3 Switches)

All Switch ports are dynamically assigned to any Layer 3 VLAN on Foundry Layer 2 Switches and any non-routable VLAN on Foundry Layer 3 Switches. To maintain explicit control of the VLAN, you can explicitly exclude ports when configuring any Layer 3 VLAN on a Foundry Layer 2 Switch or any non-routable Layer 3 VLAN on a Foundry Layer 3 Switch.

If you do not want the ports to have dynamic membership, you can add them statically. This eliminates the need to explicitly exclude the ports that you do not want to participate in a particular Layer 3 VLAN.

Assigning a Different VLAN ID to the Default VLAN

When you enable port-based VLANs, all ports in the system are added to the default VLAN. By default, the default VLAN ID is "VLAN 1". The default VLAN is not configurable. If you want to use the VLAN ID "VLAN 1" as a configurable VLAN, you can assign a different VLAN ID to the default VLAN.

To reassign the default VLAN to a different VLAN ID, enter the following command:

```
FastIron SuperX Router(config)# default-vlan-id 4095
```

Syntax: [no] default-vlan-d <vlan-id>

You must specify a valid VLAN ID that is not already in use. For example, if you have already defined VLAN 10, do not try to use "10" as the new VLAN ID for the default VLAN. Valid VLAN IDs are numbers from 1 – 4096.

NOTE: Changing the default VLAN name does not change the properties of the default VLAN. Changing the name allows you to use the VLAN ID "1" as a configurable VLAN.

Assigning Trunk Group Ports

When a "lead" trunk group port is assigned to a VLAN, all other members of the trunk group are automatically added to that VLAN. A lead port is the first port of a trunk group port range; for example, "1" in 1 – 4 or "5" in 5 – 8. See "Trunk Group Rules" on page 10-3 for more information.

Configuring Port-Based VLANs

Port-based VLANs allow you to provide separate spanning tree protocol (STP) domains or broadcast domains on a port-by-port basis.

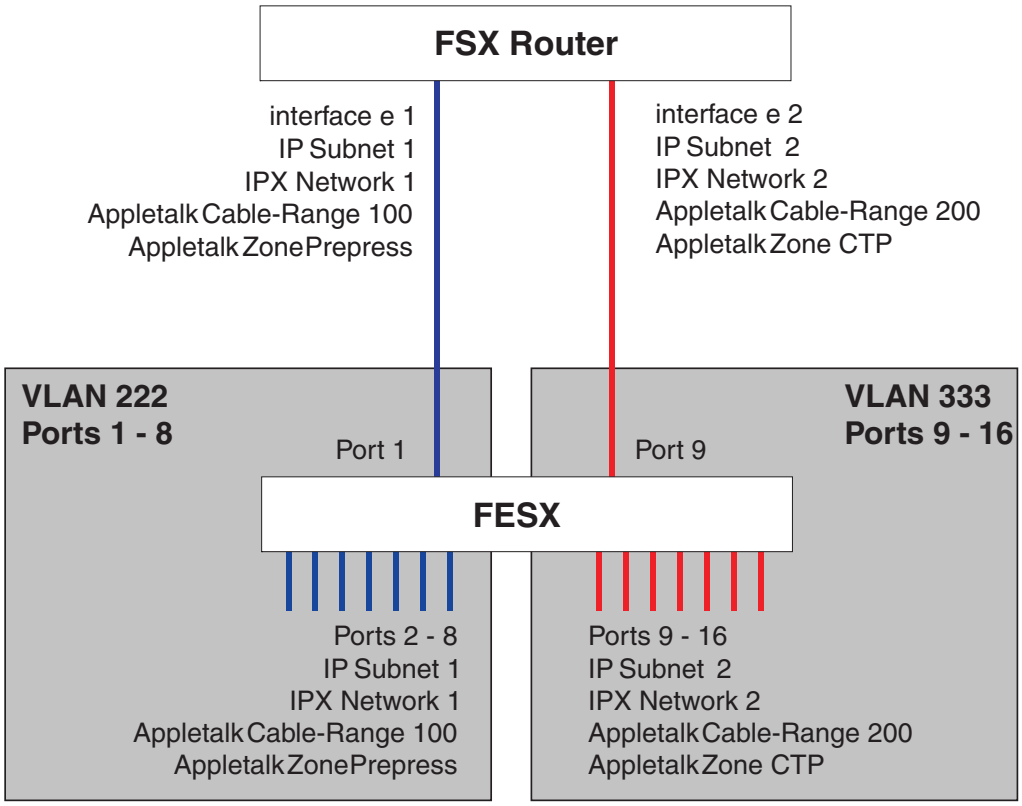
This section describes how to perform the following tasks for port-based VLANs using the CLI:

- Create a VLAN
- Delete a VLAN
- Modify a VLAN
- Change a VLAN's priority
- Enable or disable STP on the VLAN

EXAMPLE: 1

Figure 11.9 shows a simple port-based VLAN configuration using a single Foundry Layer 2 Switch. All ports within each VLAN are untagged. One untagged port within each VLAN is used to connect the Layer 2 Switch to a Layer 3 Switch (in this example, a FSX) for Layer 3 connectivity between the two port-based VLANs.

Figure 11.9 Port-based VLANs 222 and 333



To create the two port-based VLANs shown in Figure 11.9, enter the following commands:

```
FESX424 Switch(config)# vlan 222 by port
FESX424 Switch(config-vlan-222)# untag e 1 to 8
FESX424 Switch(config-vlan-222)# vlan 333 by port
FESX424 Switch(config-vlan-333)# untag e 9 to 16
```

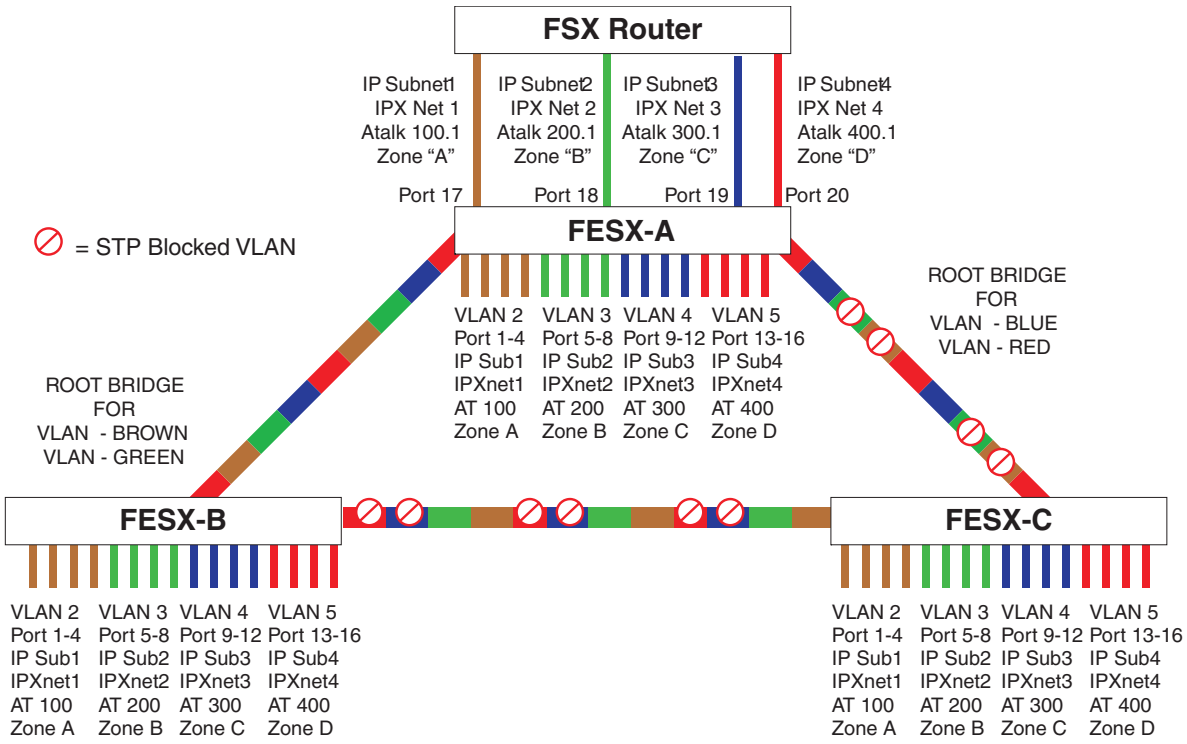
Syntax: vlan <vlan-id> by port

Syntax: untagged ethernet [<slotnum>/]<portnum> [to [<slotnum>/]<portnum> | ethernet [<slotnum>/]<portnum>]

EXAMPLE: 2

Figure 11.10 shows a more complex port-based VLAN configuration using multiple Layer 2 Switches and IEEE 802.1Q VLAN tagging. The backbone link connecting the three Layer 2 Switches is tagged. One untagged port within each port-based VLAN on FESX-A connects each separate network wide Layer 2 broadcast domain to the router for Layer 3 forwarding between broadcast domains. The STP priority is configured to force FESX-A to be the root bridge for VLANs RED and BLUE. The STP priority on FESX-B is configured so that FESX-B is the root bridge for VLANs GREEN and BROWN.

Figure 11.10 More complex port-based VLAN



To configure the Port-based VLANs on the FESX Layer 2 Switches in Figure 11.10, use the following method.

Configuring FESX-A

Enter the following commands to configure FESX-A:

```
FESX424 Switch> enable
FESX424 Switch# configure terminal
FESX424 Switch(config)# hostname FESX-A
FESX424 Switch-A(config)# vlan 2 name BROWN
FESX424 Switch-A(config-vlan-2)# untag ethernet 1 to 4 ethernet 17
FESX424 Switch-A(config-vlan-2)# tag ethernet 25 to 26
FESX424 Switch-A(config-vlan-2)# spanning-tree
FESX424 Switch-A(config-vlan-2)# vlan 3 name GREEN
FESX424 Switch-A(config-vlan-3)# untag ethernet 5 to 8 ethernet 18
FESX424 Switch-A(config-vlan-3)# tag ethernet 25 to 26
FESX424 Switch-A(config-vlan-3)# spanning-tree
FESX424 Switch-A(config-vlan-3)# vlan 4 name BLUE
FESX424 Switch-A(config-vlan-4)# untag ethernet 9 to 12 ethernet 19
FESX424 Switch-A(config-vlan-4)# tag ethernet 25 to 26
FESX424 Switch-A(config-vlan-4)# spanning-tree
FESX424 Switch-A(config-vlan-4)# spanning-tree priority 500
FESX424 Switch-A(config-vlan-4)# vlan 5 name RED
FESX424 Switch-A(config-vlan-5)# untag ethernet 13 to 16 ethernet 20
FESX424 Switch-A(config-vlan-5)# tag ethernet 25 to 26
FESX424 Switch-A(config-vlan-5)# spanning-tree
FESX424 Switch-A(config-vlan-5)# spanning-tree priority 500
FESX424 Switch-A(config-vlan-5)# end
FESX424 Switch-A# write memory
```

Configuring FESX-B

Enter the following commands to configure FESX-B:

```
FESX424 Switch> en
FESX424 Switch# configure terminal
FESX424 Switch(config)# hostname FESX-B
FESX424 Switch-B(config)# vlan 2 name BROWN
FESX424 Switch-B(config-vlan-2)# untag ethernet 1 to 4
FESX424 Switch-B(config-vlan-2)# tag ethernet 25 to 26
FESX424 Switch-B(config-vlan-2)# spanning-tree
FESX424 Switch-B(config-vlan-2)# spanning-tree priority 500
FESX424 Switch-B(config-vlan-2)# vlan 3 name GREEN
FESX424 Switch-B(config-vlan-3)# untag ethernet 5 to 8
FESX424 Switch-B(config-vlan-3)# tag ethernet 25 to 26
FESX424 Switch-B(config-vlan-3)# spanning-tree
FESX424 Switch-B(config-vlan-3)# spanning-tree priority 500
FESX424 Switch-B(config-vlan-3)# vlan 4 name BLUE
FESX424 Switch-B(config-vlan-4)# untag ethernet 9 to 12
FESX424 Switch-B(config-vlan-4)# tag ethernet 25 to 26
FESX424 Switch-B(config-vlan-4)# vlan 5 name RED
FESX424 Switch-B(config-vlan-5)# untag ethernet 13 to 16
FESX424 Switch-B(config-vlan-5)# tag ethernet 25 to 26
FESX424 Switch-B(config-vlan-5)# end
FESX424 Switch-B# write memory
```

Configuring FESX-C

Enter the following commands to configure FESX-C:

```
FESX424 Switch> en
FESX424 Switch# configure terminal
FESX424 Switch(config)# hostname FESX-C
FESX424 Switch-C(config)# vlan 2 name BROWN
FESX424 Switch-C(config-vlan-2)# untag ethernet 1 to 4
FESX424 Switch-C(config-vlan-2)# tag ethernet 25 to 26
FESX424 Switch-C(config-vlan-2)# vlan 3 name GREEN
FESX424 Switch-C(config-vlan-3)# untag ethernet 5 to 8
FESX424 Switch-C(config-vlan-3)# tag ethernet 25 to 26
FESX424 Switch-C(config-vlan-3)# vlan 4 name BLUE
FESX424 Switch-C(config-vlan-4)# untag ethernet 9 to 12
FESX424 Switch-C(config-vlan-4)# tag ethernet 25 to 26
FESX424 Switch-C(config-vlan-4)# vlan 5 name RED
FESX424 Switch-C(config-vlan-5)# untag ethernet 13 to 16
FESX424 Switch-C(config-vlan-5)# tag ethernet 25 to 26
FESX424 Switch-C(config-vlan-5)# end
FESX424 Switch-C# write memory
```

Syntax: vlan <vlan-id> by port

Syntax: untagged ethernet [<slotnum>/]<portnum> [to [<slotnum>/]<portnum> | ethernet [<slotnum>/]<portnum>]

Syntax: tagged ethernet [<slotnum>/]<portnum> [to <[<slotnum>/]portnum> | ethernet [<slotnum>/]<portnum>]

Syntax: [no] spanning-tree

Syntax: spanning-tree [ethernet [<slotnum>/]<portnum> path-cost <value> priority <value>] forward-delay <value> hello-time <value> maximum-age <time> priority <value>

Modifying a Port-Based VLAN

You can make the following modifications to a port-based VLAN:

- Add or delete a VLAN port.

- Enable or disable STP.

Removing a Port-Based VLAN

Suppose you want to remove VLAN 5 from the example in Figure 11.10. To do so, use the following procedure.

1. Access the global CONFIG level of the CLI on FESX-A by entering the following commands:

```
FESX424 Switch-A> enable
No password has been assigned yet...
FESX424 Switch-A# configure terminal
FESX424 Switch-A(config)#
```

2. Enter the following command:

```
FESX424 Switch-A(config)# no vlan 5
FESX424 Switch-A(config)#
```

3. Enter the following commands to exit the CONFIG level and save the configuration to the system-config file on flash memory:

```
FESX424 Switch-A(config)#
FESX424 Switch-A(config)# end
FESX424 Switch-A# write memory
FESX424 Switch-A#
```

4. Repeat steps 1 – 3 on FESX-B.

Syntax: no vlan <vlan-id> by port

Removing a Port from a VLAN

Suppose you want to remove port 11 from VLAN 4 on FESX-A shown in Figure 11.10. To do so, use the following procedure.

1. Access the global CONFIG level of the CLI on FESX424 Switch-A by entering the following command:

```
FESX424 Switch-A> enable
No password has been assigned yet...
FESX424 Switch-A# configure terminal
FESX424 Switch-A(config)#
```

2. Access the level of the CLI for configuring port-based VLAN 4 by entering the following command:

```
FESX424 Switch-A(config)#
FESX424 Switch-A(config)# vlan 4
FESX424 Switch-A(config-vlan-4)#
```

3. Enter the following commands:

```
FESX424 Switch-A(config-vlan-4)#
FESX424 Switch-A(config-vlan-4)# no untag ethernet 11
deleted port ethe 11 from port-vlan 4.
FESX424 Switch-A(config-vlan-4)#
```

4. Enter the following commands to exit the VLAN CONFIG mode and save the configuration to the system-config file on flash memory:

```
FESX424 Switch-A(config-vlan-4)#
FESX424 Switch-A(config-vlan-4)# end
FESX424 Switch-A# write memory
```

You can remove all the ports from a port-based VLAN without losing the rest of the VLAN's configuration. However, you cannot configure an IP address on a virtual routing interface unless the VLAN contains ports. If the

VLAN has a virtual routing interface, the virtual routing interface's IP address is deleted when the ports associated with the interface are deleted. The rest of the VLAN configuration is retained.

Enable Spanning Tree on a VLAN

The spanning tree bridge and port parameters are configurable using one CLI command set at the Global Configuration Level of each Port-based VLAN. Suppose you want to enable the IEEE 802.1d STP across VLAN 3. To do so, use the following method.

NOTE: When port-based VLANs are not operating on the system, STP is set on a system-wide level at the global CONFIG level of the CLI.

1. Access the global CONFIG level of the CLI on FESX-A by entering the following commands:

```
FESX424 Switch-A> enable
No password has been assigned yet...
FESX424 Switch-A# configure terminal
FESX424 Switch-A(config)#
```

2. Access the level of the CLI for configuring port-based VLAN 3 by entering the following command:

```
FESX424 Switch-A(config)#
FESX424 Switch-A(config)# vlan 3
FESX424 Switch-A(config-vlan-3)#
```

3. From VLAN 3's configuration level of the CLI, enter the following command to enable STP on all tagged and untagged ports associated with VLAN 3.

```
FESX424 Switch-B(config-vlan-3)#
FESX424 Switch-B(config-vlan-3)# spanning-tree
FESX424 Switch-B(config-vlan-3)#
```

4. Enter the following commands to exit the VLAN CONFIG mode and save the configuration to the system-config file on flash memory:

```
FESX424 Switch-B(config-vlan-3)#
FESX424 Switch-B(config-vlan-3)# end
FESX424 Switch-B# write memory
FESX424 Switch-B#
```

5. Repeat steps 1 – 4 on FESX-B.

NOTE: You do not need to configure values for the STP parameters. All parameters have default values as noted below. Additionally, all values will be globally applied to all ports on the system or on the port-based VLAN for which they are defined.

To configure a specific path-cost or priority value for a given port, enter those values using the key words in the brackets [ ] shown in the syntax summary below. If you do not want to specify values for any given port, this portion of the command is not required.

Syntax: vlan <vlan-id> by port

Syntax: [no] spanning-tree

Syntax: spanning-tree [ethernet [<slotnum>/]<portnum> path-cost <value> priority <value>] forward-delay <value> hello-time <value> maximum-age <time> priority <value>

Bridge STP Parameters (applied to all ports within a VLAN)

- Forward Delay – the period of time a bridge will wait (the listen and learn period) before forwarding data packets. Possible values: 4 – 30 seconds. Default is 15.
- Maximum Age – the interval a bridge will wait for receipt of a hello packet before initiating a topology change. Possible values: 6 – 40 seconds. Default is 20.
- Hello Time – the interval of time between each configuration BPDU sent by the root bridge. Possible values:

1 – 10 seconds. Default is 2.

- Priority – a parameter used to identify the root bridge in a network. The bridge with the lowest value has the highest priority and is the root. Possible values: 1 – 65,535. Default is 32,678.

Port Parameters (applied to a specified port within a VLAN)

- Path Cost – a parameter used to assign a higher or lower path cost to a port. Possible values: 1 – 65535. Default is (1000/Port Speed) for Half-Duplex ports and is (1000/Port Speed)/2 for Full-Duplex ports.
- Priority – value determines when a port will be rerouted in relation to other ports. Possible values: 0 – 255. Default is 128.

Configuring IP Sub-net, IPX Network and Protocol-Based VLANs

Protocol-based VLANs provide the ability to define separate broadcast domains for several unique Layer 3 protocols within a single Layer 2 broadcast domain. Some applications for this feature might include security between departments with unique protocol requirements. This feature enables you to limit the amount of broadcast traffic end-stations, servers, and routers need to accept.

Configuration Example

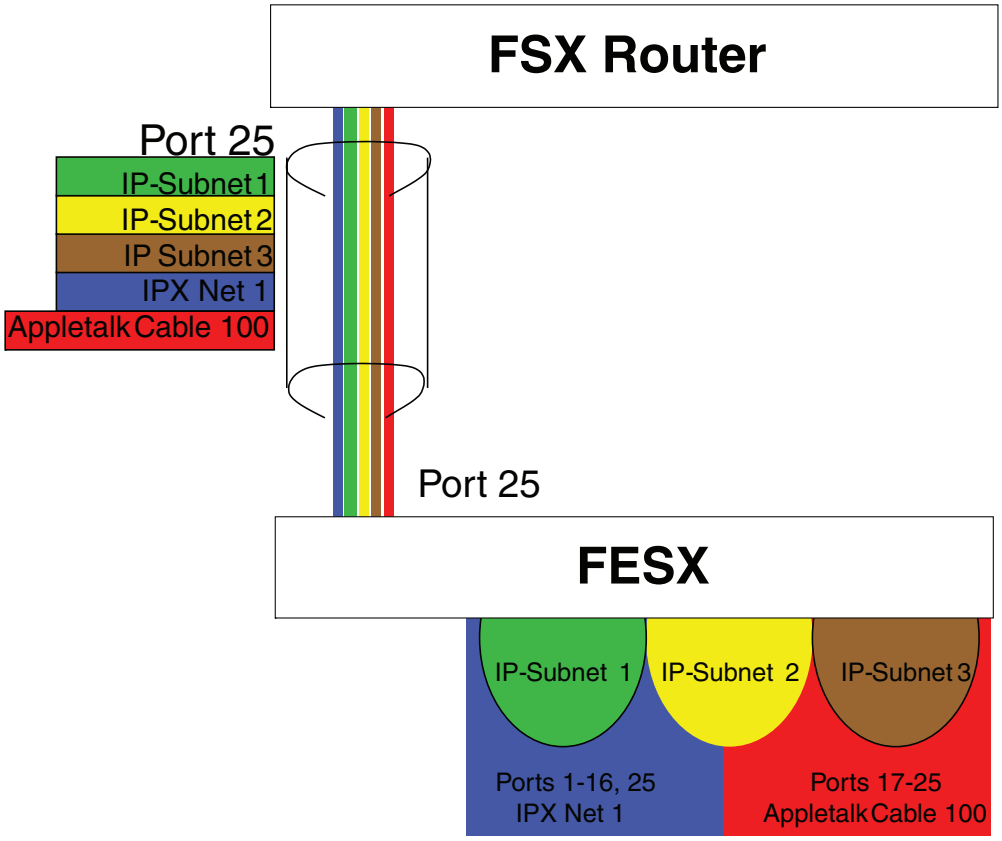
Suppose you want to create five separate Layer 3 broadcast domains within a single Layer 2 STP broadcast domain:

- Three broadcast domains, one for each of three separate IP sub-nets
- One for IPX Network 1
- One for the Appletalk protocol

Also suppose you want a single router interface to be present within all of these separate broadcast domains, without using IEEE 802.1Q VLAN tagging or any proprietary form of VLAN tagging.

Figure 11.11 shows this configuration.

Figure 11.11 Protocol-based (Layer 3) VLANs



To configure the VLANs shown in Figure 11.11, use the following procedure.

1. To permanently assign ports 1 – 8 and port 25 to IP sub-net VLAN 1.1.1.0, enter the following commands:

```
FESX424 Switch> en
No password has been assigned yet...
FESX424 Switch# config t
FESX424 Switch(config)#
FESX424 Switch(config)# ip-subnet 1.1.1.0/24 name Green
FESX424 Switch(config-ip-subnet)# no dynamic
FESX424 Switch(config-ip-subnet)# static ethernet 1 to 8 ethernet 25
```

2. To permanently assign ports 9 – 16 and port 25 to IP sub-net VLAN 1.1.2.0, enter the following commands:

```
FESX424 Switch(config-ip-subnet)# ip-subnet 1.1.2.0/24 name Yellow
FESX424 Switch(config-ip-subnet)# no dynamic
FESX424 Switch(config-ip-subnet)# static ethernet 9 to 16 ethernet 25
```

3. To permanently assign ports 17 – 25 to IP sub-net VLAN 1.1.3.0, enter the following commands:

```
FESX424 Switch(config-ip-subnet)# ip-subnet 1.1.3.0/24 name Brown
FESX424 Switch(config-ip-subnet)# no dynamic
FESX424 Switch(config-ip-subnet)# static ethernet 17 to 25
```

4. To permanently assign ports 1 – 12 and port 25 to IPX network 1 VLAN, enter the following commands:

```
FESX424 Switch(config-ip-subnet)# ipx-network 1 ethernet_802.3 name Blue
FESX424 Switch(config-ipx-network)# no dynamic
```



```
FESX424 Switch(config-ipx-network)# static ethernet 1 to 12 ethernet 25
FESX424 Switch(config-ipx-network)#
```

5. To permanently assign ports 12 – 25 to Appletalk VLAN, enter the following commands:

```
FESX424 Switch(config-ipx-proto)# atalk-proto name Red
FESX424 Switch(config-atalk-proto)# no dynamic
FESX424 Switch(config-atalk-proto)# static ethernet 13 to 25
FESX424 Switch(config-atalk-proto)# end
FESX424 Switch# write memory
FESX424 Switch#
```

Syntax: ip-subnet <ip-addr> <ip-mask> [name <string>]

Syntax: ipx-network <ipx-network-number> <frame-encapsulation-type> netbios-allow | netbios-disallow [name <string>]

Syntax: ip-proto | ipx-proto | atalk-proto | decnet-proto | netbios-proto | other-proto
static | exclude | dynamic
ethernet [<slotnum>/]<portnum> [to [<slotnum>/]<portnum>] [name <string>]

Configuring IP Sub-net, IPX Network, and Protocol-Based VLANs Within Port-Based VLANs

If you plan to use port-based VLANs in conjunction with protocol-based VLANs, you must create the port-based VLANs first. Once you create a port-based VLAN, then you can assign Layer 3 protocol VLANs within the boundaries of the port-based VLAN. Generally, you create port-based VLANs to allow multiple separate STP domains.

EXAMPLE:

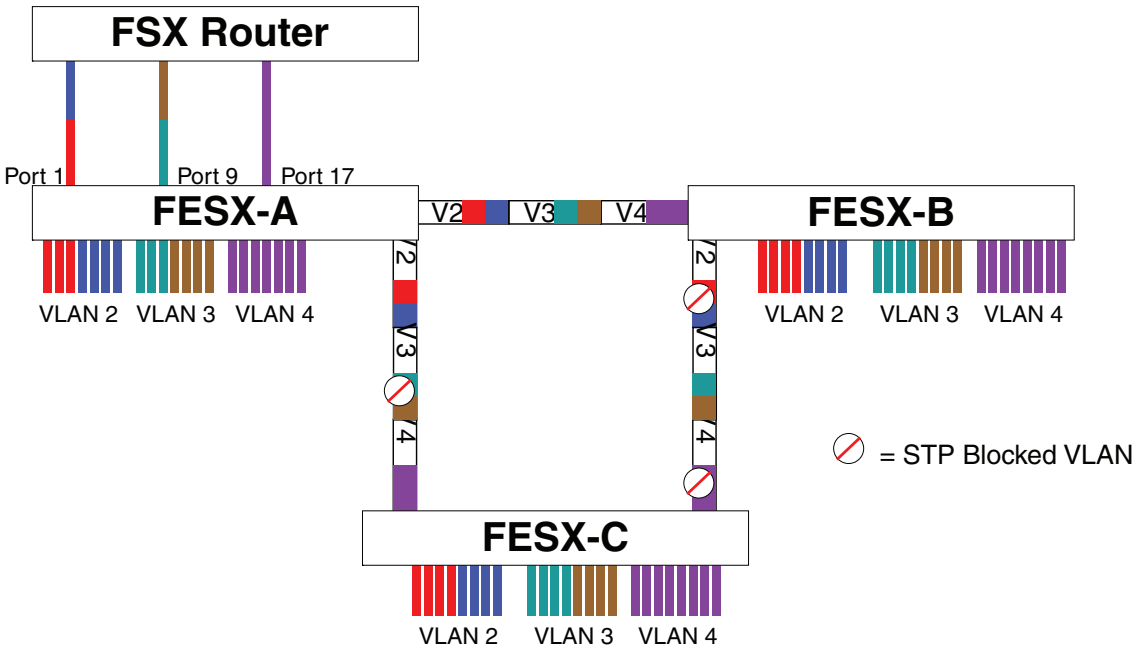
Suppose you need to provide three separate STP domains across an enterprise campus backbone. The first STP domain (VLAN 2) requires a set of ports at each Layer 2 Switch location to be statically mapped to IP only. No other protocols can enter the switches on this set of ports.

A second set of ports within STP domain VLAN 2 will be restricted to only IPX traffic. The IP and IPX protocol VLANs will overlap on Port 1 of FESX-A to support both protocols on the same router interface. The IP sub-nets and IPX network that span the two protocol VLANs will be determined by the NetIron router configuration. The IP and IPX Protocol VLANs ensure that only the ports included in the each Layer 3 protocol VLAN will see traffic from the NetIron router.

The second STP domain (VLAN 3) requires that half the ports in the domain are dedicated to IP sub-net 1.1.1.0/24 and the other ports are dedicated to IPX network 1. Similar to VLAN 2, Port 9 from VLAN 3 will be used to carry this IP sub-net and IPX network to the NetIron router. No other protocols will be allowed to enter the network on VLAN 3. Also, no IP packets with a source address on sub-net 1.1.1.0/24 or IPX packets with a source address on network 1 will be allowed to enter the switches on VLAN 3.

There is no need to segment Layer 3 broadcast domains within the STP broadcast domain (VLAN 4). The NetIron router will dictate the IP sub-nets and IPX network that are on VLAN 4. There are no Layer 3 protocol restrictions on VLAN 4; however, the NetIron router is configured to only forward IP and IPX between STP domains.

Figure 11.12 More protocol-based VLANs



To configure the Layer 3 VLANs on the FESX Layer 2 Switches in Figure 11.12, use the following procedure.

Configuring FESX-A

Enter the following commands to configure FESX-A:

1. Create port-based VLAN 2 and assign the untagged and tagged ports that will participate in this VLAN:

```
FESX424 Switch-A >en
FESX424 Switch-A# config t
FESX424 Switch-A(config)# vlan 2 name IP_IPX_Protocol
FESX424 Switch-A(config-vlan-2)# untag e1 to 8
FESX424 Switch-A(config-vlan-2)# tag e25 to 26
```

2. Enable STP and set the priority to force FESX-A to be the root bridge for VLAN 2:

```
FESX424 Switch-A(config-vlan-2)# spanning-tree
FESX424 Switch-A(config-vlan-2)# spanning-tree priority 500
FESX424 Switch-A(config-vlan-2)#
```

3. Create the IP and IPX protocol-based VLANs and statically assign the ports within VLAN 2 that will be associated with each protocol-based VLAN:

```
FESX424 Switch-A(config-vlan-2)# ip-proto name Red
FESX424 Switch-A(config-vlan-ip-proto)# no dynamic
FESX424 Switch-A(config-vlan-ip-proto)# static e1 to 4 e25 to 26
FESX424 Switch-A(config-vlan-ip-proto)# exclude e5 to 8
FESX424 Switch-A(config-vlan-ip-proto)# ipx-proto name Blue
FESX424 Switch-A(config-vlan-ipx-proto)# no dynamic
FESX424 Switch-A(config-vlan-ipx-proto)# static e1 e5 to 8 e25 to 26
```



```
FESX424 Switch-A(config-vlan-ipx-proto)# exclude e2 to 4
```

4. To prevent machines with non-IP protocols from getting into the IP portion of VLAN 2, create another Layer 3 protocol VLAN to exclude all other protocols from the ports that contains the IP-protocol VLAN. To do so, enter the following commands:

```
FESX424 Switch-A(config-vlan-ipx-proto)# other-proto name Block_other_proto
FESX424 Switch-A(config-vlan-other-proto)# no dynamic
FESX424 Switch-A(config-vlan-other-proto)# exclude e1 to 8
FESX424 Switch-A(config-vlan-other-proto)#
```

5. Create port-based VLAN 3. Note that FESX-B will be the root for this STP domain, so you do not need to adjust the STP priority.

```
FESX424 Switch-A(config-vlan-other-proto)# vlan 3 name IP-Sub_IPX-Net_Vlans
FESX424 Switch-A(config-vlan-3)# untag e9 to 16
FESX424 Switch-A(config-vlan-3)# tag e25 to 26
FESX424 Switch-A(config-vlan-3)# spanning-tree
FESX424 Switch-A(config-vlan-3)#
```

6. Create IP sub-net VLAN 1.1.1.0/24, IPX network 1, and other-protocol VLANs

```
FESX424 Switch-A(config-vlan-3)# ip-subnet 1.1.1.0/24 name Green
FESX424 Switch-A(config-vlan-ip-subnet)# no dynamic
FESX424 Switch-A(config-vlan-ip-subnet)# static e9 to 12 e25 to 26
FESX424 Switch-A(config-vlan-ip-subnet)# exclude e13 to 16
FESX424 Switch-A(config-vlan-ip-subnet)# ipx-net 1 ethernet_802.3 name Brown
FESX424 Switch-A(config-vlan-ipx-network)# no dynamic
FESX424 Switch-A(config-vlan-ipx-network)# static e9 e13 to 16 e25 to 26
FESX424 Switch-A(config-vlan-ipx-network)# exclude e10 to 12
FESX424 Switch-A(config-vlan-ipx-network)# other-proto name Block_other_proto
FESX424 Switch-A(config-vlan-other-proto)# no dynamic
FESX424 Switch-A(config-vlan-other-proto)# exclude e9 to 16
FESX424 Switch-A(config-vlan-other-proto)#
```

7. Configure the last port-based VLAN 4. You need to set the STP priority for this VLAN because FESX-A will be the root bridge for this VLAN. Since you do not need to partition this STP domain into multiple Layer 3 broadcast domains, this is the only configuration required for VLAN 4:

```
FESX424 Switch-A(config-vlan-other-proto)# vlan 4 name Purple_ALL-Protocols
FESX424 Switch-A(config-vlan-4)# untag e17 to 24
FESX424 Switch-A(config-vlan-4)# tag e25 to 26
FESX424 Switch-A(config-vlan-4)# spanning-tree
FESX424 Switch-A(config-vlan-4)# spanning-tree priority 500
FESX424 Switch-A(config-vlan-4)#
```

Configuring FESX-B

Enter the following commands to configure FESX-B:

```
FESX424 Switch# config t
FESX424 Switch(config)# host FESX424 Switch-B
FESX424 Switch-B(config)# vlan 2 name IP_IPX_Protocol
FESX424 Switch-B(config-vlan-2)# untag e1 to 8
FESX424 Switch-B(config-vlan-2)# tag e25 to 26
FESX424 Switch-B(config-vlan-2)# spanning-tree
FESX424 Switch-B(config-vlan-2)# ip-proto name Red
FESX424 Switch-B(config-vlan-ip-proto)# no dynamic
FESX424 Switch-B(config-vlan-ip-proto)# static e1 to 4 e25 to 26
FESX424 Switch-B(config-vlan-ip-proto)# exclude e5 to 8
FESX424 Switch-B(config-vlan-ip-proto)# ipx-proto name Blue
FESX424 Switch-B(config-vlan-ipx-proto)# no dynamic
FESX424 Switch-B(config-vlan-ipx-proto)# static e5 to 8 e25 to 26
FESX424 Switch-B(config-vlan-ipx-proto)# exclude e1 to 4
```

```
FESX424 Switch-B(config-vlan-other-proto)# vlan 3 name IP-Sub_IPX-Net_VLANs
FESX424 Switch-B(config-vlan-3)# untag e9 to 16
FESX424 Switch-B(config-vlan-3)# tag e25 to 26
FESX424 Switch-B(config-vlan-3)# spanning-tree
FESX424 Switch-B(config-vlan-3)# spanning-tree priority 500
FESX424 Switch-B(config-vlan-3)# ip-sub 1.1.1.0/24 name Green
FESX424 Switch-B(config-vlan-ip-subnet)# no dynamic
FESX424 Switch-B(config-vlan-ip-subnet)# static e9 to 12 e25 to 26
FESX424 Switch-B(config-vlan-ip-subnet)# exclude e13 to 16
FESX424 Switch-B(config-vlan-ip-subnet)# ipx-net 1 ethernet_802.3 name Brown
FESX424 Switch-B(config-vlan-ipx-network)# no dynamic
FESX424 Switch-B(config-vlan-ipx-network)# static e13 to 16 e25 to 26
FESX424 Switch-B(config-vlan-ipx-network)# exclude e9 to 12
FESX424 Switch-B(config-vlan-ipx-network)# vlan 4 name Purple_ALL-Protocols
FESX424 Switch-B(config-vlan-4)# untag e17 to 24
FESX424 Switch-B(config-vlan-4)# tag e25 to 26
FESX424 Switch-B(config-vlan-4)# spanning-tree
```

Configuring FESX-C

Enter the following commands to configure FESX-C:

```
FESX424 Switch# config t
FESX424 Switch(config)# host FESX424 Switch-C
FESX424 Switch-C(config)# vlan 2 name IP_IPX_Protocol
FESX424 Switch-C(config-vlan-2)# untag e1 to 8
FESX424 Switch-C(config-vlan-2)# tag e25 to 26
FESX424 Switch-C(config-vlan-2)# spanning-tree
FESX424 Switch-C(config-vlan-2)# ip-proto name Red
FESX424 Switch-C(config-vlan-ip-proto)# no dynamic
FESX424 Switch-C(config-vlan-ip-proto)# static e1 to 4 e25 to 26
FESX424 Switch-C(config-vlan-ip-proto)# exclude e5 to 8
FESX424 Switch-C(config-vlan-ip-proto)# ipx-proto name Blue
FESX424 Switch-C(config-vlan-ipx-proto)# no dynamic
FESX424 Switch-C(config-vlan-ipx-proto)# static e5 to 8 e25 to 26
FESX424 Switch-C(config-vlan-ipx-proto)# exclude e1 to 4
FESX424 Switch-C(config-vlan-other-proto)# vlan 3 name IP-Sub_IPX-Net_VLANs
FESX424 Switch-C(config-vlan-3)# untag e9 to 16
FESX424 Switch-C(config-vlan-3)# tag e25 to 26
FESX424 Switch-C(config-vlan-3)# spanning-tree
FESX424 Switch-C(config-vlan-3)# ip-sub 1.1.1.0/24 name Green
FESX424 Switch-C(config-vlan-ip-subnet)# no dynamic
FESX424 Switch-C(config-vlan-ip-subnet)# static e9 to 12 e25 to 26
FESX424 Switch-C(config-vlan-ip-subnet)# exclude e13 to 16
FESX424 Switch-C(config-vlan-ip-subnet)# ipx-net 1 ethernet_802.3 name Brown
FESX424 Switch-C(config-vlan-ipx-network)# no dynamic
FESX424 Switch-C(config-vlan-ipx-network)# static e13 to 16 e25 to 26
FESX424 Switch-C(config-vlan-ipx-network)# exclude e9 to 12
FESX424 Switch-C(config-vlan-ipx-network)# vlan 4 name Purple_ALL-Protocols
FESX424 Switch-C(config-vlan-4)# untag e17 to 24
FESX424 Switch-C(config-vlan-4)# tag e25 to 26
FESX424 Switch-C(config-vlan-4)# spanning-tree
```

Configuring an IPv6 Protocol VLAN

You can configure a protocol-based VLAN as a broadcast domain for IPv6 traffic. When the Layer 3 Switch receives an IPv6 multicast packet (a packet with 06 in the version field and 0xFF as the beginning of the destination address), the Layer 3 Switch forwards the packet to all other ports in the VLAN.

NOTE: The Layer 3 Switch forwards all IPv6 multicast packets to all ports in the VLAN except the port that received the packet, and does not distinguish among sub-net directed multicasts.

You can add the VLAN ports as static ports or dynamic ports. A static port is always an active member of the VLAN. Dynamic ports within any protocol VLAN age out after 10 minutes, if no member protocol traffic is received on a port within the VLAN. The aged out port, however, remains as a candidate dynamic port for that VLAN. The port becomes active in the VLAN again if member protocol traffic is received on that port.

Once a port is re-activated, the aging out period for the port is reset to 20 minutes. Each time a member protocol packet is received by a candidate dynamic port (aged out port) the port becomes active again and the aging out period is reset for 20 minutes.

To configure an IPv6 VLAN, enter commands such as the following:

```
FastIron SuperX Router(config)# vlan 2
FastIron SuperX Router(config-vlan-2)# untag ethernet 1/1 to 1/8
FastIron SuperX Router(config-vlan-2)# ipv6-proto name V6
FastIron SuperX Router(config-ipv6-subnet)# static ethernet 1/1 to 1/6
FastIron SuperX Router(config-ipv6-subnet)# dynamic
```

The first two commands configure a port-based VLAN and add ports 1/1 – 1/8 to the VLAN. The remaining commands configure an IPv6 VLAN within the port-based VLAN. The **static** command adds ports 1/1 – 1/6 as static ports, which do not age out. The **dynamic** command adds the remaining ports, 1/7 – 1/8, as dynamic ports. These ports are subject to aging as described above.

Syntax: [no] ipv6-proto [name <string>]

Routing Between VLANs Using Virtual Routing Interfaces (Layer 3 Switches Only)

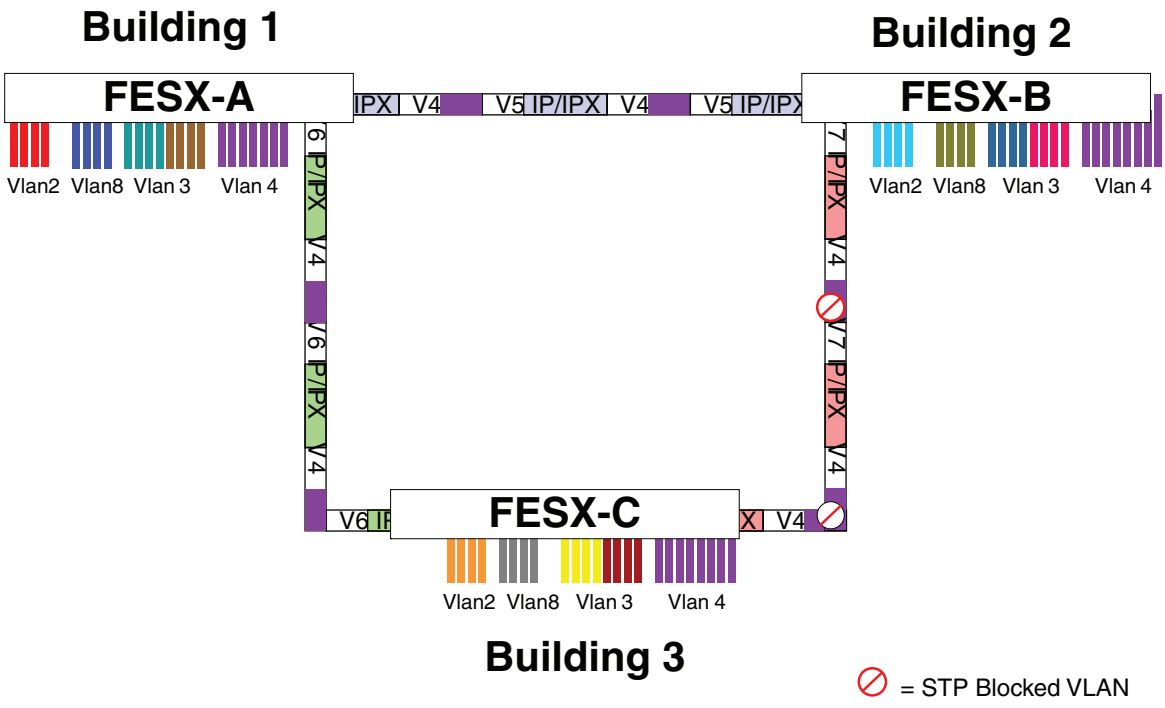
Foundry Layer 3 Switches offer the ability to create a virtual routing interface within a Layer 2 STP port-based VLAN or within each Layer 3 protocol, IP sub-net, or IPX network VLAN. This combination of multiple Layer 2 and/or Layer 3 broadcast domains and virtual routing interfaces are the basis for Foundry Networks' very powerful Integrated Switch Routing (ISR) technology. ISR is very flexible and can solve many networking problems. The following example is meant to provide ideas by demonstrating some of the concepts of ISR.

Example: Suppose you want to move routing out to each of three buildings in a network. Remember that the only protocols present on VLAN 2 and VLAN 3 are IP and IPX. Therefore, you can eliminate tagged ports 25 and 26 from both VLAN 2 and VLAN 3 and create new tagged port-based VLANs to support separate IP sub-nets and IPX networks for each backbone link.

You also need to create unique IP sub-nets and IPX networks within VLAN 2 and VLAN 3 at each building. This will create a fully routed IP and IPX backbone for VLAN 2 and VLAN 3. However, VLAN 4 has no protocol restrictions across the backbone. In fact there are requirements for NetBIOS and DecNet to be bridged among the three building locations. The IP sub-net and IPX network that exists within VLAN 4 must remain a flat Layer 2 switched STP domain. You enable routing for IP and IPX on a virtual routing interface only on NetIron-A. This will provide the flat IP and IPX segment with connectivity to the rest of the network. Within VLAN 4 IP and IPX will follow the STP topology. All other IP sub-nets and IPX networks will be fully routed and have use of all paths at all times during normal operation.

Figure 11.13 shows the configuration described above.

Figure 11.13 Routing between protocol-based VLANs



To configure the Layer 3 VLANs and virtual routing interfaces on the NetIron Layer 3 Switch in Figure 11.13, use the following procedure.

Configuring NetIron-A

Enter the following commands to configure FESX-A. The following commands enable OSPF or RIP routing.

```
FESX424 Router> en
No password has been assigned yet...
FESX424 Router# configure terminal
FESX424 Router(config)# hostname FESX-A
FESX424 Router-A(config)# router ospf
FESX424 Router-A(config-ospf-router)# area 0.0.0.0 normal
Please save configuration to flash and reboot.
FESX424 Router-A(config-ospf-router)#
```

The following commands create the port-based VLAN 2. In the previous example, an external FESX defined the router interfaces for VLAN 2. With ISR, routing for VLAN 2 is done locally within each FESX. Therefore, there are two ways you can solve this problem. One way is to create a unique IP sub-net and IPX network VLAN, each with its own virtual routing interface and unique IP or IPX address within VLAN 2 on each FESX. In this example, this is the configuration used for VLAN 3. The second way is to split VLAN 2 into two separate port-based VLANs and create a virtual router interface within each port-based VLAN. Later in this example, this second option is used to create a port-based VLAN 8 to show that there are multiple ways to accomplish the same task with ISR.

You also need to create the Other-Protocol VLAN within port-based VLAN 2 and 8 to prevent unwanted protocols from being Layer 2 switched within port-based VLAN 2 or 8. Note that the only port-based VLAN that requires STP in this example is VLAN 4. You will need to configure the rest of the network to prevent the need to run STP.

```
FESX424 Router-A(config-ospf-router)# vlan 2 name IP-Subnet_1.1.2.0/24
FESX424 Router-A(config-vlan-2)# untag e 1 to 4
FESX424 Router-A(config-vlan-2)# no spanning-tree
```



```
FESX424 Router-A(config-vlan-2)# router-interface ve1
FESX424 Router-A(config-vlan-2)# other-proto name block_other_protocols
FESX424 Router-A(config-vlan-other-proto)# no dynamic
FESX424 Router-A(config-vlan-other-proto)# exclude e 1 to 4
```

Once you have defined the port-based VLAN and created the virtual routing interface, you need to configure the virtual routing interface just as you would configure a physical interface.

```
FESX424 Router-A(config-vlan-other-proto)# interface ve1
FESX424 Router-A(config-vif-1)# ip address 1.1.2.1/24
FESX424 Router-A(config-vif-1)# ip ospf area 0.0.0.0
```

Do the same thing for VLAN 8.

```
FESX424 Router-A(config-vif-1)# vlan 8 name IPX_Network2
FESX424 Router-A(config-vlan-8)# untag ethernet 5 to 8
FESX424 Router-A(config-vlan-8)# no spanning-tree
FESX424 Router-A(config-vlan-8)# router-interface ve 2
FESX424 Router-A(config-vlan-8)# other-proto name block-other-protocols
FESX424 Router-A(config-vlan-other-proto)# no dynamic
FESX424 Router-A(config-vlan-other-proto)# exclude ethernet 5 to 8
FESX424 Router-A(config-vlan-other-proto)# int ve2
FESX424 Router-A(config-vif-2)# ipx network 2 ethernet_802.3
FESX424 Router-A(config-vif-2)#
```

The next thing you need to do is create VLAN 3. This is very similar to the previous example with the addition of virtual routing interfaces to the IP sub-net and IPX network VLANs. Also there is no need to exclude ports from the IP sub-net and IPX network VLANs on the router.

```
FESX424 Router-A(config-vif-2)# vlan 3 name IP_Sub_&_IPX_Net_VLAN
FESX424 Router-A(config-vlan-3)# untag e 9 to 16
FESX424 Router-A(config-vlan-3)# no spanning-tree
FESX424 Router-A(config-vlan-3)# ip-subnet 1.1.1.0/24
FESX424 Router-A(config-vlan-ip-subnet)# static e 9 to 12
FESX424 Router-A(config-vlan-ip-subnet)# router-interface ve3
FESX424 Router-A(config-vlan-ip-subnet)# ipx-network 1 ethernet_802.3
FESX424 Router-A(config-vlan-ipx-network)# static e 13 to 16
FESX424 Router-A(config-vlan-ipx-network)# router-interface ve4
FESX424 Router-A(config-vlan-ipx-network)# other-proto name block-other-protocols
FESX424 Router-A(config-vlan-other-proto)# exclude e 9 to 16
FESX424 Router-A(config-vlan-other-proto)# no dynamic
FESX424 Router-A(config-vlan-other-proto)# interface ve 3
FESX424 Router-A(config-vif-3)# ip addr 1.1.1.1/24
FESX424 Router-A(config-vif-3)# ip ospf area 0.0.0.0
FESX424 Router-A(config-vif-3)# int ve4
FESX424 Router-A(config-vif-4)# ipx network 1 ethernet_802.3
FESX424 Router-A(config-vif-4)#
```

Now configure VLAN 4. Remember this is a flat segment that, in the previous example, obtained its IP default gateway and IPX router services from an external FESX. In this example, FESX-A will provide the routing services for VLAN 4. You also want to configure the STP priority for VLAN 4 to make FESX-A the root bridge for this VLAN.

```
FESX424 Router-A(config-vif-4)# vlan 4 name Bridged_ALL_Protocols
FESX424 Router-A(config-vlan-4)# untag ethernet 17 to 24
FESX424 Router-A(config-vlan-4)# tag ethernet 25 to 26
FESX424 Router-A(config-vlan-4)# spanning-tree
FESX424 Router-A(config-vlan-4)# spanning-tree priority 500
FESX424 Router-A(config-vlan-4)# router-interface ve5
FESX424 Router-A(config-vlan-4)# int ve5
FESX424 Router-A(config-vif-5)# ip address 1.1.3.1/24
FESX424 Router-A(config-vif-5)# ip ospf area 0.0.0.0
```

```
FESX424 Router-A(config-vif-5)# ipx network 3 ethernet_802.3
FESX424 Router-A(config-vif-5)#
```

It is time to configure a separate port-based VLAN for each of the routed backbone ports (Ethernet 25 and 26). If you do not create a separate tagged port-based VLAN for each point-to-point backbone link, you need to include tagged interfaces for Ethernet 25 and 26 within VLANs 2, 3, and 8. This type of configuration makes the entire backbone a single STP domain for each VLAN 2, 3, and 8. This is the configuration used in the example in "Configuring IP Sub-net, IPX Network and Protocol-Based VLANs" on page 11-21. In this scenario, the virtual routing interfaces within port-based VLANs 2, 3, and 8 will be accessible using only one path through the network. The path that is blocked by STP is not available to the routing protocols until it is in the STP FORWARDING state.

```
FESX424 Router-A(config-vif-5)# vlan 5 name Rtr_BB_to_Bldg.2
FESX424 Router-A(config-vlan-5)# tag e 25
FESX424 Router-A(config-vlan-5)# no spanning-tree
FESX424 Router-A(config-vlan-5)# router-interface ve6
FESX424 Router-A(config-vlan-5)# vlan 6 name Rtr_BB_to_Bldg.3
FESX424 Router-A(config-vlan-6)# tag ethernet 26
FESX424 Router-A(config-vlan-6)# no spanning-tree
FESX424 Router-A(config-vlan-6)# router-interface ve7
FESX424 Router-A(config-vlan-6)# int ve6
FESX424 Router-A(config-vif-6)# ip addr 1.1.4.1/24
FESX424 Router-A(config-vif-6)# ip ospf area 0.0.0.0
FESX424 Router-A(config-vif-6)# ipx network 4 ethernet_802.3
FESX424 Router-A(config-vif-6)# int ve7
FESX424 Router-A(config-vif-7)# ip addr 1.1.5.1/24
FESX424 Router-A(config-vif-7)# ip ospf area 0.0.0.0
FESX424 Router-A(config-vif-7)# ipx network 5 ethernet_802.3
FESX424 Router-A(config-vif-7)#
```

This completes the configuration for FESX-A. The configuration for FESX-B and C is very similar except for a few issues.

- IP sub-nets and IPX networks configured on FESX-B and FESX-C must be unique across the entire network, except for the backbone port-based VLANs 5, 6, and 7 where the sub-net is the same but the IP address must change.
- There is no need to change the default priority of STP within VLAN 4.
- There is no need to include a virtual router interface within VLAN 4.
- The backbone VLAN between FESX-B and FESX-C must be the same at both ends and requires a new VLAN ID. The VLAN ID for this port-based VLAN is VLAN 7.

Configuration for FESX-B

Enter the following commands to configure FESX-B.

```
FESX424 Router> en
No password has been assigned yet...
FESX424 Router# config t
FESX424 Router(config)# hostname FESX-B
FESX424 Router-B(config)# router ospf
FESX424 Router-B(config-ospf-router)# area 0.0.0.0 normal
FESX424 Router-B(config-ospf-router)# router ipx
FESX424 Router-B(config-ospf-router)# vlan 2 name IP-Subnet_1.1.6.0/24
FESX424 Router-B(config-vlan-2)# untag e 1 to 4
FESX424 Router-B(config-vlan-2)# no spanning-tree
FESX424 Router-B(config-vlan-2)# router-interface ve1
FESX424 Router-B(config-vlan-2)# other-proto name block-other-protocols
FESX424 Router-B(config-vlan-other-proto)# no dynamic
FESX424 Router-B(config-vlan-other-proto)# exclude e 1 to 4
FESX424 Router-B(config-vlan-other-proto)# int ve1
FESX424 Router-B(config-vif-1)# ip addr 1.1.6.1/24
```

```
FESX424 Router-B(config-vif-1)# ip ospf area 0.0.0.0
FESX424 Router-B(config-vif-1)# vlan 8 name IPX_Network6
FESX424 Router-B(config-vlan-8)# untag e 5 to 8
FESX424 Router-B(config-vlan-8)# no span
FESX424 Router-B(config-vlan-8)# router-int ve2
FESX424 Router-B(config-vlan-8)# other-proto name block-other-protocols
FESX424 Router-B(config-vlan-other-proto)# no dynamic
FESX424 Router-B(config-vlan-other-proto)# exclude e 5 to 8
FESX424 Router-B(config-vlan-other-proto)# int ve2
FESX424 Router-B(config-vif-2)# ipx net 6 ethernet_802.3
FESX424 Router-B(config-vif-2)# vlan 3 name IP_Sub_&_IPX_Net_VLAN
FESX424 Router-B(config-vlan-3)# untag e 9 to 16
FESX424 Router-B(config-vlan-3)# no spanning-tree
FESX424 Router-B(config-vlan-3)# ip-subnet 1.1.7.0/24
FESX424 Router-B(config-vlan-ip-subnet)# static e 9 to 12
FESX424 Router-B(config-vlan-ip-subnet)# router-interface ve3
FESX424 Router-B(config-vlan-ip-subnet)# ipx-network 7 ethernet_802.3
FESX424 Router-B(config-vlan-ipx-network)# static e 13 to 16
FESX424 Router-B(config-vlan-ipx-network)# router-interface ve4
FESX424 Router-B(config-vlan-ipx-network)# other-proto name block-other-protocols
FESX424 Router-B(config-vlan-other-proto)# exclude e 9 to 16
FESX424 Router-B(config-vlan-other-proto)# no dynamic
FESX424 Router-B(config-vlan-other-proto)# interface ve 3
FESX424 Router-B(config-vif-3)# ip addr 1.1.7.1/24
FESX424 Router-B(config-vif-3)# ip ospf area 0.0.0.0
FESX424 Router-B(config-vif-3)# int ve4
FESX424 Router-B(config-vif-4)# ipx network 7 ethernet_802.3
FESX424 Router-B(config-vif-4)# vlan 4 name Bridged_ALL_Protocols
FESX424 Router-B(config-vlan-4)# untag ethernet 17 to 24
FESX424 Router-B(config-vlan-4)# tag ethernet 25 to 26
FESX424 Router-B(config-vlan-4)# spanning-tree
FESX424 Router-B(config-vlan-4)# vlan 5 name Rtr_BB_to_Bldg.1
FESX424 Router-B(config-vlan-5)# tag e 25
FESX424 Router-B(config-vlan-5)# no spanning-tree
FESX424 Router-B(config-vlan-5)# router-interface ve5
FESX424 Router-B(config-vlan-5)# vlan 7 name Rtr_BB_to_Bldg.3
FESX424 Router-B(config-vlan-7)# tag ethernet 26
FESX424 Router-B(config-vlan-7)# no spanning-tree
FESX424 Router-B(config-vlan-7)# router-interface ve6
FESX424 Router-B(config-vlan-7)# int ve5
FESX424 Router-B(config-vif-5)# ip addr 1.1.4.2/24
FESX424 Router-B(config-vif-5)# ip ospf area 0.0.0.0
FESX424 Router-B(config-vif-5)# ipx network 4 ethernet_802.3
FESX424 Router-B(config-vif-5)# int ve6
FESX424 Router-B(config-vif-6)# ip addr 1.1.8.1/24
FESX424 Router-B(config-vif-6)# ip ospf area 0.0.0.0
FESX424 Router-B(config-vif-6)# ipx network 8 ethernet_802.3
FESX424 Router-B(config-vif-6)#
```

Configuration for FESX-C

Enter the following commands to configure FESX-C.

```
FESX424 Router> en
```
No password has been assigned yet...
```
FESX424 Router# config t
FESX424 Router(config)# hostname FESX-C
FESX424 Router-C(config)# router ospf
FESX424 Router-C(config-ospf-router)# area 0.0.0.0 normal
FESX424 Router-C(config-ospf-router)# router ipx
```

```
FESX424 Router-C(config-ospf-router)# vlan 2 name IP-Subnet_1.1.9.0/24
FESX424 Router-C(config-vlan-2)# untag e 1 to 4
FESX424 Router-C(config-vlan-2)# no spanning-tree
FESX424 Router-C(config-vlan-2)# router-interface ve1
FESX424 Router-C(config-vlan-2)# other-proto name block-other-protocols
FESX424 Router-C(config-vlan-other-proto)# no dynamic
FESX424 Router-C(config-vlan-other-proto)# exclude e 1 to 4
FESX424 Router-C(config-vlan-other-proto)# int ve1
FESX424 Router-C(config-vif-1)# ip addr 1.1.9.1/24
FESX424 Router-C(config-vif-1)# ip ospf area 0.0.0.0
FESX424 Router-C(config-vif-1)# vlan 8 name IPX_Network9
FESX424 Router-C(config-vlan-8)# untag e 5 to 8
FESX424 Router-C(config-vlan-8)# no span
FESX424 Router-C(config-vlan-8)# router-int ve2
FESX424 Router-C(config-vlan-8)# other-proto name block-other-protocols
FESX424 Router-C(config-vlan-other-proto)# no dynamic
FESX424 Router-C(config-vlan-other-proto)# exclude e 5 to 8
FESX424 Router-C(config-vlan-other-proto)# int ve2
FESX424 Router-C(config-vif-2)# ipx net 9 ethernet_802.3
FESX424 Router-C(config-vif-2)# vlan 3 name IP_Sub_&_IPX_Net_VLAN
FESX424 Router-C(config-vlan-3)# untag e 9 to 16
FESX424 Router-C(config-vlan-3)# no spanning-tree
FESX424 Router-C(config-vlan-3)# ip-subnet 1.1.10.0/24
FESX424 Router-C(config-vlan-ip-subnet)# static e 9 to 12
FESX424 Router-C(config-vlan-ip-subnet)# router-interface ve3
FESX424 Router-C(config-vlan-ip-subnet)# ipx-network 10 ethernet_802.3
FESX424 Router-C(config-vlan-ipx-network)# static e 13 to 16
FESX424 Router-C(config-vlan-ipx-network)# router-interface ve4
FESX424 Router-C(config-vlan-ipx-network)# other-proto name block-other-protocols
FESX424 Router-C(config-vlan-other-proto)# exclude e 9 to 16
FESX424 Router-C(config-vlan-other-proto)# no dynamic
FESX424 Router-C(config-vlan-other-proto)# interface ve 3
FESX424 Router-C(config-vif-3)# ip addr 1.1.10.1/24
FESX424 Router-C(config-vif-3)# ip ospf area 0.0.0.0
FESX424 Router-C(config-vif-3)# int ve4
FESX424 Router-C(config-vif-4)# ipx network 10 ethernet_802.3
FESX424 Router-C(config-vif-4)# vlan 4 name Bridged_ALL_Protocols
FESX424 Router-C(config-vlan-4)# untag ethernet 17 to 24
FESX424 Router-C(config-vlan-4)# tag ethernet 25 to 26
FESX424 Router-C(config-vlan-4)# spanning-tree
FESX424 Router-C(config-vlan-4)# vlan 7 name Rtr_BB_to_Bldg.2
FESX424 Router-C(config-vlan-7)# tag e 25
FESX424 Router-C(config-vlan-7)# no spanning-tree
FESX424 Router-C(config-vlan-7)# router-interface ve5
FESX424 Router-C(config-vlan-7)# vlan 6 name Rtr_BB_to_Bldg.1
FESX424 Router-C(config-vlan-6)# tag ethernet 26
FESX424 Router-C(config-vlan-6)# no spanning-tree
FESX424 Router-C(config-vlan-6)# router-interface ve6
FESX424 Router-C(config-vlan-6)# int ve5
FESX424 Router-C(config-vif-5)# ip addr 1.1.8.2/24
FESX424 Router-C(config-vif-5)# ip ospf area 0.0.0.0
FESX424 Router-C(config-vif-5)# ipx network 8 ethernet_802.3
FESX424 Router-C(config-vif-5)# int ve6
FESX424 Router-C(config-vif-6)# ip addr 1.1.5.2/24
FESX424 Router-C(config-vif-6)# ip ospf area 0.0.0.0
```

```
FESX424 Router-C(config-vif-6)# ipx network 5 ethernet_802.3
FESX424 Router-C(config-vif-6)#
```

Configuring Protocol VLANs With Dynamic Ports

The configuration examples for protocol VLANs in the sections above show how to configure the VLANs using static ports. You also can configure the following types of protocol VLANs with dynamic ports:

- AppleTalk protocol
- IP protocol
- IPX protocol
- IP sub-net
- IPX network

NOTE: The software does not support dynamically adding ports to AppleTalk cable VLANs. Conceptually, an AppleTalk cable VLAN consists of a single network cable, connected to a single port. Therefore, dynamic addition and removal of ports is not applicable.

NOTE: You cannot route to or from protocol VLANs with dynamically added ports.

Aging of Dynamic Ports

When you add the ports to the VLAN, the software automatically adds them all to the VLAN. However, dynamically added ports age out. If the age time for a dynamic port expires, the software removes the port from the VLAN. If that port receives traffic for the IP sub-net or IPX network, the software adds the port to the VLAN again and starts the aging timer over. Each time the port receives traffic for the VLAN's IP sub-net or IPX network, the aging timer starts over.

Dynamic ports within any protocol VLAN age out after 10 minutes, if no member protocol traffic is received on a port within the VLAN. The aged out port, however, remains as a candidate dynamic port for that VLAN. The port becomes active in the VLAN again if member protocol traffic is received on that port.

Once a port is re-activated, the aging out period for the port is reset to 20 minutes. Each time a member protocol packet is received by a candidate dynamic port (aged out port) the port becomes active again and the aging out period is reset for 20 minutes.

Configuration Guidelines

- You cannot dynamically add a port to a protocol VLAN if the port has any routing configuration parameters. For example, the port cannot have a virtual routing interface, IP sub-net address, IPX network address, or AppleTalk network address configured on it.
- Once you dynamically add a port to a protocol VLAN, you cannot configure routing parameters on the port.
- Dynamic VLAN ports are not required or supported on AppleTalk cable VLANs.

Configuring an IP, IPX, or AppleTalk Protocol VLAN with Dynamic Ports

To configure an IP, IPX, or AppleTalk protocol VLAN with dynamic ports, use the following method.

To configure port-based VLAN 10, then configure an IP protocol VLAN within the port-based VLAN with dynamic ports, enter the following commands such as the following:

```
FastIron SuperX Router(config)# vlan 10 by port
FastIron SuperX Router(config-vlan-10)# untag ethernet 1/1 to 1/6
added untagged port ethe 1/1 to 1/6 to port-vlan 30.
FastIron SuperX Router(config-vlan-10)# ip-proto name IP_Prot_VLAN
FastIron SuperX Router(config-vlan-10)# dynamic
```

```
FastIron SuperX Router(config)# write memory
```

Syntax: vlan <vlan-id> by port [name <string>]

Syntax: untagged ethernet [<slotnum>/]<portnum> to [<slotnum>/]<portnum>

Or

Syntax: untagged ethernet [<slotnum>/]<portnum> ethernet [<slotnum>/]<portnum>

NOTE: Use the first **untagged** command for adding a range of ports. Use the second command for adding separate ports (not in a range).

Syntax: ip-proto [name <string>]

Syntax: ipx-proto [name <string>]

Syntax: appletalk-cable-vlan <num> [name <string>]

Syntax: dynamic

The procedure is similar for IPX and AppleTalk protocol VLANs. Enter **ipx-proto** or **atalk-proto** instead of **ip-proto**.

Configuring an IP Sub-Net VLAN with Dynamic Ports

To configure port-based VLAN 10, then configure an IP sub-net VLAN within the port-based VLAN with dynamic ports, enter commands such as the following:

```
FastIron SuperX Router(config)# vlan 10 by port name IP_VLAN
FastIron SuperX Router(config-vlan-10)# untag ethernet 1/1 to 1/6
added untagged port ethe 1/1 to 1/6 to port-vlan 10.
FastIron SuperX Router(config-vlan-10)# ip-subnet 1.1.1.0/24 name Mktg-LAN
FastIron SuperX Router(config-vlan-10)# dynamic
FastIron SuperX Router(config)# write memory
```

These commands create a port-based VLAN on chassis ports 1/1 – 1/6 named “Mktg-LAN”, configure an IP sub-net VLAN within the port-based VLAN, and then add ports from the port-based VLAN dynamically.

Syntax: vlan <vlan-id> by port [name <string>]

Syntax: untagged ethernet [<slotnum>/]<portnum> to [<slotnum>/]<portnum>

Or

Syntax: untagged ethernet [<slotnum>/]<portnum> ethernet [<slotnum>/]<portnum>

NOTE: Use the first **untagged** command for adding a range of ports. Use the second command for adding separate ports (not in a range).

Syntax: ip-subnet <ip-addr> <ip-mask> [name <string>]

Or

Syntax: ip-subnet <ip-addr>/<mask-bits> [name <string>]

Syntax: dynamic

Configuring an IPX Network VLAN with Dynamic Ports

To configure port-based VLAN 20, then configure an IPX network VLAN within the port-based VLAN with dynamic ports, enter commands such as the following:

```
FastIron SuperX Router(config)# vlan 20 by port name IPX_VLAN
FastIron SuperX Router(config-vlan-10)# untag ethernet 2/1 to 2/6
added untagged port ethe 2/1 to 2/6 to port-vlan 20.
FastIron SuperX Router(config-vlan-10)# ipx-network abcd ethernet_ii name Eng-LAN
```



```
FastIron SuperX Router(config-vlan-10)# dynamic
FastIron SuperX Router(config)# write memory
```

These commands create a port-based VLAN on chassis ports 2/1 – 2/6 named “Eng-LAN”, configure an IPX network VLAN within the port-based VLAN, and then add ports from the port-based VLAN dynamically.

Syntax: vlan <vlan-id> by port [name <string>]

Syntax: untagged ethernet [<slotnum>/]<portnum> to [<slotnum>/]<portnum>

Or

Syntax: untagged ethernet [<slotnum>/]<portnum> ethernet [<slotnum>/]<portnum>

NOTE: Use the first **untagged** command for adding a range of ports. Use the second command for adding separate ports (not in a range).

Syntax: ipx-network <network-addr> ethernet_ii | ethernet_802.2 | ethernet_802.3 | ethernet_snap [name <string>]

Syntax: dynamic

Configuring Uplink Ports Within a Port-Based VLAN

You can configure a subset of the ports in a port-based VLAN as uplink ports. When you configure uplink ports in a port-based VLAN, the device sends all broadcast and unknown-unicast traffic from a port in the VLAN to the uplink ports, but not to other ports within the VLAN. Thus, the uplink ports provide tighter broadcast control within the VLAN.

For example, if two ports within a port-based VLAN are Gigabit ports attached to the network and the other ports in the VLAN are 10/100 ports attached to clients, you can configure the two ports attached to the network as uplink ports. In this configuration, broadcast and unknown-unicast traffic in the VLAN does not go to all ports in the VLAN. The traffic goes only to the uplink ports. The clients on the network do not receive broadcast and unknown-unicast traffic from other ports, including other clients.

To configure a port-based VLAN containing uplink ports, enter commands such as the following:

```
FastIron SuperX Router(config)# vlan 10 by port
FastIron SuperX Router(config-vlan-10)# untag ethernet 1/1 to 1/24
FastIron SuperX Router(config-vlan-10)# untag ethernet 2/1 to 2/2
FastIron SuperX Router(config-vlan-10)# uplink-switch ethernet 2/1 to 2/2
```

Syntax: [no] uplink-switch ethernet [<slotnum>/]<portnum> [to [<slotnum>/]<portnum> | ethernet [<slotnum>/]<portnum>]

In this example, 24 ports on a 10/100 module and two Gigabit ports on a Gigabit module are added to port-based VLAN 10. The two Gigabit ports are then configured as uplink ports.

Configuring the Same IP Sub-Net Address on Multiple Port-Based VLANs

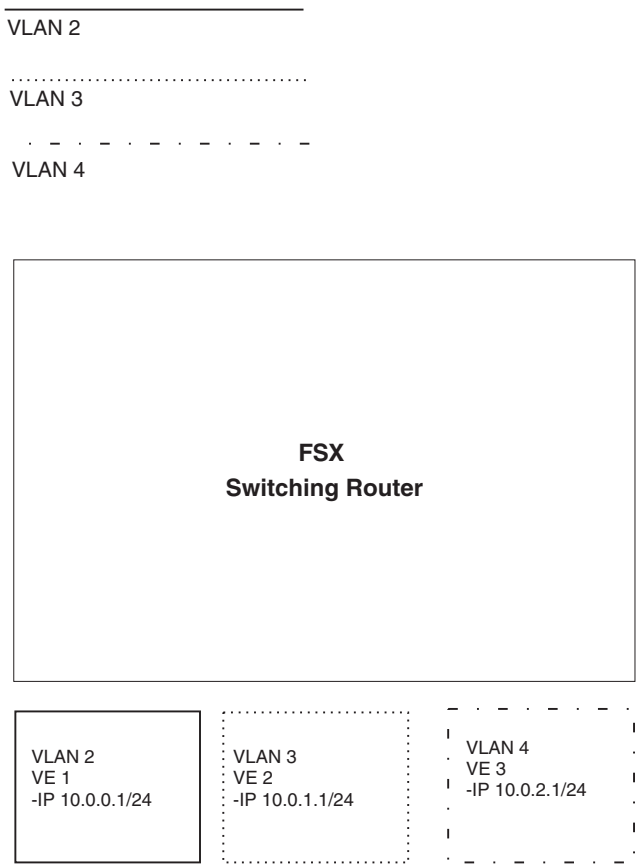
For a Foundry device to route between port-based VLANs, you must add a virtual routing interface to each VLAN. Generally, you also configure a unique IP sub-net address on each virtual routing interface. For example, if you have three port-based VLANs, you add a virtual routing interface to each VLAN, then add a separate IP sub-net address to each virtual routing interface. The IP address on each of the virtual routing interfaces must be in a separate sub-net. The Foundry device routes Layer 3 traffic between the sub-nets using the sub-net addresses.

NOTE: This feature applies only to Layer 3 Switches.

NOTE: Before using the method described in this section, see “Configuring VLAN Groups and Virtual Routing Interface Groups” on page 11-40. You might be able to achieve the results you want using the methods in that section instead.

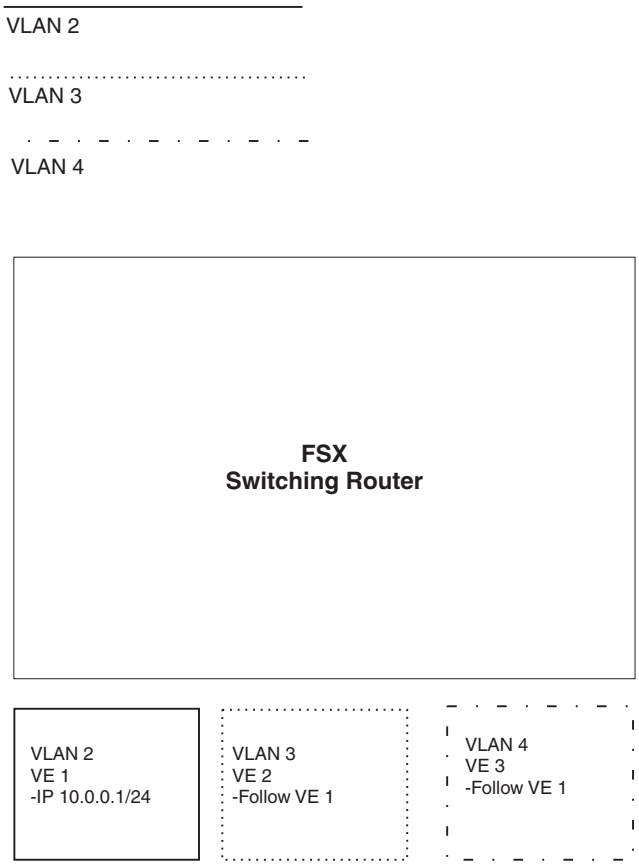
Figure 11.14 shows an example of this type of configuration.

Figure 11.14 Multiple port-based VLANs with separate protocol addresses



As shown in this example, each VLAN has a separate IP sub-net address. If you need to conserve IP sub-net addresses, you can configure multiple VLANs with the same IP sub-net address, as shown in Figure 11.15.

Figure 11.15 Multiple port-based VLANs with the same protocol address



Each VLAN still requires a separate virtual routing interface. However, all three VLANs now use the same IP sub-net address.

In addition to conserving IP sub-net addresses, this feature allows containment of Layer 2 broadcasts to segments within an IP sub-net. For ISP environments where the same IP sub-net is allocated to different customers, placing each customer in a separate VLAN allows all customers to share the IP sub-net address, while at the same time isolating them from one another's Layer 2 broadcasts.

NOTE: You can provide redundancy to an IP sub-net address that contains multiple VLANs using a pair of Foundry Layer 3 Switches configured for Foundry's VRRP (Virtual Router Redundancy Protocol).

The Foundry device performs proxy Address Resolution Protocol (ARP) for hosts that want to send IP traffic to hosts in other VLANs that are sharing the same IP sub-net address. If the source and destination hosts are in the same VLAN, the Foundry device does not need to use ARP.

- If a host attached to one VLAN sends an ARP message for the MAC address of a host in one of the other VLANs using the same IP sub-net address, the Foundry device performs a proxy ARP on behalf of the other host. The Foundry device then replies to the ARP by sending the virtual routing interface MAC address. The Foundry device uses the same MAC address for all virtual routing interfaces.

  When the host that sent the ARP then sends a unicast packet addressed to the virtual routing interface's MAC address, the device switches the packet on Layer 3 to the destination host on the VLAN.

NOTE: If the Foundry device's ARP table does not contain the requested host, the Foundry device forwards the ARP request on Layer 2 to the same VLAN as the one that received the ARP request. Then the device sends an ARP for the destination to the other VLANs that are using the same IP sub-net address.

- If the destination is in the same VLAN as the source, the Foundry device does not need to perform a proxy ARP.

To configure multiple VLANs to use the same IP sub-net address:

- Configure each VLAN, including adding tagged or untagged ports.
- Configure a separate virtual routing interface for each VLAN, but do not add an IP sub-net address to more than one of the virtual routing interfaces.
- Configure the virtual routing interfaces that do not have the IP sub-net address to "follow" the virtual routing interface that does have the address.

To configure the VLANs shown in Figure 11.15, you could enter the following commands.

```
FastIron SuperX Router(config)# vlan 1 by port
FastIron SuperX Router(config-vlan-1)# untag ethernet 1/1
FastIron SuperX Router(config-vlan-1)# tag ethernet 1/8
FastIron SuperX Router(config-vlan-1)# router-interface ve 1
```

Syntax: ip follow ve <num>

The commands above configure port-based VLAN 1. The VLAN has one untagged port (1/1) and a tagged port (1/8). In this example, all three VLANs contain port 1/8 so the port must be tagged to allow the port to be in multiple VLANs. You can configure VLANs to share a Layer 3 protocol interface regardless of tagging. A combination of tagged and untagged ports is shown in this example to demonstrate that sharing the interface does not change other VLAN features.

Notice that each VLAN still requires a unique virtual routing interface.

The following commands configure port-based VLANs 2 and 3.

```
FastIron SuperX Router(config-vlan-1)# vlan 2 by port
FastIron SuperX Router(config-vlan-2)# untag ethernet 1/2
FastIron SuperX Router(config-vlan-2)# tag ethernet 1/8
FastIron SuperX Router(config-vlan-2)# router-interface ve 2
FastIron SuperX Router(config-vlan-2)# vlan 3 by port
FastIron SuperX Router(config-vlan-3)# untag ethernet 1/5 to 1/6
FastIron SuperX Router(config-vlan-3)# tag ethernet 1/8
FastIron SuperX Router(config-vlan-3)# router-interface ve 3
```

The following commands configure an IP sub-net address on virtual routing interface 1.

```
FastIron SuperX Router(config-vlan-3)# interface ve 1
FastIron SuperX Router(config-vif-1)# ip address 10.0.0.1/24
```

The following commands configure virtual routing interfaces 2 and 3 to "follow" the IP sub-net address configured on virtual routing interface 1.

```
FastIron SuperX Router(config-vif-1)# interface ve 2
FastIron SuperX Router(config-vif-2)# ip follow ve 1
FastIron SuperX Router(config-vif-2)# interface ve 3
FastIron SuperX Router(config-vif-3)# ip follow ve 1
```

NOTE: Since virtual routing interfaces 2 and 3 do not have their own IP sub-net addresses but instead are "following" virtual routing interface a's IP address, you still can configure an IPX or AppleTalk interface on virtual routing interfaces 2 and 3.

Using Separate ACLs on IP Follower Virtual Routing Interfaces

NOTE: This section applies to flow-based ACLs only.

The IP follower feature allows multiple virtual routing interfaces to share the same IP address. One virtual routing interface has the IP address and the other virtual routing interfaces are configured to follow the virtual routing interface that has the address.

By default, the follower interfaces are secured by the ACLs that are applied to the interface that has the address. In fact, an ACL applied to a follower interface is ignored. For example, if you configure virtual routing interfaces 1, 2, and 3, and configure interfaces 2 and 3 to follow interface 1, then the ACLs applied to interface 1 also apply to interfaces 2 and 3. Any ACLs applied separately to interface 2 or 3 are ignored.

You can enable a follower virtual routing interface to use the ACLs you apply to it instead of using the ACLs applied to the interface that has the address. For example, you can enable virtual routing interface 2 to use its own ACLs instead of using interface 1's ACLs.

To enable a virtual routing interface to use its own ACLs instead of the ACLs of the interface it is following, enter the following command at the configuration level for the interface:

```
FastIron SuperX Router(config-vif-2)# no ip follow acl
```

Syntax: [no] ip follow acl

The following commands show a complete IP follower configuration. Virtual routing interfaces 2 and 3 have been configured to share the IP address of virtual routing interface 1, but also have been configured to use their own ACLs instead of virtual routing interface 1's ACLs.

```
FastIron SuperX Router(config)# vlan 1 name primary_vlan
FastIron SuperX Router(config-vlan-1)# untag ethernet 1/1
FastIron SuperX Router(config-vlan-1)# tag ethernet 1/8
FastIron SuperX Router(config-vlan-1)# router-interface ve 1
FastIron SuperX Router(config-vlan-1)# exit
FastIron SuperX Router(config)# interface ve 1
FastIron SuperX Router(config-ve-1)# ip address 10.0.0.1/24
FastIron SuperX Router(config-ve-1)# ip access-group 1 in
FastIron SuperX Router(config-ve-1)# exit

FastIron SuperX Router(config)# vlan 2 name followerA
FastIron SuperX Router(config-vlan-2)# untag ethernet 1/2
FastIron SuperX Router(config-vlan-2)# tag ethernet 1/8
FastIron SuperX Router(config-vlan-2)# router-interface ve 2
FastIron SuperX Router(config-vlan-2)# exit
FastIron SuperX Router(config)# interface ve 2
FastIron SuperX Router(config-ve-2)# ip follow ve 1
FastIron SuperX Router(config-v2-2)# no ip follow acl
FastIron SuperX Router(config-ve-2)# ip access-group 2 in
FastIron SuperX Router(config-ve-2)# exit

FastIron SuperX Router(config)# vlan 3 name followerB
FastIron SuperX Router(config-vlan-3)# untag ethernet 1/5 to 1/6
FastIron SuperX Router(config-vlan-3)# tag ethernet 1/8
FastIron SuperX Router(config-vlan-3)# router-interface ve 3
FastIron SuperX Router(config-vlan-3)# exit
FastIron SuperX Router(config)# interface ve 3
FastIron SuperX Router(config-ve-3)# ip follow ve 1
FastIron SuperX Router(config-ve-3)# no ip follow acl
FastIron SuperX Router(config-ve-3)# ip access-group 3 out
FastIron SuperX Router(config-ve-3)# exit
```

Configuring VLAN Groups and Virtual Routing Interface Groups

To simplify configuration when you have many VLANs with the same configuration, you can configure VLAN groups and virtual routing interface groups.

NOTE: VLAN groups are supported on Layer 3 Switches and Layer 2 Switches. Virtual routing interface groups are supported only on Layer 3 Switches.

When you create a VLAN group, the VLAN parameters you configure for the group apply to all the VLANs within the group. Additionally, you can easily associate the same IP sub-net interface with all the VLANs in a group by configuring a virtual routing interface group with the same ID as the VLAN group.

- The VLAN group feature allows you to create multiple port-based VLANs with identical port members. Since the member ports are shared by all the VLANs within the group, you must add the ports as tagged ports. This feature not only simplifies VLAN configuration but also allows you to have a large number of identically configured VLANs in a startup-config file on the device's flash memory module. Normally, a startup-config file with a large number of VLANs might not fit on the flash memory module. By grouping the identically configured VLANs, you can conserve space in the startup-config file so that it fits on the flash memory module.
- The virtual routing interface group feature is useful when you want to configure the same IP sub-net address on all the port-based VLANs within a VLAN group. You can configure a virtual routing interface group only after you configure a VLAN group with the same ID. The virtual routing interface group automatically applies to the VLANs in the VLAN group that has the same ID and cannot be applied to other VLAN groups or to individual VLANs.

You can create up to 32 VLAN groups and 32 virtual routing interface groups. A virtual routing interface group always applies only to the VLANs in the VLAN group with the same ID.

NOTE: Depending on the size of the VLAN ID range you want to use for the VLAN group, you might need to allocate additional memory for VLANs. On Layer 3 Switches, if you allocate additional memory for VLANs, you also need to allocate the same amount of memory for virtual routing interfaces. This is true regardless of whether you use the virtual routing interface groups. To allocate additional memory, see "Allocating Memory for More VLANs or Virtual Routing Interfaces" on page 11-42.

Configuring a VLAN Group

To configure a VLAN group, enter commands such as the following:

```
FastIron SuperX Router(config)# vlan-group 1 vlan 2 to 1000
FastIron SuperX Router(config-vlan-group-1)# tagged 1/1 to 1/2
```

The first command in this example begins configuration for VLAN group 1, and assigns VLANs 2 through 1000 to the group. The second command adds ports 1/1 and 1/2 as tagged ports. Since all the VLANs in the group share the ports, you must add the ports as tagged ports.

Syntax: vlan-group <num> vlan <vlan-id> to <vlan-id>

Syntax: tagged ethernet [<slotnum>/]<portnum> [to [<slotnum>/]<portnum> | ethernet [<slotnum>/]<portnum>]

The <num> parameter with the **vlan-group** command specifies the VLAN group ID and can be from 1 – 32. The **vlan** <vlan-id> **to** <vlan-id> parameters specify a contiguous range (a range with no gaps) of individual VLAN IDs. Specify the low VLAN ID first and the high VLAN ID second. The command adds all the specified VLANs to the VLAN group.

NOTE: The device's memory must be configured to contain at least the number of VLANs you specify for the higher end of the range. For example, if you specify 2048 as the VLAN ID at the high end of the range, you first must increase the memory allocation for VLANs to 2048 or higher. Additionally, on Layer 3 Switches, if you allocate additional memory for VLANs, you also need to allocate the same amount of memory for virtual routing interfaces, before you configure the VLAN groups. This is true regardless of whether you use the virtual routing interface groups. The memory allocation is required because the VLAN groups and virtual routing interface groups have a one-to-one mapping. See "Allocating Memory for More VLANs or Virtual Routing Interfaces" on page 11-42.

If a VLAN within the range you specify is already configured, the CLI does not add the group but instead displays an error message. In this case, create the group by specifying a valid contiguous range. Then add more VLANs to the group after the CLI changes to the configuration level for the group. See the following example.

You can add and remove individual VLANs or VLAN ranges from at the VLAN group configuration level. For example, if you want to add VLANs 1001 and 1002 to VLAN group 1 and remove VLANs 900 through 1000, enter the following commands:

```
FastIron SuperX Router(config-vlan-group-1)# add-vlan 1001 to 1002
FastIron SuperX Router(config-vlan-group-1)# remove-vlan 900 to 1000
```

Syntax: add-vlan <vlan-id> [to <vlan-id>]

Syntax: remove-vlan <vlan-id> [to <vlan-id>]

Displaying Information about VLAN Groups

To display VLAN group configuration information, enter the following command:

Syntax: show vlan-group [<group-id>]

This example shows configuration information for two VLAN groups, group 1 and group 2.

The <group-id> specifies a VLAN group. If you do not use this parameter, the configuration information for all the configured VLAN groups is displayed.

Configuring a Virtual Routing Interface Group

A virtual routing interface group allows you to associate the same IP sub-net interface with multiple port-based VLANs. For example, if you associate a virtual routing interface group with a VLAN group, all the VLANs in the group have the IP interface of the virtual routing interface group.

NOTE: When you configure a virtual routing interface group, all members of the group have the same IP sub-net address. This feature is useful in collocation environments where the device has many IP addresses and you want to conserve the IP address space.

To configure a virtual routing interface group, enter commands such as the following:

```
FastIron SuperX Router(config)# vlan-group 1
FastIron SuperX Router(config-vlan-group-1)# group-router-interface
FastIron SuperX Router(config-vlan-group-1)# exit
FastIron SuperX Router(config)# interface group-ve 1
FastIron SuperX Router(config-vif-group-1)# ip address 10.10.10.1/24
```

These commands enable VLAN group 1 to have a group virtual routing interface, then configure virtual routing interface group 1. The software always associates a virtual routing interface group only with the VLAN group that has the same ID. In this example, the VLAN group ID is 1, so the corresponding virtual routing interface group also must have ID 1.

Syntax: group-router-interface

Syntax: interface group-ve <num>

Syntax: [no] ip address <ip-addr> <ip-mask> [secondary]

or

Syntax: [no] ip address <ip-addr>/<mask-bits> [secondary]

The **router-interface-group** command enables a VLAN group to use a virtual routing interface group. Enter this command at the configuration level for the VLAN group. This command configures the VLAN group to use the virtual routing interface group that has the same ID as the VLAN group. You can enter this command when you configure the VLAN group for the first time or later, after you have added tagged ports to the VLAN and so on.

The <num> parameter in the **interface group-ve** <num> command specifies the ID of the VLAN group with which you want to associate this virtual routing interface group. The VLAN group must already be configured and enabled to use a virtual routing interface group. The software automatically associates the virtual routing interface group with the VLAN group that has the same ID. You can associate a virtual routing interface group only with the VLAN group that has the same ID.

The syntax and usage for the **ip address** command is the same as when you use the command at the interface level to add an IP interface.

Displaying the VLAN Group and Virtual Routing Interface Group Information

To verify configuration of VLAN groups and virtual routing interface groups, display the running-config file. If you have saved the configuration to the startup-config file, you also can verify the configuration by displaying the startup-config file. The following example shows the running-config information for the VLAN group and virtual routing interface group configured in the previous examples. The information appears in the same way in the startup-config file.

```
FastIron SuperX Router(config)# show running-config
```

lines not related to the VLAN group omitted...

```
vlan-group 1 vlan 2 to 900
 add-vlan 1001 to 1002
 tagged ethe 1/1 to 1/2
 router-interface-group
```

lines not related to the virtual routing interface group omitted...

```
interface group-ve 1
 ip address 10.10.10.1 255.255.255.0
```

NOTE: If you have enabled display of sub-net masks in CIDR notation, the IP address information is shown as follows: 10.10.10.1/24.

Allocating Memory for More VLANs or Virtual Routing Interfaces

Layer 3 Switches can support up to 4095 VLANs and 4095 virtual routing interfaces.

The number of VLANs and virtual routing interfaces supported on your product depends on the device and, for Chassis devices, the amount of DRAM on the management module. Table 11.2 lists the default and configurable

maximum numbers of VLANs and virtual routing interfaces for Layer 3 Switches and Layer 2 Switches. Unless otherwise noted, the values apply to both types of switches.

Table 11.2: VLAN and Virtual Routing Interface Support

VLANs		Virtual Routing Interfaces	
Default Maximum	**Configurable Maximum**	**Default Maximum**	**Configurable Maximum**
64	4094	255	512

NOTE: If many of your VLANs will have an identical configuration, you might want to configure VLAN groups and virtual routing interface groups after you increase the system capacity for VLANs and virtual routing interfaces. See “Configuring VLAN Groups and Virtual Routing Interface Groups” on page 11-40.

Increasing the Number of VLANs You Can Configure

NOTE: Although you can specify up to 4095 VLANs, you can configure only 4094 VLANs. VLAN ID 4094 is reserved for use by the Single Spanning Tree feature.

To increase the maximum number of VLANs you can configure, enter commands such as the following at the global CONFIG level of the CLI:

```
FESX424 Router(config)# system-max vlan 2048
FESX424 Router(config)# write memory
FESX424 Router(config)# end
FESX424 Router# reload
```

Syntax: system-max vlan <num>

The <num> parameter indicates the maximum number of VLANs. The range of valid values depends on the device you are configuring. See Table 11.2.

Increasing the Number of Virtual Routing Interfaces You Can Configure

To increase the maximum number of virtual routing interfaces you can configure, enter commands such as the following at the global CONFIG level of the CLI:

```
FESX424 Router(config)# system-max virtual-interface 4095
FESX424 Router(config)# write memory
FESX424 Router(config)# end
FESX424 Router# reload
```

Syntax: system-max virtual-interface <num>

The <num> parameter indicates the maximum number of virtual routing interfaces. The range of valid values depends on the device you are configuring. See Table 11.2.

Configuring Super Aggregated VLANs

You can aggregate multiple VLANs within another VLAN. This feature allows you to construct Layer 2 paths and channels. This feature is particularly useful for Virtual Private Network (VPN) applications in which you need to provide a private, dedicated Ethernet connection for an individual client to transparently reach its sub-net across multiple networks.

Conceptually, the paths and channels are similar to Asynchronous Transfer Mode (ATM) paths and channels. A path contains multiple channels, each of which is a dedicated circuit between two end points. The two devices at

the end points of the channel appear to each other to be directly attached. The network that connects them is transparent to the two devices.

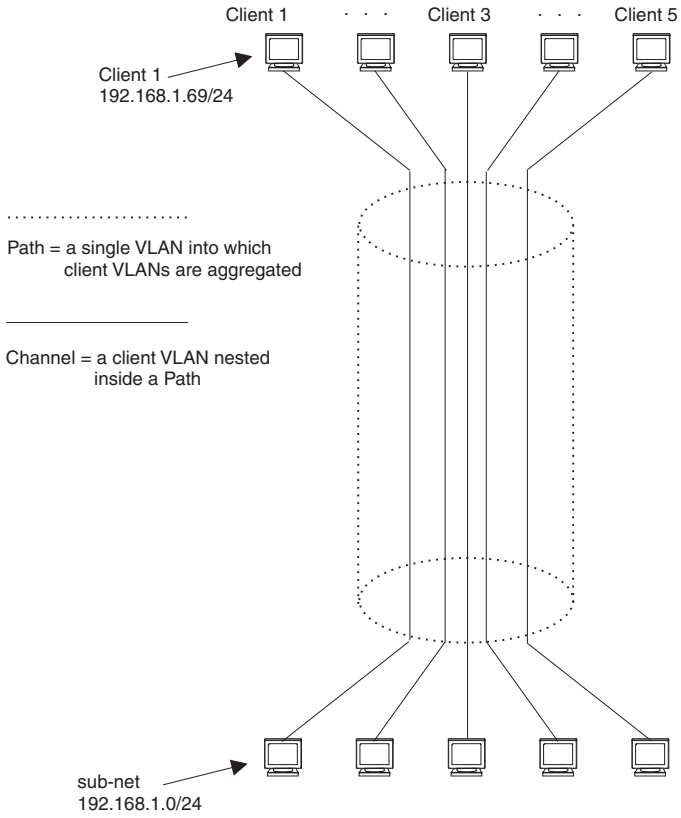
You can aggregate up to 4094 VLANs within another VLAN. This provides a total VLAN capacity on one Foundry device of 16,760,836 channels (4094 * 4094).

The devices connected through the channel are not visible to devices in other channels. Therefore, each client has a private link to the other side of the channel.

The feature allows point-to-point and point-to-multipoint connections.

Figure 11.16 shows a conceptual picture of the service that aggregated VLANs provide. Aggregated VLANs provide a path for multiple client channels. The channels do not receive traffic from other channels. Thus, each channel is a private link.

Figure 11.16 Conceptual Model of the Super Aggregated VLAN Application

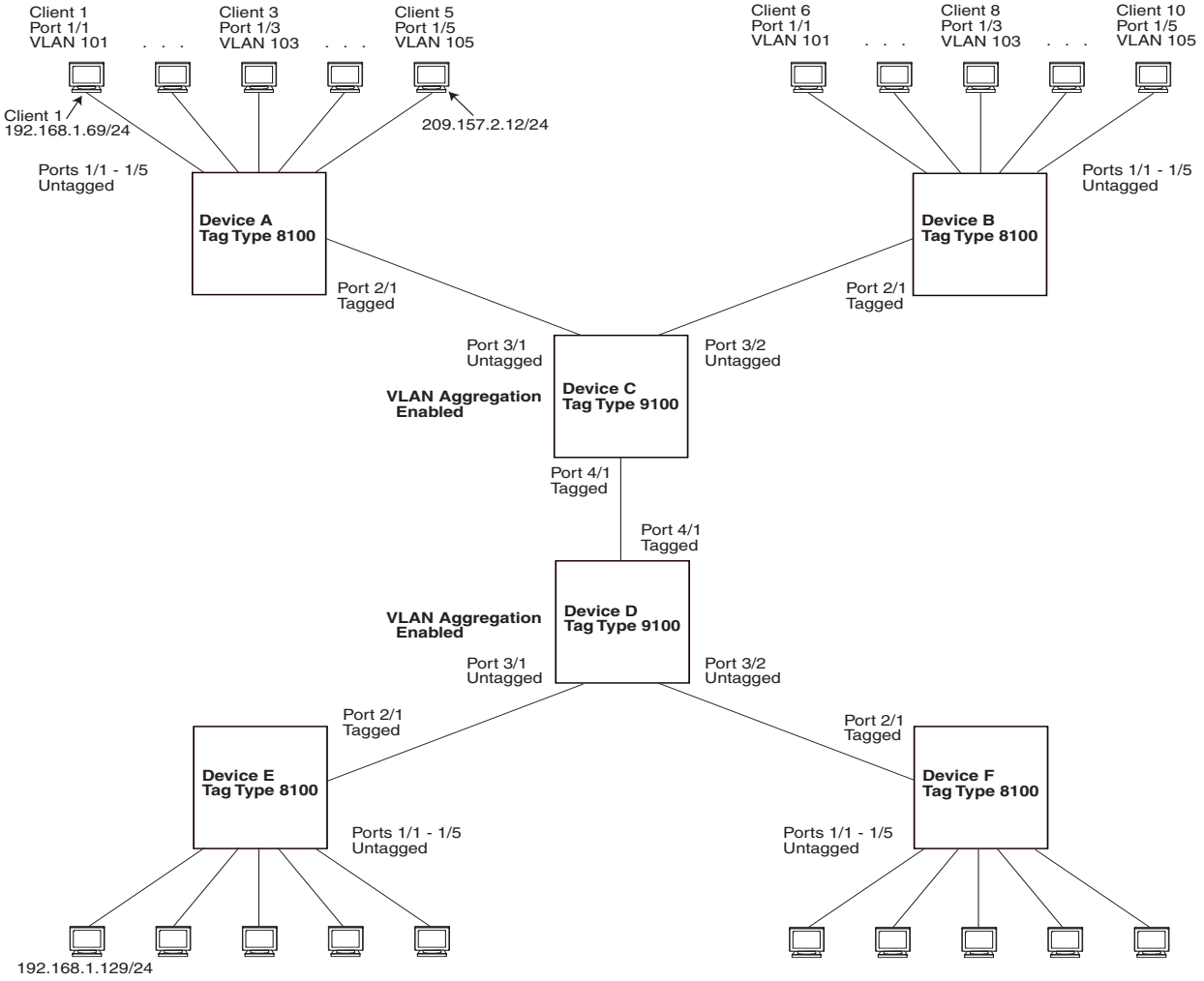


Each client connected to the edge device is in its own port-based VLAN, which is like an ATM channel. All the clients' VLANs are aggregated by the edge device into a single VLAN for connection to the core. The single VLAN that aggregates the clients' VLANs is like an ATM path.

The device that aggregates the VLANs forwards the aggregated VLAN traffic through the core. The core can consist of multiple devices that forward the aggregated VLAN traffic. The edge device at the other end of the core separates the aggregated VLANs into the individual client VLANs before forwarding the traffic. The edge devices forward the individual client traffic to the clients. For the clients' perspective, the channel is a direct point-to-point link.

Figure 11.17 shows an example application that uses aggregated VLANs. This configuration includes the client connections shown in Figure 11.16.

Figure 11.17 Example Super Aggregated VLAN Application



In this example, a collocation service provides private channels for multiple clients. Although the same devices are used for all the clients, the VLANs ensure that each client receives its own Layer 2 broadcast domain, separate from the broadcast domains of other clients. For example, client 1 cannot ping client 5.

The clients at each end of a channel appear to each other to be directly connected and thus can be on the same sub-net and use network services that require connection to the same sub-net. In this example, client 1 is in sub-net 192.168.1.0/24 and so is the device at the other end of client 1's channel.

Since each VLAN configured on the core devices is an aggregate of multiple client VLANs, the aggregated VLANs greatly increase the number of clients a core device can accommodate.

This example shows a single link between the core devices. However, you can use a trunk group to add link-level redundancy.

Configuring Aggregated VLANs

To configure aggregated VLANs, perform the following tasks:

- On each edge device, configure a separate port-based VLAN for each client connected to the edge device. In each client VLAN:
  - Add the port connected to the client as an untagged port.
  - Add the port connected to the core device (the device that will aggregate the VLANs) as a tagged port.

This port must be tagged because all the client VLANs share the port as an uplink to the core device.

- On each core device:
  - Enable VLAN aggregation. This support allows the core device to add an additional tag to each Ethernet frame that contains a VLAN packet from the edge device. The additional tag identifies the aggregate VLAN (the path). However, the additional tag can cause the frame to be longer than the maximum supported frame size. The larger frame support allows Ethernet frames up to 1530 bytes long.

NOTE: Enable the VLAN aggregation option only on the core devices.

  - Configure a VLAN tag type (tag ID) that is different than the tag type used on the edge devices. If you use the default tag type (8100) on the edge devices, set the tag type on the core devices to another value, such as 9100. The tag type must be the same on all the core devices. The edge devices also must have the same tag type but the type must be different from the tag type on the core devices.

NOTE: You can enable the Spanning Tree Protocol (STP) on the edge devices or the core devices, but not both. If you enable STP on the edge devices and the core devices, STP will prevent client traffic from travelling through the core to the other side.

Configuring Aggregated VLANs on an Edge Device

To configure the aggregated VLANs on device A in Figure 11.17 on page 11-45, enter the following commands:

```
FastIron SuperX Router(config)# vlan 101 by port
FastIron SuperX Router(config-vlan-101)# tagged ethernet 2/1
FastIron SuperX Router(config-vlan-101)# untagged ethernet 1/1
FastIron SuperX Router(config-vlan-101)# exit
FastIron SuperX Router(config)# vlan 102 by port
FastIron SuperX Router(config-vlan-102)# tagged ethernet 2/1
FastIron SuperX Router(config-vlan-102)# untagged ethernet 1/2
FastIron SuperX Router(config-vlan-102)# exit
FastIron SuperX Router(config)# vlan 103 by port
FastIron SuperX Router(config-vlan-103)# tagged ethernet 2/1
FastIron SuperX Router(config-vlan-103)# untagged ethernet 1/3
FastIron SuperX Router(config-vlan-103)# exit
FastIron SuperX Router(config)# vlan 104 by port
FastIron SuperX Router(config-vlan-104)# tagged ethernet 2/1
FastIron SuperX Router(config-vlan-104)# untagged ethernet 1/4
FastIron SuperX Router(config-vlan-104)# exit
FastIron SuperX Router(config)# vlan 105 by port
FastIron SuperX Router(config-vlan-105)# tagged ethernet 2/1
FastIron SuperX Router(config-vlan-105)# untagged ethernet 1/5
FastIron SuperX Router(config-vlan-105)# exit
FastIron SuperX Router(config)# write memory
```

Syntax: [no] vlan <vlan-id> [by port]

Syntax: [no] tagged ethernet [<slotnum>/]<portnum> [to [<slotnum>/]<portnum> | ethernet [<slotnum>/]<portnum>]

Syntax: [no] untagged ethernet [<slotnum>/]<portnum> [to [<slotnum>/]<portnum> | ethernet [<slotnum>/]<portnum>]

Use the **tagged** command to add the port that the device uses for the uplink to the core device. Use the **untagged** command to add the ports connected to the individual clients.

Configuring Aggregated VLANs on a Core Device

To configure the aggregated VLANs on device C in Figure 11.17 on page 11-45, enter the following commands:

```
FastIron SuperX Router(config)# tag-type 9100
FastIron SuperX Router(config)# aggregated-vlan
```



```
FastIron SuperX Router(config)# vlan 101 by port
FastIron SuperX Router(config-vlan-101)# tagged ethernet 4/1
FastIron SuperX Router(config-vlan-101)# untagged ethernet 3/1
FastIron SuperX Router(config-vlan-101)# exit
FastIron SuperX Router(config)# vlan 102 by port
FastIron SuperX Router(config-vlan-102)# tagged ethernet 4/1
FastIron SuperX Router(config-vlan-102)# untagged ethernet 3/2
FastIron SuperX Router(config-vlan-102)# exit
FastIron SuperX Router(config)# write memory
```

Syntax: [no] tag-type <num>

Syntax: [no] aggregated-vlan

The <num> parameter specifies the tag type can be a hexadecimal value from 0 – ffff. The default is 8100.

Verifying the Configuration

You can verify the VLAN, VLAN aggregation option, and tag configuration by viewing the running-config. To display the running-config, enter the **show running-config** command from any CLI prompt. After you save the configuration changes to the startup-config, you also can display the settings in that file by entering the **show configuration** command from any CLI prompt.

Complete CLI Examples

The following sections show all the Aggregated VLAN configuration commands on the devices in Figure 11.17 on page 11-45.

NOTE: In these examples, the configurations of the edge devices (A, B, E, and F) are identical. The configurations of the core devices (C and D) also are identical. The aggregated VLAN configurations of the edge and core devices on one side must be symmetrical (in fact, a mirror image) to the configurations of the devices on the other side. For simplicity, the example in Figure 11.17 on page 11-45 is symmetrical in terms of the port numbers. This allows the configurations for both sides of the link to be the same. If your configuration does not use symmetrically arranged port numbers, the configurations should not be identical but must use the correct port numbers.

Commands for Device A

```
FastIron SuperX RouterA(config)# vlan 101 by port
FastIron SuperX RouterA(config-vlan-101)# tagged ethernet 2/1
FastIron SuperX RouterA(config-vlan-101)# untagged ethernet 1/1
FastIron SuperX RouterA(config-vlan-101)# exit
FastIron SuperX RouterA(config)# vlan 102 by port
FastIron SuperX RouterA(config-vlan-102)# tagged ethernet 2/1
FastIron SuperX RouterA(config-vlan-102)# untagged ethernet 1/2
FastIron SuperX RouterA(config-vlan-102)# exit
FastIron SuperX RouterA(config)# vlan 103 by port
FastIron SuperX RouterA(config-vlan-103)# tagged ethernet 2/1
FastIron SuperX RouterA(config-vlan-103)# untagged ethernet 1/3
FastIron SuperX RouterA(config-vlan-103)# exit
FastIron SuperX RouterA(config)# vlan 104 by port
FastIron SuperX RouterA(config-vlan-104)# tagged ethernet 2/1
FastIron SuperX RouterA(config-vlan-104)# untagged ethernet 1/4
FastIron SuperX RouterA(config-vlan-104)# exit
FastIron SuperX RouterA(config)# vlan 105 by port
FastIron SuperX RouterA(config-vlan-105)# tagged ethernet 2/1
FastIron SuperX RouterA(config-vlan-105)# untagged ethernet 1/5
FastIron SuperX RouterA(config-vlan-105)# exit
FastIron SuperX RouterA(config)# write memory
```


Commands for Device B

The commands for configuring device B are identical to the commands for configuring device A. Notice that you can use the same channel VLAN numbers on each device. The devices that aggregate the VLANs into a path can distinguish between the identically named channel VLANs based on the ID of the path VLAN.

```
FastIron SuperX RouterB(config)# vlan 101 by port
FastIron SuperX RouterB(config-vlan-101)# tagged ethernet 2/1
FastIron SuperX RouterB(config-vlan-101)# untagged ethernet 1/1
FastIron SuperX RouterB(config-vlan-101)# exit
FastIron SuperX RouterB(config)# vlan 102 by port
FastIron SuperX RouterB(config-vlan-102)# tagged ethernet 2/1
FastIron SuperX RouterB(config-vlan-102)# untagged ethernet 1/2
FastIron SuperX RouterB(config-vlan-102)# exit
FastIron SuperX RouterB(config)# vlan 103 by port
FastIron SuperX RouterB(config-vlan-103)# tagged ethernet 2/1
FastIron SuperX RouterB(config-vlan-103)# untagged ethernet 1/3
FastIron SuperX RouterB(config-vlan-103)# exit
FastIron SuperX RouterB(config)# vlan 104 by port
FastIron SuperX RouterB(config-vlan-104)# tagged ethernet 2/1
FastIron SuperX RouterB(config-vlan-104)# untagged ethernet 1/4
FastIron SuperX RouterB(config-vlan-104)# exit
FastIron SuperX RouterB(config)# vlan 105 by port
FastIron SuperX RouterB(config-vlan-105)# tagged ethernet 2/1
FastIron SuperX RouterB(config-vlan-105)# untagged ethernet 1/5
FastIron SuperX RouterB(config-vlan-105)# exit
FastIron SuperX RouterB(config)# write memory
```

Commands for Device C

Since device C is aggregating channel VLANs from devices A and B into a single path, you need to change the tag type and enable VLAN aggregation.

```
FastIron SuperX RouterC(config)# tag-type 9100
FastIron SuperX RouterC(config)# aggregated-vlan
FastIron SuperX RouterC(config)# vlan 101 by port
FastIron SuperX RouterC(config-vlan-101)# tagged ethernet 4/1
FastIron SuperX RouterC(config-vlan-101)# untagged ethernet 3/1
FastIron SuperX RouterC(config-vlan-101)# exit
FastIron SuperX RouterC(config)# vlan 102 by port
FastIron SuperX RouterC(config-vlan-102)# tagged ethernet 4/1
FastIron SuperX RouterC(config-vlan-102)# untagged ethernet 3/2
FastIron SuperX RouterC(config-vlan-102)# exit
FastIron SuperX RouterC(config)# write memory
```

Commands for Device D

Device D is at the other end of path and separates the channels back into individual VLANs. The tag type must be the same as tag type configured on the other core device (Device C). In addition, VLAN aggregation also must be enabled.

```
FastIron SuperX RouterD(config)# tag-type 9100
FastIron SuperX RouterD(config)# aggregated-vlan
FastIron SuperX RouterD(config)# vlan 101 by port
FastIron SuperX RouterD(config-vlan-101)# tagged ethernet 4/1
FastIron SuperX RouterD(config-vlan-101)# untagged ethernet 3/1
FastIron SuperX RouterD(config-vlan-101)# exit
FastIron SuperX RouterD(config)# vlan 102 by port
FastIron SuperX RouterD(config-vlan-102)# tagged ethernet 4/1
FastIron SuperX RouterD(config-vlan-102)# untagged ethernet 3/2
FastIron SuperX RouterD(config-vlan-102)# exit
FastIron SuperX RouterD(config)# write memory
```


Commands for Device E

Since the configuration in Figure 11.17 on page 11-45 is symmetrical, the commands for configuring device E are identical to the commands for configuring device A.

```
FastIron SuperX RouterE(config)# vlan 101 by port
FastIron SuperX RouterE(config-vlan-101)# tagged ethernet 2/1
FastIron SuperX RouterE(config-vlan-101)# untagged ethernet 1/1
FastIron SuperX RouterE(config-vlan-101)# exit
FastIron SuperX RouterE(config)# vlan 102 by port
FastIron SuperX RouterE(config-vlan-102)# tagged ethernet 2/1
FastIron SuperX RouterE(config-vlan-102)# untagged ethernet 1/2
FastIron SuperX RouterE(config-vlan-102)# exit
FastIron SuperX RouterE(config)# vlan 103 by port
FastIron SuperX RouterE(config-vlan-103)# tagged ethernet 2/1
FastIron SuperX RouterE(config-vlan-103)# untagged ethernet 1/3
FastIron SuperX RouterE(config-vlan-103)# exit
FastIron SuperX RouterE(config)# vlan 104 by port
FastIron SuperX RouterE(config-vlan-104)# tagged ethernet 2/1
FastIron SuperX RouterE(config-vlan-104)# untagged ethernet 1/4
FastIron SuperX RouterE(config-vlan-104)# exit
FastIron SuperX RouterE(config)# vlan 105 by port
FastIron SuperX RouterE(config-vlan-105)# tagged ethernet 2/1
FastIron SuperX RouterE(config-vlan-105)# untagged ethernet 1/5
FastIron SuperX RouterE(config-vlan-105)# exit
FastIron SuperX RouterE(config)# write memory
```

Commands for Device F

The commands for configuring device F are identical to the commands for configuring device E. In this example, since the port numbers on each side of the configuration in Figure 11.17 on page 11-45 are symmetrical, the configuration of device F is also identical to the configuration of device A and device B.

```
FastIron SuperX RouterF(config)# vlan 101 by port
FastIron SuperX RouterF(config-vlan-101)# tagged ethernet 2/1
FastIron SuperX RouterF(config-vlan-101)# untagged ethernet 1/1
FastIron SuperX RouterF(config-vlan-101)# exit
FastIron SuperX RouterF(config)# vlan 102 by port
FastIron SuperX RouterF(config-vlan-102)# tagged ethernet 2/1
FastIron SuperX RouterF(config-vlan-102)# untagged ethernet 1/2
FastIron SuperX RouterF(config-vlan-102)# exit
FastIron SuperX RouterF(config)# vlan 103 by port
FastIron SuperX RouterF(config-vlan-103)# tagged ethernet 2/1
FastIron SuperX RouterF(config-vlan-103)# untagged ethernet 1/3
FastIron SuperX RouterF(config-vlan-103)# exit
FastIron SuperX RouterF(config)# vlan 104 by port
FastIron SuperX RouterF(config-vlan-104)# tagged ethernet 2/1
FastIron SuperX RouterF(config-vlan-104)# untagged ethernet 1/4
FastIron SuperX RouterF(config-vlan-104)# exit
FastIron SuperX RouterF(config)# vlan 105 by port
FastIron SuperX RouterF(config-vlan-105)# tagged ethernet 2/1
FastIron SuperX RouterF(config-vlan-105)# untagged ethernet 1/5
FastIron SuperX RouterF(config-vlan-105)# exit
FastIron SuperX RouterF(config)# write memory
```

Configuring 802.1Q-in-Q Tagging

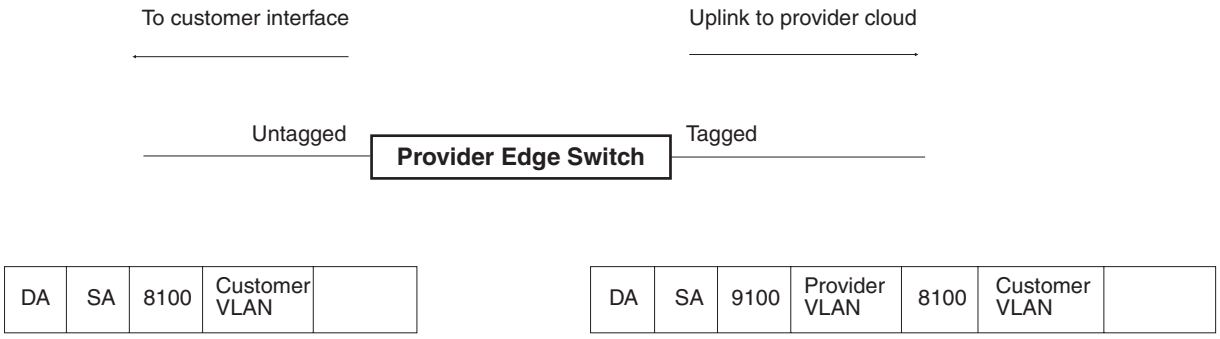
802.1Q tagging is an IEEE standard that enables a networking device to add information to a Layer 2 packet in order to identify the VLAN membership of the packet. Foundry devices tag a packet by adding a four-byte tag to

the packet. The tag contains the tag value, which identifies the data as a tag, and also contains the VLAN ID of the VLAN from which the packet was sent. The tag and VLAN ID keep traffic from each VLAN segregated and private.

- FESX releases prior to 01.1.00 enable you to configure a single 802.1Q tag type on all ports on the device. The default 802.1Q tag on a Foundry device is 8100 (hexadecimal), compliant with the 802.1Q specification.

  Figure 11.18 shows an 802.1Q configuration example with a single 802.1Q tag type.

Figure 11.18 802.1Q Configuration Example



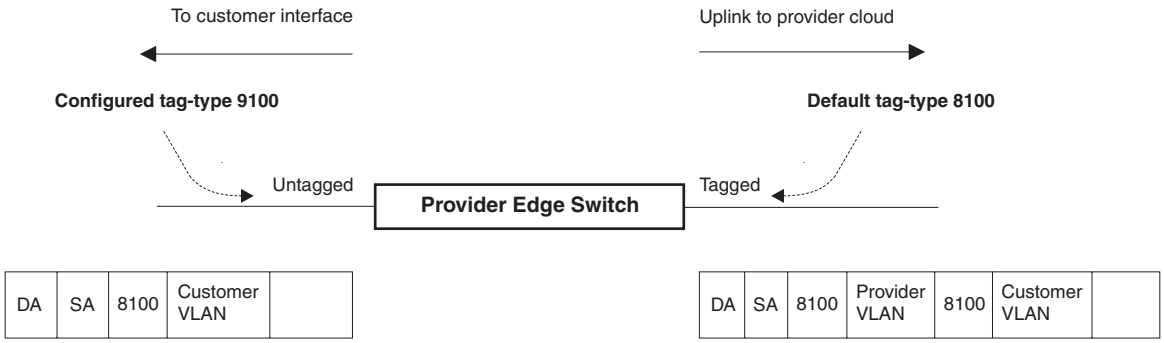
As shown in Figure 11.18, the ports to customer interfaces are untagged, whereas the uplink ports to the provider cloud are tagged, because multiple client VLANs share the uplink to the provider cloud. In this example, the Foundry device treats the customer's private VLAN ID and 8100 tag type as normal payload, and adds the 9100 tag type to the packet when the packet is sent to the uplink and forwarded along the provider cloud.

As long as the switches in the provider's network are Foundry devices or devices that can use the 9100 tag type, the data gets switched along the network. However, devices along the provider's cloud that do not support the 9100 tag type may not properly handle the packets.

- FESX releases 01.1.00 and later, and all FSX and FWSX releases, provide finer granularity for configuring 802.1Q tagging, enabling you to configure 802.1Q tag-types on a group of ports. This type of configuration is called ***802.1Q-in-Q tagging***. This feature enables the creation of two identical 802.1Q tags (802.1Q-in-Q tagging) on a single device. This enhancement improves SAV interoperability between Foundry devices and other vendors' devices that support the 802.1Q tag-types, but are not very flexible with the tag-types they accept.

  Figure 11.19 shows an example application with 802.1Q-in-Q tagging.

Figure 11.19 802.1Q-in-Q Configuration Example



In Figure 11.19, the untagged ports (to customer interfaces) accept frames that have any 802.1Q tag other than the configured tag-type 9100. These packets are considered untagged on this incoming port and are re-tagged when they are sent out of the uplink towards the provider. The 802.1Q tag-type on the uplink port is 8100, so the Foundry device will switch the frames to the uplink device with an additional 8100 tag, thereby supporting devices that only support this method of VLAN tagging.

Configuration Rules

- Since the uplink (to the provider cloud) and the edge link (to the customer port) must have different 802.1Q tags, make sure the uplink and edge link are in different port regions. See “About Port Regions” on page 4-2 for a list of valid port regions.
- If you configure a port with an 802.1Q tag-type, the Foundry device automatically applies the 802.1Q tag-type to all ports within the same port region. Likewise, if you remove the 802.1Q tag-type from a port, the Foundry device automatically removes the 802.1Q tag-type from all ports within the same port region.
- X-Series devices support one configured tag-type per device along with the default tag-type of 8100. For example, if you configure an 802.1Q tag of 9100 on ports 1 – 12, then later configure an 802.1Q tag of 5100 on port 15, the device automatically applies the 5100 tag to all ports in the same port region as port 15, and also changes the 802.1Q tag-type on ports 1 – 12 to 5100.
- 802.1Q-in-Q tagging and VSRP are not supported together on the same device.

Enabling 802.1Q-in-Q Tagging

To enable 802.1Q-in-Q tagging, configure an 802.1Q tag on the untagged edge links (the customer ports) to any value other than the 802.1Q tag for incoming traffic. For example, in Figure 11.20, the 802.1Q tag on the untagged edge links (ports 11 and 12) is 9100, whereas, the 802.1Q tag for incoming traffic is 8100.

To configure 802.1 Q-in-Q tagging as shown in Figure 11.20, enter commands such as the following on the untagged edge links of devices C and D:

```
FESX424 Switch(config)# tag-type 9100 e 11 to 12
FESX424 Switch(config)# aggregated-vlan
```

Note that since ports 11 and 12 belong to the port region 1 – 12, the 802.1Q tag actually applies to ports 1 – 12.

Syntax: [no] tag-type <num> [ethernet [<slotnum>/] <port number> [to <port number>]]

The **<num>** parameter specifies the tag-type number and can be a hexadecimal value from 0 - ffff. The default is 8100.

The **<slotnum>** parameter is required on chassis devices.

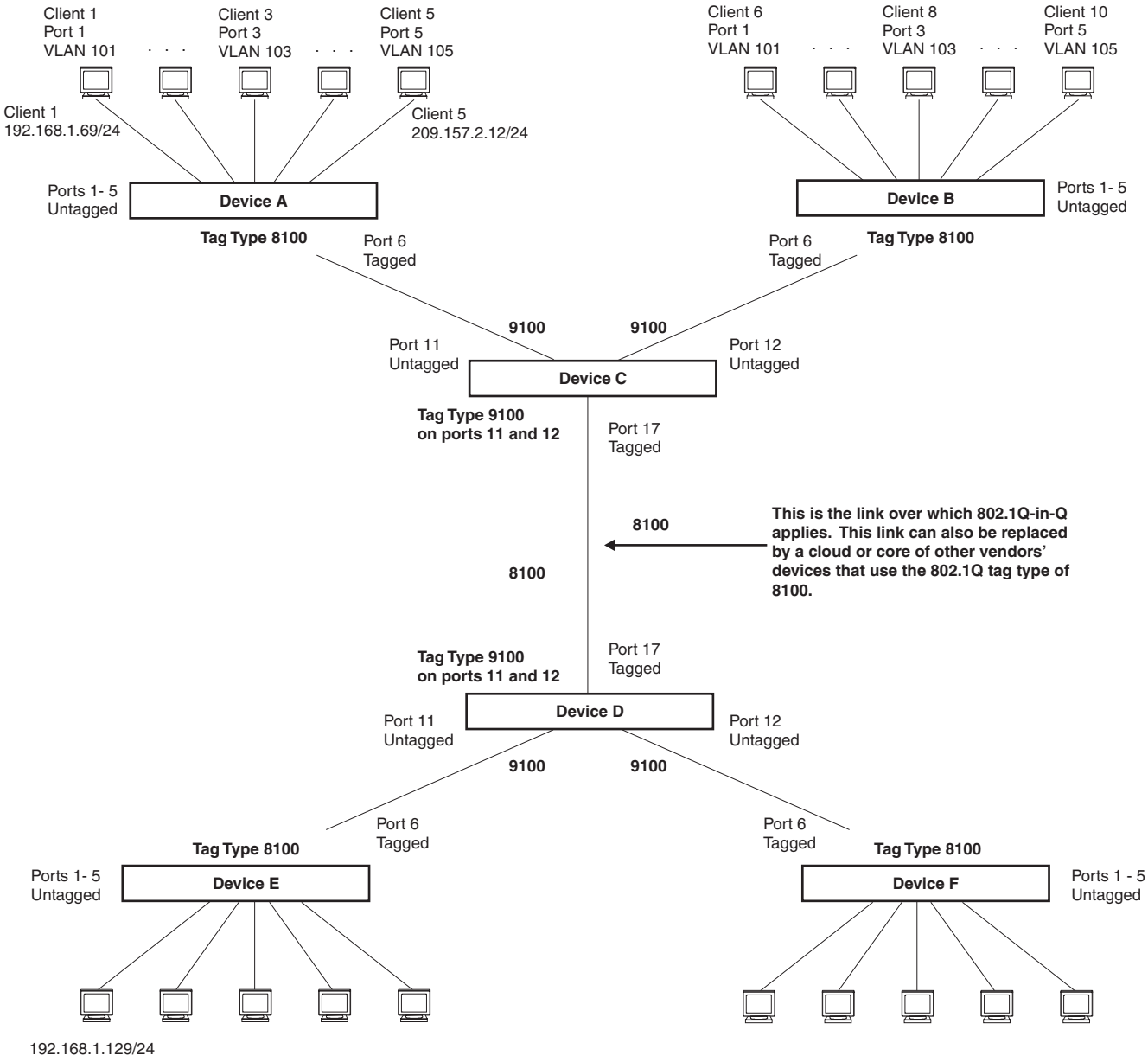
The **ethernet <port number> to <port number>** parameter specifies the port(s) that will use the defined 802.1Q tag. This parameter operates with the following rules:

- If you specify a single port number, the 802.1Q tag applies to all ports within the port region. For example, if you enter the command **tag-type 9100 e 1**, the Foundry device automatically applies the 802.1Q tag to ports 1 – 12 since all of these ports are in the same port region. You can use the **show running-config** command to view how the command has been applied.
- If you do not specify a port or range of ports, the 802.1Q tag applies to all Ethernet ports on the device.

Example Configuration

Figure 11.20 shows an example 802.1Q-in-Q configuration.

Figure 11.20 Example 802.1Q-in-Q Configuration

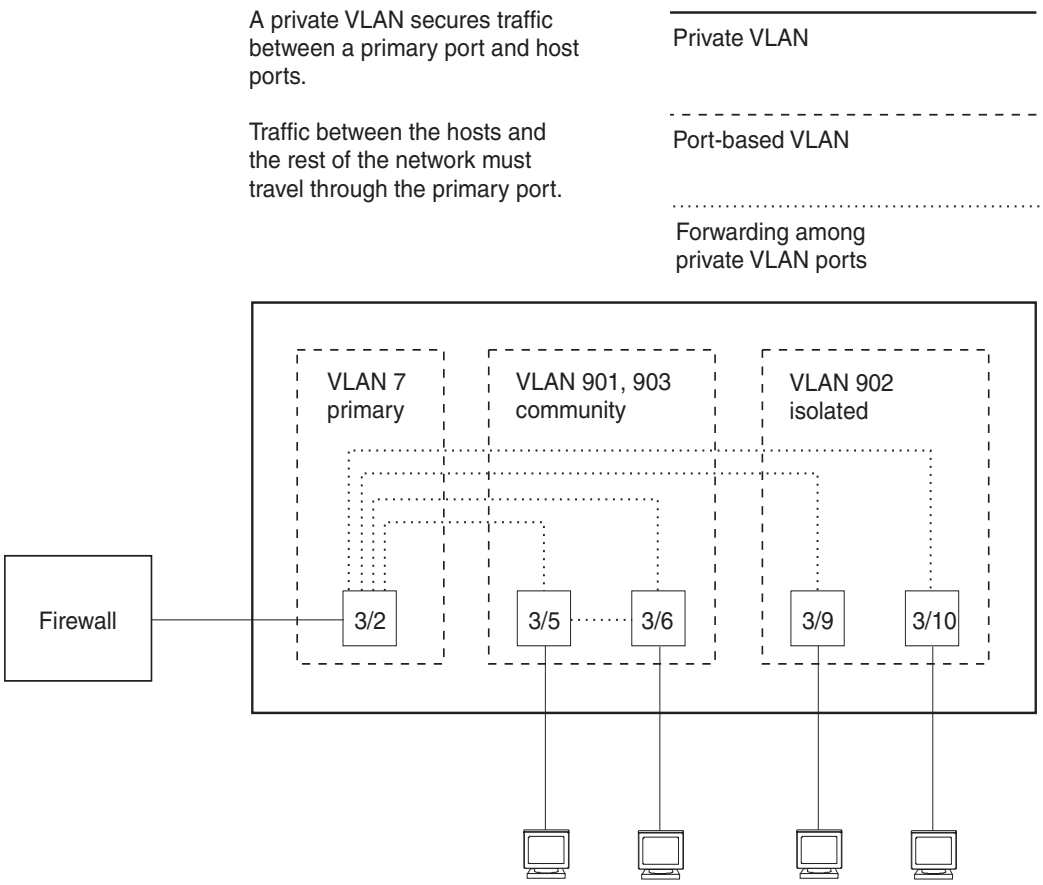


Configuring Private VLANs

NOTE: Software releases 02.4.00 and later support private VLANs on untagged ports. You cannot configure isolated, community, or primary VLANs on 802.1Q tagged ports.

A private VLAN is a VLAN that has the properties of standard Layer 2 port-based VLANs but also provides additional control over flooding packets on a VLAN. Figure 11.21 shows an example of an application using a private VLAN.

Figure 11.21 Private VLAN used to secure communication between a workstation and servers



This example uses a private VLAN to secure traffic between hosts and the rest of the network through a firewall. Five ports in this example are members of a private VLAN. The first port (port 3/2) is attached to a firewall. The next four ports (ports 3/5, 3/6, 3/9, and 3/10) are attached to hosts that rely on the firewall to secure traffic between the hosts and the rest of the network. In this example, two of the hosts (on ports 3/5 and 3/6) are in a community private VLAN, and thus can communicate with one another as well as through the firewall. The other two hosts (on ports 3/9 and 3/10), are in an isolated VLAN and thus can communicate only through the firewall. The two hosts are secured from communicating with one another even though they are in the same VLAN.

By default, the private VLAN does not forward broadcast or unknown-unicast packets from outside sources into the private VLAN. If needed, you can override this behavior for broadcast packets, unknown-unicast packets, or both. (See “Enabling Broadcast or Unknown Unicast Traffic to the Private VLAN” on page 11-55.)

You can configure a combination of the following types of private VLANs:

- Primary – The primary private VLAN ports are “promiscuous”. They can communicate with all the isolated private VLAN ports and community private VLAN ports in the isolated and community VLANs that are mapped to the promiscuous port.
- Isolated – Broadcasts and unknown unicasts received on isolated ports are sent only to the primary port. They are not flooded to other ports in the isolated VLAN.
- Community – Broadcasts and unknown unicasts received on community ports are sent to the primary port and also are flooded to the other ports in the community VLAN.

Each private VLAN must have a primary VLAN. The primary VLAN is the interface between the secured ports and the rest of the network. The private VLAN can have any combination of community and isolated VLANs.

Table 11.3 list the differences between private VLANs and standard VLANs.

Table 11.3: Comparison of Private VLANs and Standard Port-Based VLANs

Forwarding Behavior	Private VLANs	Standard VLANs
All ports within a VLAN constitute a common Layer broadcast domain	No	Yes
Broadcasts and unknown unicasts are forwarded to all the VLAN's ports by default	No (isolated VLAN) Yes (community VLAN)	Yes
Known unicasts	Yes	Yes

Implementation Notes

- Private VLANs are supported in releases 02.4.00 and later on untagged ports only. You cannot configure isolated, community, or primary VLANs on 802.1Q tagged ports.
- When Protocol or Subnet VLANs are enabled, or if Private VLAN mappings are enabled, the Foundry device forwards unknown unicast, unknown multicast, and broadcast packets in software. Normally, the Foundry device forwards unknown unicast, unknown multicast, and broadcast packets in hardware.
- The X-Series devices forward all known unicast and multicast traffic in hardware. This differs from the way the BigIron implements private VLANs, in that the BigIron uses the CPU to forward packets on the primary VLAN's "promiscuous" port. In addition, on the BigIron, support for the hardware forwarding in this feature sometimes results in multiple MAC address entries for the same MAC address in the device's MAC address table. On the X-Series devices, multiple MAC entries do not appear in the MAC address table because the X-Series transparently manages multiple MAC entries in hardware.
- You can configure private VLANs and dual-mode VLAN ports on the same device. However, the dual-mode VLAN ports cannot be members of Private VLANs.
- A primary VLAN can have multiple ports. All these ports are active, but the ports that will be used depends on the private VLAN mappings. Also, secondary VLANs (isolated and community VLANs) can be mapped to multiple primary VLAN ports. For example:

```
pvlan mapping 901 ethernet 1
pvlan mapping 901 ethernet 2
pvlan mapping 901 ethernet 3
```

Command Syntax

To configure a private VLAN, configure each of the component VLANs (isolated, community, and public) as a separate port-based VLAN.

- Use standard VLAN configuration commands to create the VLAN and add ports.
- Identify the private VLAN type (isolated, community, or public)
- For the primary VLAN, map the other private VLANs to the port(s) in the primary VLAN

Configuring an Isolated or Community Private VLAN

To configure a community private VLAN, enter commands such as the following:

```
FastIron SuperX Router(config)# vlan 901
FastIron SuperX Router(config-vlan-901)# untagged ethernet 3/5 to 3/6
FastIron SuperX Router(config-vlan-901)# pvlan type community
```

These commands create port-based VLAN 901, add ports 3/5 and 3/6 to the VLAN as untagged ports, then specify that the VLAN is a community private VLAN.

Syntax: untagged ethernet [<slotnum>/]<portnum> [to [<slotnum>/]<portnum> | ethernet [<slotnum>/]<portnum>]

Syntax: [no] pvlan type community | isolated | primary

The **untagged** command adds the ports to the VLAN.

The **pvlan type** command specifies that this port-based VLAN is a private VLAN.

- **community** – Broadcasts and unknown unicasts received on community ports are sent to the primary port and also are flooded to the other ports in the community VLAN.
- **isolated** – Broadcasts and unknown unicasts received on isolated ports are sent only to the primary port. They are not flooded to other ports in the isolated VLAN.
- **primary** – The primary private VLAN ports are “promiscuous”. They can communicate with all the isolated private VLAN ports and community private VLAN ports in the isolated and community VLANs that are mapped to the promiscuous port.

Configuring the Primary VLAN

NOTE: The primary private VLAN has only one active port. If you configure the VLAN to have more than one port, the lowest-numbered port is the active one. The additional ports provide redundancy. If the active port becomes unavailable, the lowest-numbered available port becomes the active port for the VLAN.

To configure a primary private VLAN, enter commands such as the following:

```
FastIron SuperX Router(config)# vlan 7
FastIron SuperX Router(config-vlan-7)# untagged ethernet 3/2
FastIron SuperX Router(config-vlan-7)# pvlan type primary
FastIron SuperX Router(config-vlan-7)# pvlan mapping 901 ethernet 3/2
```

These commands create port-based VLAN 7, add port 3/2 as an untagged port, identify the VLAN as the primary VLAN in a private VLAN, and map the other private VLANs to the port(s) in this VLAN.

Syntax: untagged ethernet [<slotnum>/]<portnum> [to [<slotnum>/]<portnum> | ethernet [<slotnum>/]<portnum>]

Syntax: [no] pvlan type community | isolated | primary

Syntax: [no] pvlan mapping <vlan-id> ethernet [<slotnum>/]<portnum>

The **untagged** command adds the port(s) to the VLAN.

The **pvlan type** command specifies that this port-based VLAN is a private VLAN. Specify **primary** as the type.

The **pvlan mapping** command identifies the other private VLANs for which this VLAN is the primary. The command also specifies the primary VLAN ports to which you are mapping the other private VLANs.

- The <vlan-id> parameter specifies another private VLAN. The other private VLAN you want to specify must already be configured.
- The **ethernet** <portnum> parameter specifies the primary VLAN port to which you are mapping all the ports in the other private VLAN (the one specified by <vlan-id>).

Enabling Broadcast or Unknown Unicast Traffic to the Private VLAN

To enhance private VLAN security, the primary private VLAN does not forward broadcast or unknown unicast packets to its community and isolated VLANs. For example, if port 3/2 in Figure 11.21 on page 11-53 receives a broadcast packet from the firewall, the port does not forward the packet to the other private VLAN ports (3/5, 3/6, 3/9, and 3/10).

This forwarding restriction does not apply to traffic from the private VLAN. The primary port does forward broadcast and unknown unicast packets that are received from the isolated and community VLANs. For example, if the host on port 3/9 sends an unknown unicast packet, port 3/2 forwards the packet to the firewall.

If you want to remove the forwarding restriction, you can enable the primary port to forward broadcast or unknown unicast traffic, if desired, using the following CLI method. You can enable or disable forwarding of broadcast or unknown unicast packets separately.

NOTE: On Layer 2 Switches and Layer 3 Switches, you also can use MAC address filters to control the traffic forwarded into and out of the private VLAN. In addition, if you are using a Layer 2 Switch, you also can use ACLs.

Command Syntax

To configure the ports in the primary VLAN to forward broadcast or unknown unicast traffic received from sources outside the private VLAN, enter the following commands at the global CONFIG level of the CLI:

```
FastIron SuperX Router(config)# pvlan-preference broadcast flood
FastIron SuperX Router(config)# pvlan-preference unknown-unicast flood
```

These commands enable forwarding of broadcast and unknown-unicast packets to ports within the private VLAN. To again disable forwarding, enter a command such as the following:

```
FastIron SuperX Router(config)# no pvlan-preference broadcast flood
```

This command disables forwarding of broadcast packets within the private VLAN.

Syntax: [no] pvlan-preference broadcast | unknown-unicast flood

CLI Example for Figure 11.21

To configure the private VLANs shown in Figure 11.21 on page 11-53, enter the following commands:

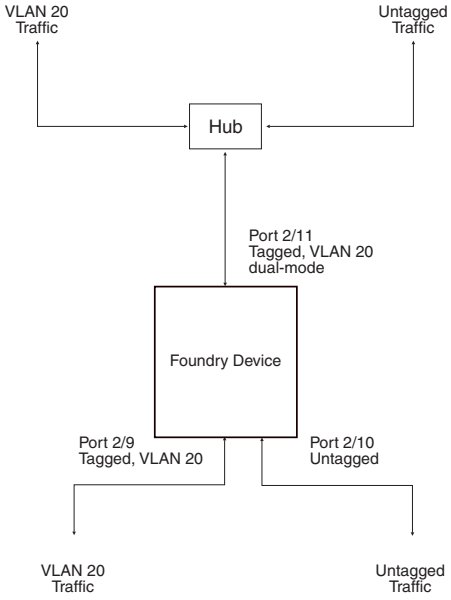
```
FastIron SuperX Router(config)# vlan 901
FastIron SuperX Router(config-vlan-901)# untagged ethernet 3/5 to 3/6
FastIron SuperX Router(config-vlan-901)# pvlan type community
FastIron SuperX Router(config-vlan-901)# exit
FastIron SuperX Router(config)# vlan 902
FastIron SuperX Router(config-vlan-902)# untagged ethernet 3/9 to 3/10
FastIron SuperX Router(config-vlan-902)# pvlan type isolated
FastIron SuperX Router(config-vlan-902)# exit
FastIron SuperX Router(config)# vlan 903
FastIron SuperX Router(config-vlan-903)# untagged ethernet 3/5 to 3/6
FastIron SuperX Router(config-vlan-903)# pvlan type community
FastIron SuperX Router(config-vlan-903)# exit
FastIron SuperX Router(config)# vlan 7
FastIron SuperX Router(config-vlan-7)# untagged ethernet 3/2
FastIron SuperX Router(config-vlan-7)# pvlan type primary
FastIron SuperX Router(config-vlan-7)# pvlan mapping 901 ethernet 3/2
FastIron SuperX Router(config-vlan-7)# pvlan mapping 902 ethernet 3/2
FastIron SuperX Router(config-vlan-7)# pvlan mapping 903 ethernet 3/2
```

Dual-Mode VLAN Ports

Configuring a tagged port as a ***dual-mode*** port allows it to accept and transmit both tagged traffic and untagged traffic at the same time. A dual-mode port accepts and transmits frames belonging to VLANs configured for the port, as well as frames belonging to the default VLAN (that is, untagged traffic).

For example, in Figure 11.22, port 2/11 is a dual-mode port belonging to VLAN 20. Traffic for VLAN 20, as well as traffic for the default VLAN, flows from a hub to this port. The dual-mode feature allows traffic for VLAN 20 and untagged traffic to go through the port at the same time.

Figure 11.22 Dual-mode VLAN port example



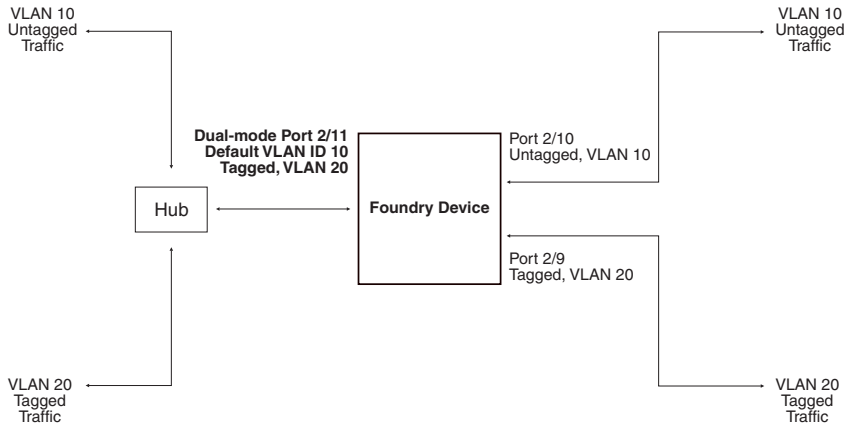
To enable the dual-mode feature on port 2/11 in Figure 11.22:

```
FastIron SuperX Router(config)# vlan 20
FastIron SuperX Router(config-vlan-20)# tagged e 2/11
FastIron SuperX Router(config-vlan-20)# tagged e 2/9
FastIron SuperX Router(config-vlan-20)# int e 2/11
FastIron SuperX Router(config-if-e1000-2/11)# dual-mode
FastIron SuperX Router(config-if-e1000-2/11)# exit
```

Syntax: [no] dual-mode

You can configure a dual-mode port to transmit traffic for a specified VLAN (other than the DEFAULT-VLAN) as untagged, while transmitting traffic for other VLANs as tagged. Figure 11.23 illustrates this enhancement.

Figure 11.23 Specifying a default VLAN ID for a dual-mode port



In Figure 11.23, tagged port 2/11 is a dual-mode port belonging to VLANs 10 and 20. The default VLAN assigned to this dual-mode port is 10. This means that the port transmits tagged traffic on VLAN 20 (and all other VLANs to which the port belongs) and transmits untagged traffic on VLAN 10.

The dual-mode feature allows tagged traffic for VLAN 20 and untagged traffic for VLAN 10 to go through port 2/11 at the same time. A dual-mode port transmits only untagged traffic on its default VLAN (that is, either VLAN 1, or a user-specified VLAN ID), and only tagged traffic on all other VLANs.

The following commands configure VLANs 10 and 20 in Figure 11.23. Tagged port 2/11 is added to VLANs 10 and 20, then designated a dual-mode port whose specified default VLAN is 10. In this configuration, port 2/11 transmits only untagged traffic on VLAN 10 and only tagged traffic on VLAN 20.

```
FastIron SuperX Router(config)# vlan 10 by port
FastIron SuperX Router(config-vlan-10)# untagged e 2/10
FastIron SuperX Router(config-vlan-10)# tagged e 2/11
FastIron SuperX Router(config-vlan-10)# exit

FastIron SuperX Router(config)# vlan 20 by port
FastIron SuperX Router(config-vlan-20)# tagged e 2/9
FastIron SuperX Router(config-vlan-20)# tagged e 2/11
FastIron SuperX Router(config-vlan-20)# exit

FastIron SuperX Router(config)# int e 2/11
FastIron SuperX Router(config-if-e1000-2/11)# dual-mode 10
FastIron SuperX Router(config-if-e1000-2/11)# exit
```

Syntax: [no] dual-mode [<vlan-id>]

Notes:

- If you do not specify a <vlan-id> in the **dual mode** command, the port's default VLAN is set to 1. The port transmits untagged traffic on the DEFAULT-VLAN.
- The dual-mode feature is disabled by default. Only tagged ports can be configured as dual-mode ports.
- In trunk group, either all of the ports must be dual-mode, or none of them can be.

The **show vlan** command displays a separate row for dual-mode ports on each VLAN. For example:

```
FastIron SuperX Router(config)# show vlan
Total PORT-VLAN entries: 3
Maximum PORT-VLAN entries: 16

legend: [S=Slot]

PORT-VLAN 1, Name DEFAULT-VLAN, Priority level0, Spanning tree Off
 Untagged Ports: (S1)  1  2  3  4  5  6  7  8
 Untagged Ports: (S2)  1  2  3  4  5  6  7  8 12 13 14 15 16 17 18 19
 Untagged Ports: (S2) 20 21 22 23 24
   Tagged Ports: None
   Uplink Ports: None
 DualMode Ports: None
PORT-VLAN 10, Name [None], Priority level0, Spanning tree Off
 Untagged Ports: (S2) 10
   Tagged Ports: None
   Uplink Ports: None
 DualMode Ports: (S2) 11
PORT-VLAN 20, Name [None], Priority level0, Spanning tree Off
 Untagged Ports: None
   Tagged Ports: (S2)  9
   Uplink Ports: None
 DualMode Ports: (S2) 11
```


Displaying VLAN Information

After you configure the VLANs, you can verify the configuration using the following methods.

NOTE: If a VLAN name begins with "GVRP_VLAN_", the VLAN was created by the GARP VLAN Registration Protocol (GVRP). If a VLAN name begins with "STATIC_VLAN_", the VLAN was created by GVRP and then was converted into a statically configured VLAN.

Displaying System-Wide VLAN Information

Use one of the following methods to display VLAN information for all the VLANs configured on the device.

Enter the following command at any CLI level. This example shows the display for the IP sub-net and IPX network VLANs configured in the examples in "Configuring an IP Sub-Net VLAN with Dynamic Ports" on page 11-34 and "Configuring an IPX Network VLAN with Dynamic Ports" on page 11-34.

```
FastIron SuperX Router(config)# show vlans

Total PORT-VLAN entries: 2
Maximum PORT-VLAN entries: 8
legend: [S=Slot]

PORT-VLAN 1, Name DEFAULT-VLAN, Priority level0, Spanning tree Off
 Untagged Ports: (S2)  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16
 Untagged Ports: (S2) 17 18 19 20 21 22 23 24
 Untagged Ports: (S4)  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16
 Untagged Ports: (S4) 17 18 19 20 21 22 23 24
   Tagged Ports: None

PORT-VLAN 10, Name IP_VLAN, Priority level0, Spanning tree Off
 Untagged Ports: (S1)  1  2  3  4  5  6
   Tagged Ports: None

 IP-subnet VLAN 1.1.1.0 255.255.255.0, Dynamic port enabled
          Name: Mktg-LAN
  Static ports: None
 Exclude ports: None
 Dynamic ports: (S1)  1  2  3  4  5  6
 PORT-VLAN 20, Name IPX_VLAN, Priority level0, Spanning tree Off
 Untagged Ports: (S2)  1  2  3  4  5  6
   Tagged Ports: None

 IPX-network VLAN 0000ABCD, frame type ethernet_ii, Dynamic port enabled
          Name: Eng-LAN
  Static ports: None
 Exclude ports: None
 Dynamic ports: (S2)  1  2  3  4  5  6
```

Syntax: show vlans [<vlan-id> | ethernet [<slotnum>/]<portnum>]

The <vlan-id> parameter specifies a VLAN for which you want to display the configuration information.

The <slotnum> parameter is required on chassis devices.

The <portnum> parameter specifies a port. If you use this parameter, the command lists all the VLAN memberships for the port.

Displaying VLAN Information for Specific Ports

Use one of the following methods to display VLAN information for specific ports.

To display VLAN information for all the VLANs of which port 7/1 is a member, enter the following command:

```
FastIron SuperX Router(config)# show vlans e 7/1

Total PORT-VLAN entries: 3
Maximum PORT-VLAN entries: 8

legend: [S=Slot]

PORT-VLAN 100, Name [None], Priority level0, Spanning tree Off
 Untagged Ports: (S7)  1  2  3  4
   Tagged Ports: None

 IP-subnet VLAN 207.95.11.0 255.255.255.0, Dynamic port disabled
 Static ports: (S7)  1  2
 Exclude ports: None
 Dynamic ports: None
```

Syntax: show vlans [<vlan-id> | ethernet [<slotnum>/]<portnum>

The <vlan-id> parameter specifies a VLAN for which you want to display the configuration information.

The <slotnum> parameter is required on chassis devices.

The <portnum> parameter specifies a port. If you use this parameter, the command lists all the VLAN memberships for the port.


```
FastIron SuperX Router# show vlan-group
vlan-group 1 vlan 2 to 20
 tagged ethe 1/1 to 1/2
!
vlan-group 2 vlan 21 to 40
 tagged ethe 1/1 to 1/2
!
```

Chapter 12
Rule-Based IP Access Control Lists (ACLs)

FESX, FSX, and FWSX devices support ***rule-based ACLs*** (sometimes called hardware-based ACLs), where the decisions to permit or deny packets are processed in hardware and all permitted packets are switched or routed in hardware.

Rule-based ACLs program the ACL entries you assign to an interface into Content Addressable Memory (CAM) space allocated for the port(s). The ACLs are programmed into hardware at startup (or as new ACLs are entered and bound to ports). Devices that use rule-based ACLs program the ACLs into the CAM entries and use these entries to permit or deny packets in the hardware, without sending the packets to the CPU for processing.

Rule-based ACLs are supported on physical interfaces, trunk groups, and virtual routing interfaces.

NOTE: The FESX, FSX, and FWSX devices support hardware-based ACLs only. These devices do not support flow-based ACLs. In contrast, FES devices support flow-based ACLs only.

This chapter contains the following information:

Table 12.1: Chapter Contents

Table 12.1: Chapter Contents

ACL Overview

This section provides an overview of ACLs.

Types of IP ACLs

You can configure the following types of IP ACLs:

- ***Standard*** – Permits or denies packets based on source IP address. Valid standard ACL IDs are 1 – 99 or a character string.
- ***Extended*** – Permits or denies packets based on source and destination IP address and also based on IP protocol information. Valid extended ACL IDs are a number from 100 – 199 or a character string.

ACL IDs and Entries

ACLs consist of ACL IDs and ACL entries:

- ***ACL ID*** – An ACL ID is a number from 1 – 99 (for a standard ACL) or 100 – 199 (for an extended ACL) or a character string. The ACL ID identifies a collection of individual ACL entries. When you apply ACL entries to an interface, you do so by applying the ACL ID that contains the ACL entries to the interface, instead of applying the individual entries to the interface. This makes applying large groups of access filters (ACL entries) to interfaces simple. See also “Numbered and Named ACLs” on page 12-3.

  NOTE: This is different from IP access policies. If you use IP access policies, you apply the individual policies to interfaces.

- ***ACL entry*** – Also called an ***ACL rule***, a filter command associated with an ACL ID. The maximum number of ACL rules you can configure is a system-wide parameter and depends on the device you are configuring. You can configure up to the maximum number of entries in any combination in different ACLs. The total number of entries in all ACLs cannot exceed the system maximum.
  - One-Gigabit ports on the FESX support up to 1016 ACL rules. On the FSX, multiple ACL groups share 1016 ACL rules per port region. Each ACL group must contain one entry for the implicit *deny all IP traffic* clause. Also, each ACL group uses a multiple of 8 ACL entries. For example, if all ACL groups contain 5 ACL entries, you could add 127ACL groups (1016/8) in that port region. If all your ACL groups contain 8 ACL entries, you could add 63 ACL groups, since you must account for the implicit deny entry.
  - 10-Gigabit ports on the FESX and FSX support up to 1024 ACL rules.

You configure ACLs on a global basis, then apply them to the incoming or outgoing traffic on specific ports. You can apply only one ACL to a port’s inbound traffic and only one ACL to a port’s outbound traffic. The software applies the entries within an ACL in the order they appear in the ACL’s configuration. As soon as a match is found,

the software takes the action specified in the ACL entry (permit or deny the packet) and stops further comparison for that packet.

Numbered and Named ACLs

When you configure an ACL, you can refer to the ACL by a numeric ID or by an alphanumeric name. The commands to configure numbered ACLs are different from the commands for named ACLs.

- ***Numbered ACL*** – If you refer to the ACL by a numeric ID, you can use 1 – 99 for a standard ACL or 100 – 199 for an extended ACL.
- ***Named ACL*** – If you refer to the ACL by a name, you specify whether the ACL is a standard ACL or an extended ACL, then specify the name.

You can configure up to 99 standard numbered IP ACLs and 99 extended numbered IP ACLs. You also can configure up to 99 standard named ACLs and 99 extended named ACLs by number. Regardless of how many ACLs you have, the device can have a maximum of 1024 ACL entries, associated with the ACLs in any combination.

Default ACL Action

The default action when no ACLs are configured on a device is to permit all traffic. However, once you configure an ACL and apply it to a port, the default action for that port is to deny all traffic that is not explicitly permitted on the port.

- If you want to tightly control access, configure ACLs consisting of permit entries for the access you want to permit. The ACLs implicitly deny all other access.
- If you want to secure access in environments with many users, you might want to configure ACLs that consist of explicit deny entries, then add an entry to permit all access to the end of each ACL. The software permits packets that are not denied by the deny entries.

NOTE: Do not apply an empty ACL (an ACL ID without any corresponding entries) to an interface. If you accidentally do this, the software applies the default ACL action, deny all, to the interface and thus denies all traffic.

How Hardware-Based ACLs Work

When you bind an ACL to inbound traffic on an interface, the device programs the Layer 4 CAM with the ACL. Permit and deny rules are programmed. Most ACL rules require one Layer 4 CAM entry. However, ACL rules that match on more than one TCP or UDP application port may require several CAM entries. The Layer 4 CAM entries for ACLs do not age out. They remain in the CAM until you remove the ACL.

- If a packet received on the interface matches an ACL rule in the Layer 4 CAM, the device permits or denies the packet according to the ACL.
- If a packet does not match an ACL rule, the packet is dropped, since the default action on an interface that has ACLs is to deny the packet.

How Fragmented Packets are Processed

The descriptions above apply to non-fragmented packets. The default processing of fragments by hardware-based ACLs is as follows:

- The first fragment of a packet is permitted or denied using the ACLs. The first fragment is handled the same way as non-fragmented packets, since the first fragment contains the Layer 4 source and destination application port numbers. The device uses the Layer 4 CAM entry if one is programmed, or applies the interface's ACL entries to the packet and permits or denies the packet according to the first matching ACL.
- For other fragments of the same packet, they are subject to a rule only if there is no Layer 4 information in the rule or in any preceding rules.

The fragments are forwarded even if the first fragment, which contains the Layer 4 information, was denied. Generally, denying the first fragment of a packet is sufficient, since a transaction cannot be completed without the entire packet.

For tighter control, you can configure the port to drop all packet fragments. See "Enabling Strict Control of ACL Filtering of Fragmented Packets" on page 12-20.

Hardware Aging of Layer 4 CAM Entries

Rule-based ACLs use Layer 4 CAM entries. The device permanently programs rule-based ACLs into the CAM. The entries never age out.

Configuration Considerations

- Hardware-based ACLs are supported on all Ethernet ports and on 10 Gigabit Ethernet ports.
- Hardware-based ACLs are supported on physical interfaces, trunk groups, and virtual routing interfaces.
- Hardware-based ACLs are supported only for inbound traffic.
- ACLs on the FESX, FSX, and FWSX apply to all traffic, including management traffic.
- ACL logging is supported for packets that are sent to the CPU for processing. ACL logging is not supported for packets that are processed in hardware.
- Hardware-based ACLs support only one ACL per port. The ACL of course can contain multiple entries (rules). For example, hardware-based ACLs do not support ACLs 101 and 102 on port 1, but hardware-based ACLs do support ACL 101 containing multiple entries.
- One-Gigabit ports on all FESX and FWSX devices support up to 1016 ACL rules. 10-Gigabit ports on all FESX and FWSX devices support up to 1024 ACL rules. ACLs on the FSX are affected by port regions. Multiple ACL groups share 1016 ACL rules per port region. Each ACL group must contain one entry for the implicit *deny all IP traffic* clause. Also, each ACL group uses a multiple of 8 ACL entries. For example, if all ACL groups contain 5 ACL entries, you could add 127ACL groups (1016/8) in that port region. If all your ACL groups contain 8 ACL entries, you could add 63 ACL groups, since you must account for the implicit deny entry.
- By default, the first fragment of a fragmented packet received by the Foundry device is permitted or denied using the ACLs, but subsequent fragments of the same packet are forwarded in hardware. Generally, denying the first fragment of a packet is sufficient, since a transaction cannot be completed without the entire packet.
- The following ACL features and options are not supported on the FESX and FSX:
  - Applying an ACL on a device that has Super Aggregated VLANs (SAVs) enabled.
  - Enabling CPU filtering of all fragmented packets on a port (**ip access-group frag inspect** command)
  - Configuring a port to drop all packet fragments (**ip access-group frag deny** command)
  - Flow-based ACLs
  - ACL statistics

NOTE: You can apply an ACL to a port that has TCP SYN protection and/or ICMP smurf protection enabled.

Configuring Standard Numbered ACLs

This section describes how to configure standard numbered ACLs with numeric IDs and provides configuration examples.

Standard ACLs permit or deny packets based on source IP address. You can configure up to 99 standard numbered ACLs. There is no limit to the number of ACL entries an ACL can contain except for the system-wide limitation. For the number of ACL entries supported on a device, see "ACL IDs and Entries" on page 12-2.

Standard Numbered ACL Syntax

Syntax: [no] access-list <acl-num> deny | permit <source-ip> | <hostname> <wildcard> [log]

or

Syntax: [no] access-list <acl-num> deny | permit <source-ip>/<mask-bits> | <hostname> [log]

Syntax: [no] access-list <acl-num> deny | permit host <source-ip> | <hostname> [log]

Syntax: [no] access-list <acl-num> deny | permit any [log]

Syntax: [no] ip access-group <acl-num> in

The <acl-num> parameter is the access list number from 1 – 99.

The **deny | permit** parameter indicates whether packets that match a policy in the access list are denied (dropped) or permitted (forwarded).

The <source-ip> parameter specifies the source IP address. Alternatively, you can specify the host name.

NOTE: To specify the host name instead of the IP address, the host name must be configured using the Foundry device's DNS resolver. To configure the DNS resolver name, use the **ip dns server-address…** command at the global CONFIG level of the CLI.

The <wildcard> parameter specifies the mask value to compare against the host address specified by the <source-ip> parameter. The <wildcard> is a four-part value in dotted-decimal notation (IP address format) consisting of ones and zeros. Zeros in the mask mean the packet's source address must match the <source-ip>. Ones mean any value matches. For example, the <source-ip> and <wildcard> values 209.157.22.26 0.0.0.255 mean that all hosts in the Class C subnet 209.157.22.x match the policy.

If you prefer to specify the wildcard (mask value) in CIDR format, you can enter a forward slash after the IP address, then enter the number of significant bits in the mask. For example, you can enter the CIDR equivalent of "209.157.22.26 0.0.0.255" as "209.157.22.26/24". The CLI automatically converts the CIDR number into the appropriate ACL mask (where zeros instead of ones are the significant bits) and changes the non-significant portion of the IP address into ones. For example, if you specify 209.157.22.26/24 or 209.157.22.26 0.0.0.255, then save the changes to the startup-config file, the value appears as 209.157.22.0/24 (if you have enabled display of subnet lengths) or 209.157.22.0 0.0.0.255 in the startup-config file.

If you enable the software to display IP subnet masks in CIDR format, the mask is saved in the file in "/<mask-bits>" format. To enable the software to display the CIDR masks, enter the **ip show-subnet-length** command at the global CONFIG level of the CLI. You can use the CIDR format to configure the ACL entry regardless of whether the software is configured to display the masks in CIDR format.

NOTE: If you use the CIDR format, the ACL entries appear in this format in the running-config and startup-config files, but are shown with subnet mask in the display produced by the **show ip access-list** command.

The **host** <source-ip> | <hostname> parameter lets you specify a host IP address or name. When you use this parameter, you do not need to specify the mask. A mask of all zeros (0.0.0.0) is implied.

The **any** parameter configures the policy to match on all host addresses.

The **log** argument configures the device to generate Syslog entries and SNMP traps for packets that are permitted or denied by the access policy. If you use the **log** argument, the ACL entry is sent to the CPU for processing.

NOTE: You can enable logging on ACLs and filters that support logging even when the ACLs and filters are already in use. To do so, re-enter the ACL or filter command and add the **log** parameter to the end of the ACL or filter. The software replaces the ACL or filter command with the new one. The new ACL or filter, with logging enabled, takes effect immediately.

The **in** parameter applies the ACL to incoming traffic on the interface to which you apply the ACL. You can apply the ACL to an Ethernet port or virtual interface.

NOTE: If the ACL is for a virtual routing interface, you also can specify a subset of ports within the VLAN containing that interface when assigning an ACL to the interface.

Configuration Example for Standard Numbered ACLs

To configure a standard ACL and apply it to incoming traffic on port 1/1, enter the following commands.

```
FastIron SuperX Router(config)# access-list 1 deny host 209.157.22.26 log
FastIron SuperX Router(config)# access-list 1 deny 209.157.29.12 log
FastIron SuperX Router(config)# access-list 1 deny host IPHost1 log
FastIron SuperX Router(config)# access-list 1 permit any
FastIron SuperX Router(config)# int eth 1/1
FastIron SuperX Router(config-if-1/1)# ip access-group 1 in
FastIron SuperX Router(config)# write memory
```

The commands in this example configure an ACL to deny packets from three source IP addresses from being received on port 1/1. The last ACL entry in this ACL permits all packets that are not explicitly denied by the first three ACL entries.

Configuring Standard Named ACLs

This section describes how to configure standard named ACLs with alphanumeric IDs. This section also provides configuration examples.

Standard ACLs permit or deny packets based on source IP address. You can configure up to 99 standard named ACLs. There is no limit to the number of ACL entries an ACL can contain except for the system-wide limitation. For the number of ACL entries supported on a device, see “ACL IDs and Entries” on page 12-2.

The commands for configuring named ACL entries are different from the commands for configuring numbered ACL entries. The command to configure a numbered ACL is **access-list**. The command for configuring a named ACL is **ip access-list**. In addition, when you configure a numbered ACL entry, you specify all the command parameters on the same command. When you configure a named ACL, you specify the ACL type (standard or extended) and the ACL name with one command, which places you in the configuration level for that ACL. Once you enter the configuration level for the ACL, the command syntax is the same as the syntax for numbered ACLs.

Standard Named ACL Syntax

Syntax: [no] ip access-list standard <acl-name> | <acl-num>

Syntax: deny | permit <source-ip> | <hostname> <wildcard> [log]

or

Syntax: deny | permit <source-ip>/<mask-bits> | <hostname> [log]

Syntax: deny | permit host <source-ip> | <hostname> [log]

Syntax: deny | permit any [log]

Syntax: [no] ip access-group <acl-name> in

The <acl-name> parameter is the access list name. You can specify a string of up to 256 alphanumeric characters. You can use blanks in the ACL name if you enclose the name in quotation marks (for example, “ACL for Net1”).

The <acl-num> parameter allows you to specify an ACL number if you prefer. If you specify a number, you can specify from 1 – 99 for standard ACLs.

NOTE: For convenience, the software allows you to configure numbered ACLs using the syntax for named ACLs. The software also still supports the older syntax for numbered ACLs. Although the software allows both methods for configuring numbered ACLs, numbered ACLs are always formatted in the startup-config and running-config files in using the older syntax, as follows.


```
access-list 1 deny host 209.157.22.26 log
access-list 1 deny 209.157.22.0 0.0.0.255 log
access-list 1 permit any
access-list 101 deny tcp any any eq http log
```

The **deny** | **permit** parameter indicates whether packets that match a policy in the access list are denied (dropped) or permitted (forwarded).

The <source-ip> parameter specifies the source IP address. Alternatively, you can specify the host name.

NOTE: To specify the host name instead of the IP address, the host name must be configured using the Foundry device's DNS resolver. To configure the DNS resolver name, use the **ip dns server-address…** command at the global CONFIG level of the CLI.

The <wildcard> parameter specifies the mask value to compare against the host address specified by the <source-ip> parameter. The <wildcard> is a four-part value in dotted-decimal notation (IP address format) consisting of ones and zeros. Zeros in the mask mean the packet's source address must match the <source-ip>. Ones mean any value matches. For example, the <source-ip> and <wildcard> values 209.157.22.26 0.0.0.255 mean that all hosts in the Class C subnet 209.157.22.x match the policy.

If you prefer to specify the wildcard (mask value) in CIDR format, you can enter a forward slash after the IP address, then enter the number of significant bits in the mask. For example, you can enter the CIDR equivalent of "209.157.22.26 0.0.0.255" as "209.157.22.26/24". The CLI automatically converts the CIDR number into the appropriate ACL mask (where zeros instead of ones are the significant bits) and changes the non-significant portion of the IP address into ones. For example, if you specify 209.157.22.26/24 or 209.157.22.26 0.0.0.255, then save the changes to the startup-config file, the value appears as 209.157.22.0/24 (if you have enabled display of subnet lengths) or 209.157.22.0 0.0.0.255 in the startup-config file.

If you enable the software to display IP subnet masks in CIDR format, the mask is saved in the file in "/<mask-bits>" format. To enable the software to display the CIDR masks, enter the **ip show-subnet-length** command at the global CONFIG level of the CLI. You can use the CIDR format to configure the ACL entry regardless of whether the software is configured to display the masks in CIDR format.

NOTE: If you use the CIDR format, the ACL entries appear in this format in the running-config and startup-config files, but are shown with subnet mask in the display produced by the **show ip access-list** command.

The **host** <source-ip> | <hostname> parameter lets you specify a host IP address or name. When you use this parameter, you do not need to specify the mask. A mask of all zeros (0.0.0.0) is implied.

The **any** parameter configures the policy to match on all host addresses.

The **log** argument configures the device to generate Syslog entries and SNMP traps for packets that are permitted or denied by the access policy. If you use the **log** argument, the ACL entry is sent to the CPU for processing.

NOTE: You can enable logging on ACLs and filters that support logging even when the ACLs and filters are already in use. To do so, re-enter the ACL or filter command and add the **log** parameter to the end of the ACL or filter. The software replaces the ACL or filter command with the new one. The new ACL or filter, with logging enabled, takes effect immediately.

The **in** parameter applies the ACL to incoming traffic on the interface to which you apply the ACL. You can apply the ACL to an Ethernet port or virtual interface.

NOTE: If the ACL is bound to a virtual routing interface, you also can specify a subset of ports within the VLAN containing that interface when assigning an ACL to the interface.

Configuration Example for Standard Named ACLs

To configure a standard named ACL, enter commands such as the following.

```
FESX424 Router(config)# ip access-list standard Net1
FESX424 Router(config-std-nacl)# deny host 209.157.22.26 log
FESX424 Router(config-std-nacl)# deny 209.157.29.12 log
FESX424 Router(config-std-nacl)# deny host IPHost1 log
FESX424 Router(config-std-nacl)# permit any
FESX424 Router(config-std-nacl)# exit
FESX424 Router(config)# int eth 1
FESX424 Router(config-if-e1000-1)# ip access-group Net1 in
```

The commands in this example configure a standard ACL named “Net1”. The entries in this ACL deny packets from three source IP addresses from being forwarded on port 1. Since the implicit action for an ACL is “deny”, the last ACL entry in this ACL permits all packets that are not explicitly denied by the first three ACL entries. For an example of how to configure the same entries in a numbered ACL, see “Configuring Standard Numbered ACLs” on page 12-4.

Notice that the command prompt changes after you enter the ACL type and name. The “std” in the command prompt indicates that you are configuring entries for a standard ACL. For an extended ACL, this part of the command prompt is “ext“. The “nacl” indicates that you are configuring a named ACL.

Configuring Extended Numbered ACLs

This section describes how to configure extended numbered ACLs.

Extended ACLs let you permit or deny packets based on the following information:

- IP protocol
- Source IP address or host name
- Destination IP address or host name
- Source TCP or UDP port (if the IP protocol is TCP or UDP)
- Destination TCP or UDP port (if the IP protocol is TCP or UDP)

The IP protocol can be one of the following well-known names or any IP protocol number from 0 – 255:

- Internet Control Message Protocol (ICMP)
- Internet Group Management Protocol (IGMP)
- Internet Gateway Routing Protocol (IGRP)
- Internet Protocol (IP)
- Open Shortest Path First (OSPF)
- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP)

For TCP and UDP, you also can specify a comparison operator and port name or number. For example, you can configure a policy to block web access to a specific website by denying all TCP port 80 (HTTP) packets from a specified source IP address to the website’s IP address.

Extended Numbered ACL Syntax

[no] access-list <acl-num> deny | permit <ip-protocol> <source-ip> | <hostname> <wildcard> [<operator> <source-tcp/udp-port>] <destination-ip> | <hostname> [<icmp-num> | <icmp-type>] <wildcard> [<tcp/udp comparison operator> <destination-tcp/udp-port>] [dscp-cos-mapping] [dscp-marking <0-63> [802.1p-priority-

marking <0 –7>... | dscp-cos-mapping]] [dscp-matching <0-63>] [log] [precedence <name> | <0 – 7>] [tos <0 – 63> | <name>] [traffic policy <name>]

[no] access-list <acl-num> deny | permit host <ip-protocol> any any

Syntax: [no] ip access-group <acl-num> in

The <acl-num> parameter is the extended access list number. Specify a number from 100 – 199.

The **deny | permit** parameter indicates whether packets that match the policy are dropped or forwarded.

The <ip-protocol> parameter indicates the type of IP packet you are filtering. You can specify a well-known name for any protocol whose number is less than 255. For other protocols, you must enter the number. Enter "?" instead of a protocol to list the well-known names recognized by the CLI.

The <source-ip> | <hostname> parameter specifies the source IP host for the policy. If you want the policy to match on all source addresses, enter **any**.

The <wildcard> parameter specifies the portion of the source IP host address to match against. The <wildcard> is a four-part value in dotted-decimal notation (IP address format) consisting of ones and zeros. Zeros in the mask mean the packet's source address must match the <source-ip>. Ones mean any value matches. For example, the <source-ip> and <wildcard> values 209.157.22.26 0.0.0.255 mean that all hosts in the Class C subnet 209.157.22.x match the policy.

If you prefer to specify the wildcard (mask value) in Classless Interdomain Routing (CIDR) format, you can enter a forward slash after the IP address, then enter the number of significant bits in the mask. For example, you can enter the CIDR equivalent of "209.157.22.26 0.0.0.255" as "209.157.22.26/24". The CLI automatically converts the CIDR number into the appropriate ACL mask (where zeros instead of ones are the significant bits) and changes the non-significant portion of the IP address into zeros. For example, if you specify 209.157.22.26/24 or 209.157.22.26 0.0.0.255, then save the changes to the startup-config file, the value appears as 209.157.22.0/24 (if you have enabled display of subnet lengths) or 209.157.22.0 0.0.0.255 in the startup-config file.

If you enable the software to display IP subnet masks in CIDR format, the mask is saved in the file in "/<mask-bits>" format. To enable the software to display the CIDR masks, enter the **ip show-subnet-length** command at the global CONFIG level of the CLI. You can use the CIDR format to configure the ACL entry regardless of whether the software is configured to display the masks in CIDR format.

NOTE: If you use the CIDR format, the ACL entries appear in this format in the running-config and startup-config files, but are shown with subnet mask in the display produced by the **show ip access-list** command.

The <destination-ip> | <hostname> parameter specifies the destination IP host for the policy. If you want the policy to match on all destination addresses, enter **any**.

The <icmp-type> | <icmp-num> parameter specifies the ICMP protocol type.

- This parameter applies only if you specified **icmp** as the <ip-protocol> value.
- If you use this parameter, the ACL entry is sent to the CPU for processing.
- If you do not specify a message type, the ACL applies to all types of ICMP messages.

The <icmp-num> parameter can be a value from 0 – 255.

The <icmp-type> parameter can have one of the following values, depending on the software version the device is running:

- any-icmp-type
- echo
- echo-reply
- information-request
- log
- mask-reply

- mask-request
- parameter-problem
- redirect
- source-quench
- time-exceeded
- timestamp-reply
- timestamp-request
- traffic policy
- unreachable
- <num>

The <tcp/udp comparison operator> parameter specifies a comparison operator for the TCP or UDP port number. This parameter applies only when you specify **tcp** or **udp** as the IP protocol. For example, if you are configuring an entry for HTTP, specify **tcp eq http**. You can enter one of the following operators:

- **eq** – The policy applies to the TCP or UDP port name or number you enter after **eq**.
- **established** – This operator applies only to TCP packets. If you use this operator, the policy applies to TCP packets that have the ACK (Acknowledgment) or RST (Reset) bits set on (set to "1") in the Control Bits field of the TCP packet header. Thus, the policy applies only to established TCP sessions, not to new sessions. See Section 3.1, "Header Format", in RFC 793 for information about this field.

  NOTE: This operator applies only to destination TCP ports, not source TCP ports.

- **gt** – The policy applies to TCP or UDP port numbers greater than the port number or the numeric equivalent of the port name you enter after **gt**.
- **lt** – The policy applies to TCP or UDP port numbers that are less than the port number or the numeric equivalent of the port name you enter after **lt**.
- **neq** – The policy applies to all TCP or UDP port numbers except the port number or port name you enter after **neq**.
- **range** – The policy applies to all TCP or UDP port numbers that are between the first TCP or UDP port name or number and the second one you enter following the **range** parameter. The range includes the port names or numbers you enter. For example, to apply the policy to all ports between and including 23 (Telnet) and 53 (DNS), enter the following: **range 23 53**. The first port number in the range must be lower than the last number in the range.

The <tcp/udp-port> parameter specifies the TCP or UDP port number or well-known name. You can specify a well-known name for any application port whose number is less than 1024. For other application ports, you must enter the number. Enter "?" instead of a port to list the well-known names recognized by the CLI.

The **in** parameter specifies that the ACL applies to incoming traffic on the interface to which you apply the ACL. You can apply the ACL to an Ethernet port or a virtual interface.

NOTE: If the ACL is for a virtual routing interface, you also can specify a subset of ports within the VLAN containing that interface when assigning an ACL to the interface. See "Configuring Standard Numbered ACLs" on page 12-4.

The **precedence** <name> | <num> parameter of the **ip access-list** command specifies the IP precedence. The precedence option for of an IP packet is set in a three-bit field following the four-bit header-length field of the packet's header. You can specify one of the following:

- **critical** or **5** – The ACL matches packets that have the critical precedence. If you specify the option number instead of the name, specify number 5.
- **flash** or **3** – The ACL matches packets that have the flash precedence. If you specify the option number

instead of the name, specify number 3.

- **flash-override** or **4** – The ACL matches packets that have the flash override precedence. If you specify the option number instead of the name, specify number 4.
- **immediate** or **2** – The ACL matches packets that have the immediate precedence. If you specify the option number instead of the name, specify number 2.
- **internet** or **6** – The ACL matches packets that have the internetwork control precedence. If you specify the option number instead of the name, specify number 6.
- **network** or **7** – The ACL matches packets that have the network control precedence. If you specify the option number instead of the name, specify number 7.
- **priority** or **1** – The ACL matches packets that have the priority precedence. If you specify the option number instead of the name, specify number 1.
- **routine** or **0** – The ACL matches packets that have the routine precedence. If you specify the option number instead of the name, specify number 0.

The **tos** <name> | <num> parameter of the **ip access-list** command specifies the IP ToS. You can specify one of the following:

- **max-reliability** or **2** – The ACL matches packets that have the maximum reliability ToS. The decimal value for this option is 2.
- **max-throughput** or **4** – The ACL matches packets that have the maximum throughput ToS. The decimal value for this option is 4.
- **min-delay** or **8** – The ACL matches packets that have the minimum delay ToS. The decimal value for this option is 8.
- **min-monetary-cost** or **1** – The ACL matches packets that have the minimum monetary cost ToS. The decimal value for this option is 1.

NOTE: This value is not supported on 10 Gigabit Ethernet modules.

- **normal** or **0** – The ACL matches packets that have the normal ToS. The decimal value for this option is 0.
- <num> – A number from 0 – 15 that is the sum of the numeric values of the options you want. The ToS field is a four-bit field following the Precedence field in the IP header. You can specify one or more of the following. To select more than one option, enter the decimal value that is equivalent to the sum of the numeric values of all the ToS options you want to select. For example, to select the **max-reliability** and **min-delay** options, enter number 10. To select all options, select 15.

The **dscp-cos-mapping** option maps the DSCP value in incoming packets to a hardware table that provides mapping of each of the 0 – 63 DSCP values, and distributes them among eight traffic classes (internal priorities) and eight 802.1p priorities.

NOTE: The **dscp-cos-mapping** option overrides port-based priority settings.

The **dscp-marking** option enables you to configure an ACL that marks matching packets with a specified DSCP value Enter a value from 0 – 63. See “Using an IP ACL to Mark DSCP Values (DSCP Marking)” on page 12-23.

The **dscp-matching** option matches on the packet’s DSCP value. Enter a value from 0 – 63. This option does not change the packet’s forwarding priority through the device or mark the packet. See “DSCP Matching” on page 12-24.

The **log** parameter enables SNMP traps and Syslog messages for packets denied by the ACL.

You can enable logging on ACLs and filters that support logging even when the ACLs and filters are already in use. To do so, re-enter the ACL or filter command and add the **log** parameter to the end of the ACL or filter. The software replaces the ACL or filter command with the new one. The new ACL or filter, with logging enabled, takes effect immediately.

The **traffic-policy** option enables the device to rate limit inbound traffic and to count the packets and bytes per packet to which ACL permit or deny clauses are applied. For configuration procedures and examples, see the chapter “Traffic Policies” on page 15-1.

Configuration Examples for Extended Numbered ACLs

To configure an extended access list that blocks all Telnet traffic received on port 1/1 from IP host 209.157.22.26, enter the following commands.

```
FESX424 Router(config)# access-list 101 deny tcp host 209.157.22.26 any eq telnet
log
FESX424 Router(config)# access-list 101 permit ip any any
FESX424 Router(config)# int eth 1
FESX424 Router(config-if-e1000-1)# ip access-group 101 in
FESX424 Router(config)# write memory
```

Here is another example of commands for configuring an extended ACL and applying it to an interface. These examples show many of the syntax choices. Notice that some of the entries are configured to generate log entries while other entries are not thus configured.

```
FESX424 Router(config)# access-list 102 perm icmp 209.157.22.0/24 209.157.21.0/24
FESX424 Router(config)# access-list 102 deny igmp host rkwong 209.157.21.0/24 log
FESX424 Router(config)# access-list 102 deny igrp 209.157.21.0/24 host rkwong log
FESX424 Router(config)# access-list 102 deny ip host 209.157.21.100 host
209.157.22.1 log
FESX424 Router(config)# access-list 102 deny ospf any any log
FESX424 Router(config)# access-list 102 permit ip any any
```

The first entry permits ICMP traffic from hosts in the 209.157.22.*x* network to hosts in the 209.157.21.*x* network.

The second entry denies IGMP traffic from the host device named “rkwong” to the 209.157.21.*x* network.

The third entry denies IGRP traffic from the 209.157.21.*x* network to the host device named “rkwong”.

The fourth entry denies all IP traffic from host 209.157.21.100to host 209.157.22.1 and generates Syslog entries for packets that are denied by this entry.

The fifth entry denies all OSPF traffic and generates Syslog entries for denied traffic.

The sixth entry permits all packets that are not explicitly denied by the other entries. Without this entry, the ACL would deny all incoming or outgoing IP traffic on the ports to which you assign the ACL.

The following commands apply ACL 102 to the incoming traffic on port 1/2 and to the incoming traffic on port 4/3.

```
FastIron SuperX Router(config)# int eth 1/2
FastIron SuperX Router(config-if-1/2)# ip access-group 102 in
FastIron SuperX Router(config-if-1/2)# exit
FastIron SuperX Router(config)# int eth 4/3
FastIron SuperX Router(config-if-4/3)# ip access-group 102 in
FastIron SuperX Router(config)# write memory
```


Here is another example of an extended ACL.

```
FastIron SuperX Router(config)# access-list 103 deny tcp 209.157.21.0/24
209.157.22.0/24
FastIron SuperX Router(config)# access-list 103 deny tcp 209.157.21.0/24 eq ftp
209.157.22.0/24
FastIron SuperX Router(config)# access-list 103 deny tcp 209.157.21.0/24
209.157.22.0/24 lt telnet neq 5
FastIron SuperX Router(config)# access-list 103 deny udp any range 5 6
209.157.22.0/24 range 7 8
FastIron SuperX Router(config)# access-list 103 permit ip any any
```

The first entry in this ACL denies TCP traffic from the 209.157.21.*x* network to the 209.157.22.x network.

The second entry denies all FTP traffic from the 209.157.21.*x* network to the 209.157.22.x network.

The third entry denies TCP traffic from the 209.157.21.*x* network to the 209.157.22.*x* network, if the TCP port number of the traffic is less than the well-known TCP port number for Telnet (23), and if the TCP port is not equal to 5. Thus, TCP packets whose TCP port numbers are 5 or are greater than 23 are allowed.

The fourth entry denies UDP packets from any source to the 209.157.22.*x* network, if the UDP port number from the source network is 5 or 6 and the destination UDP port is 7 or 8.

The fifth entry permits all packets that are not explicitly denied by the other entries. Without this entry, the ACL would deny all incoming or outgoing IP traffic on the ports to which you assign the ACL.

The following commands apply ACL 103 to the incoming traffic on ports 2/1 and 2/2.

```
FastIron SuperX Router(config)# int eth 2/1
FastIron SuperX Router(config-if-2/1)# ip access-group 103 in
FastIron SuperX Router(config-if-2/1)# exit
FastIron SuperX Router(config)# int eth 2/2
FastIron SuperX Router(config-if-2/2)# ip access-group 103 in
FastIron SuperX Router(config)# write memory
```

Configuring Extended Named ACLs

The commands for configuring named ACL entries are different from the commands for configuring numbered ACL entries. The command to configure a numbered ACL is **access-list**. The command for configuring a named ACL is **ip access-list**. In addition, when you configure a numbered ACL entry, you specify all the command parameters on the same command. When you configure a named ACL, you specify the ACL type (standard or extended) and the ACL number with one command, which places you in the configuration level for that ACL. Once you enter the configuration level for the ACL, the command syntax is the same as the syntax for numbered ACLs.

Extended ACLs let you permit or deny packets based on the following information:

- IP protocol
- Source IP address or host name
- Destination IP address or host name
- Source TCP or UDP port (if the IP protocol is TCP or UDP)
- Destination TCP or UDP port (if the IP protocol is TCP or UDP)

The IP protocol can be one of the following well-known names or any IP protocol number from 0 – 255:

- Internet Control Message Protocol (ICMP)
- Internet Group Management Protocol (IGMP)

- Internet Gateway Routing Protocol (IGRP)
- Internet Protocol (IP)
- Open Shortest Path First (OSPF)
- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP)

For TCP and UDP, you also can specify a comparison operator and port name or number. For example, you can configure a policy to block web access to a specific website by denying all TCP port 80 (HTTP) packets from a specified source IP address to the website's IP address.

Extended Named ACL Syntax

Syntax: [no] ip access-list extended <acl-name>

deny | permit <ip-protocol> <source-ip> | <hostname> <wildcard> [<operator> <source-tcp/udp-port>] <destination-ip> | <hostname> [<icmp-num> | <icmp-type>] <wildcard> [<tcp/udp comparison operator> <destination-tcp/udp-port>] [dscp-cos-mapping] [dscp-marking <0-63> [802.1p-priority-marking <0 –7>... | dscp-cos-mapping]] [dscp-matching <0-63>] [log] [precedence <name> | <0 – 7>] [tos <0 – 63> | <name>] [traffic policy <name>]

Syntax: [no] access-list <num> deny | permit host <ip-protocol> any any

Syntax: [no] ip access-group <num> in

The <acl-name> parameter is the access list name. You can specify a string of up to 256 alphanumeric characters. You can use blanks in the ACL name if you enclose the name in quotation marks (for example, “ACL for Net1”).

The **deny | permit** parameter indicates whether packets that match the policy are dropped or forwarded.

The <ip-protocol> parameter indicates the type of IP packet you are filtering. You can specify a well-known name for any protocol whose number is less than 255. For other protocols, you must enter the number. Enter “?” instead of a protocol to list the well-known names recognized by the CLI.

The <source-ip> | <hostname> parameter specifies the source IP host for the policy. If you want the policy to match on all source addresses, enter **any**.

The <wildcard> parameter specifies the portion of the source IP host address to match against. The <wildcard> is a four-part value in dotted-decimal notation (IP address format) consisting of ones and zeros. Zeros in the mask mean the packet’s source address must match the <source-ip>. Ones mean any value matches. For example, the <source-ip> and <wildcard> values 209.157.22.26 0.0.0.255 mean that all hosts in the Class C subnet 209.157.22.x match the policy.

If you prefer to specify the wildcard (mask value) in Classless Interdomain Routing (CIDR) format, you can enter a forward slash after the IP address, then enter the number of significant bits in the mask. For example, you can enter the CIDR equivalent of “209.157.22.26 0.0.0.255” as “209.157.22.26/24”. The CLI automatically converts the CIDR number into the appropriate ACL mask (where zeros instead of ones are the significant bits) and changes the non-significant portion of the IP address into zeros. For example, if you specify 209.157.22.26/24 or 209.157.22.26 0.0.0.255, then save the changes to the startup-config file, the value appears as 209.157.22.0/24 (if you have enabled display of subnet lengths) or 209.157.22.0 0.0.0.255 in the startup-config file.

If you enable the software to display IP subnet masks in CIDR format, the mask is saved in the file in “/<mask-bits>” format. To enable the software to display the CIDR masks, enter the **ip show-subnet-length** command at the global CONFIG level of the CLI. You can use the CIDR format to configure the ACL entry regardless of whether the software is configured to display the masks in CIDR format.

NOTE: If you use the CIDR format, the ACL entries appear in this format in the running-config and startup-config files, but are shown with subnet mask in the display produced by the **show ip access-list** command.

The <destination-ip> | <hostname> parameter specifies the destination IP host for the policy. If you want the policy to match on all destination addresses, enter **any**.

The <icmp-type> | <icmp-num> parameter specifies the ICMP protocol type.

- This parameter applies only if you specified **icmp** as the <ip-protocol> value.
- If you use this parameter, the ACL entry is sent to the CPU for processing.
- If you do not specify a message type, the ACL applies to all types of ICMP messages.

The <icmp-num> parameter can be a value from 0 – 255.

The <icmp-type> parameter can have one of the following values, depending on the software version the device is running:

- any-icmp-type

- echo
- echo-reply
- information-request
- log
- mask-reply
- mask-request
- parameter-problem
- redirect
- source-quench
- time-exceeded
- timestamp-reply
- timestamp-request
- traffic policy
- unreachable
- <num>

The <tcp/udp comparison operator> parameter specifies a comparison operator for the TCP or UDP port number. This parameter applies only when you specify **tcp** or **udp** as the IP protocol. For example, if you are configuring an entry for HTTP, specify **tcp eq http**. You can enter one of the following operators:

- **eq** – The policy applies to the TCP or UDP port name or number you enter after **eq**.
- **established** – This operator applies only to TCP packets. If you use this operator, the policy applies to TCP packets that have the ACK (Acknowledgment) or RST (Reset) bits set on (set to "1") in the Control Bits field of the TCP packet header. Thus, the policy applies only to established TCP sessions, not to new sessions. See Section 3.1, "Header Format", in RFC 793 for information about this field.

  NOTE: This operator applies only to destination TCP ports, not source TCP ports.

- **gt** – The policy applies to TCP or UDP port numbers greater than the port number or the numeric equivalent of the port name you enter after **gt**.
- **lt** – The policy applies to TCP or UDP port numbers that are less than the port number or the numeric equivalent of the port name you enter after **lt**.
- **neq** – The policy applies to all TCP or UDP port numbers except the port number or port name you enter after **neq**.
- **range** – The policy applies to all TCP or UDP port numbers that are between the first TCP or UDP port name or number and the second one you enter following the **range** parameter. The range includes the port names or numbers you enter. For example, to apply the policy to all ports between and including 23 (Telnet) and 53 (DNS), enter the following: **range 23 53**. The first port number in the range must be lower than the last number in the range.

The <tcp/udp-port> parameter specifies the TCP or UDP port number or well-known name. You can specify a well-known name for any application port whose number is less than 1024. For other application ports, you must enter the number. Enter "?" instead of a port to list the well-known names recognized by the CLI.

The **in** parameter specifies that the ACL applies to incoming traffic on the interface to which you apply the ACL. You can apply the ACL to an Ethernet port or a virtual interface.

NOTE: If the ACL is for a virtual routing interface, you also can specify a subset of ports within the VLAN containing that interface when assigning an ACL to the interface. See "Configuring Standard Numbered ACLs" on page 12-4.

The **precedence** <name> | <num> parameter of the **ip access-list** command specifies the IP precedence. The precedence option for of an IP packet is set in a three-bit field following the four-bit header-length field of the packet's header. You can specify one of the following:

- **critical** or **5** – The ACL matches packets that have the critical precedence. If you specify the option number instead of the name, specify number 5.
- **flash** or **3** – The ACL matches packets that have the flash precedence. If you specify the option number instead of the name, specify number 3.
- **flash-override** or **4** – The ACL matches packets that have the flash override precedence. If you specify the option number instead of the name, specify number 4.
- **immediate** or **2** – The ACL matches packets that have the immediate precedence. If you specify the option number instead of the name, specify number 2.
- **internet** or **6** – The ACL matches packets that have the internetwork control precedence. If you specify the option number instead of the name, specify number 6.
- **network** or **7** – The ACL matches packets that have the network control precedence. If you specify the option number instead of the name, specify number 7.
- **priority** or **1** – The ACL matches packets that have the priority precedence. If you specify the option number instead of the name, specify number 1.
- **routine** or **0** – The ACL matches packets that have the routine precedence. If you specify the option number instead of the name, specify number 0.

The **tos** <name> | <num> parameter of the **ip access-list** command specifies the IP ToS. You can specify one of the following:

- **max-reliability** or **2** – The ACL matches packets that have the maximum reliability ToS. The decimal value for this option is 2.
- **max-throughput** or **4** – The ACL matches packets that have the maximum throughput ToS. The decimal value for this option is 4.
- **min-delay** or **8** – The ACL matches packets that have the minimum delay ToS. The decimal value for this option is 8.
- **min-monetary-cost** or **1** – The ACL matches packets that have the minimum monetary cost ToS. The decimal value for this option is 1.

  NOTE: This value is not supported on 10 Gigabit Ethernet modules.

- **normal** or **0** – The ACL matches packets that have the normal ToS. The decimal value for this option is 0.
- <num> – A number from 0 – 15 that is the sum of the numeric values of the options you want. The ToS field is a four-bit field following the Precedence field in the IP header. You can specify one or more of the following. To select more than one option, enter the decimal value that is equivalent to the sum of the numeric values of all the ToS options you want to select. For example, to select the **max-reliability** and **min-delay** options, enter number 10. To select all options, select 15.

The **dscp-cos-mapping** option maps the DSCP value in incoming packets to a hardware table that provides mapping of each of the 0 – 63 DSCP values, and distributes them among eight traffic classes (internal priorities) and eight 802.1p priorities.

NOTE: The **dscp-cos-mapping** option overrides port-based priority settings.

The **dscp-marking** option enables you to configure an ACL that marks matching packets with a specified DSCP value Enter a value from 0 – 63. See “Using an IP ACL to Mark DSCP Values (DSCP Marking)” on page 12-23.

The **dscp-matching** option matches on the packet’s DSCP value. Enter a value from 0 – 63. This option does not change the packet’s forwarding priority through the device or mark the packet. See “DSCP Matching” on page 12-24.

The **log** parameter enables SNMP traps and Syslog messages for packets denied by the ACL.

You can enable logging on ACLs and filters that support logging even when the ACLs and filters are already in use. To do so, re-enter the ACL or filter command and add the **log** parameter to the end of the ACL or filter. The software replaces the ACL or filter command with the new one. The new ACL or filter, with logging enabled, takes effect immediately.

The **traffic-policy** option enables the device to rate limit inbound traffic and to count the packets and bytes per packet to which ACL permit or deny clauses are applied. For configuration procedures and examples, see the chapter “Traffic Policies” on page 15-1.

Configuration Example for Extended Named ACLs

To configure an extended named ACL, enter commands such as the following.

```
FastIron SuperX Router(config)# ip access-list extended “block Telnet”
FastIron SuperX Router(config-ext-nacl)# deny tcp host 209.157.22.26 any eq telnet
log
FastIron SuperX Router(config-ext-nacl)# permit ip any any
FastIron SuperX Router(config-ext-nacl)# exit
FastIron SuperX Router(config)# int eth 1/1
FastIron SuperX Router(config-if-1/1)# ip access-group “block Telnet” in
```

The options at the ACL configuration level and the syntax for the **ip access-group** command are the same for numbered and named ACLs and are described in “Configuring Extended Numbered ACLs” on page 12-8 and “Configuring Extended Named ACLs” on page 12-13.

Adding a Comment to an ACL Entry

You can optionally add comment text to describe entries in an ACL. The comment text appears in the output of **show** commands that display ACL information.

For example, the following commands add comments to entries to a numbered ACL, ACL 100:

```
FESX424 Router(config)# access-list 100 remark The following line permits TCP
packets
FESX424 Router(config)# access-list 100 permit tcp 192.168.4.40/24 2.2.2.2/24

FESX424 Router(config)# access-list 100 remark The following permits UDP packets
FESX424 Router(config)# access-list 100 permit udp 192.168.2.52/24 2.2.2.2/24
FESX424 Router(config)# access-list 100 deny ip any any
```


If the ACL is a named ACL, (for example, you entered TCP/UDP instead of 100), enter the following commands:

```
FESX424 Router(config)# access-list TCP/UDP remark The following line permits TCP
packets
FESX424 Router(config)# access-list TCP/UDP permit tcp 192.168.4.40/24 2.2.2.2/24
FESX424 Router(config)# access-list TCP/UDP remark The following permits UDP
packets
FESX424 Router(config)# access-list TCP/UDP permit udp 192.168.2.52/24 2.2.2.2/24
FESX424 Router(config)# access-list TCP/UDP deny ip any any
```

Syntax: [no] access-list <acl-num> | <acl-name> remark <comment-text>

Enter the number of the ACL for <acl-num>. You can add a comment to a named ACL by entering the ACL's name for <acl-name>.

The <comment-text> can be up to 128 characters in length. The comment must be entered separately from the actual ACL entry; that is, you cannot enter the ACL entry and the ACL comment with the same **access-list** command. Also, in order for the remark to be displayed correctly in the output of **show** commands, the comment must be entered immediately before the ACL entry it describes.

You can use the **show running-config** or **show access-list** commands to display the ACL and comments

The following shows an example of a numbered ACL with a comment text in a show running-config display:

```
FESX424 Router# show running-config
…
access-list 100 remark The following line permits TCP packets
access-list 100 permit tcp 192.168.4.40/24 2.2.2.2/24
access-list 100 remark The following line permits UDP packets
access-list 100 permit udp 192.168.2.52/24 2.2.2.2/24
access-list 100 deny ip any any
```

The following shows the comment text for the ACL named TCP/UDP in a show running-config display:

```
FESX424 Router# show running-config ...

access-list TCP/UDP remark The following line permits TCP packets
access-list TCP/UDP permit tcp 192.168.4.40/24 2.2.2.2/24
access-list TCP/UDP remark The following line permits UDP packets
access-list TCP/UDP permit udp 192.168.2.52/24 2.2.2.2/24
access-list TCP/UDP deny ip any any
```

Syntax: show running-config

The following example show the comment text for a numbered ACL in a show access-list display:

```
FESX424 Router# show access-list 100
IP access list rate-limit 100 aaaa.bbbb.cccc

Extended IP access list 100 (Total flows: N/A, Total packets: N/A)
  ACL Comments:  The following line permits TCP packets
  permit tcp 0.0.0.40 255.255.255.0 0.0.0.2 255.255.255.0 (Flows: N/A, Packets: N/A)
  ACL Comments:  The following line permits UDP packets
  permit udp 0.0.0.52 255.255.255.0 0.0.0.2 255.255.255.0 (Flows: N/A, Packets: N/A)
  deny ip any any (Flows: N/A, Packets: N/A)
```

The next example shows the comment text for a named ACL in a show access-list display:

```
FESX424 Router# show access-list TCP/UDP

IP access list rate-limit 100 aaaa.bbbb.cccc

Extended IP access list TCP/UDP (Total flows: N/A, Total packets: N/A)
ACL Comments:  The following line permits TCP packets
permit tcp 0.0.0.40 255.255.255.0 0.0.0.2 255.255.255.0 (Flows: N/A, Packets: N/A)
ACL Comments:  The following line permits UDP packets
permit udp 0.0.0.52 255.255.255.0 0.0.0.2 255.255.255.0 (Flows: N/A, Packets: N/A)
deny ip any any (Flows: N/A, Packets: N/A)
```

Syntax: show access-list <acl-num> | <acl-name> | all

Enabling Strict Control of ACL Filtering of Fragmented Packets

The default processing of fragments by hardware-based ACLs is as follows:

- The first fragment of a packet is permitted or denied using the ACLs. The first fragment is handled the same way as non-fragmented packets, since the first fragment contains the Layer 4 source and destination application port numbers. The device uses the Layer 4 CAM entry if one is programmed, or applies the interface's ACL entries to the packet and permits or denies the packet according to the first matching ACL.
- For other fragments of the same packet, they are subject to a rule only if there is no Layer 4 information in the rule or in any preceding rules.

The fragments are forwarded even if the first fragment, which contains the Layer 4 information, was denied. Generally, denying the first fragment of a packet is sufficient, since a transaction cannot be completed without the entire packet.

For tighter control, you can configure the port to drop all packet fragments. To do so, enter commands such as the following:

```
FastIron SuperX Router(config)# interface ethernet 1/1
FastIron SuperX Router(config-if-1/1)# ip access-group frag deny
```

This option begins dropping all fragments received by the port as soon as you enter the command. This option is especially useful if the port is receiving an unusually high rate of fragments, which can indicate a hacker attack.

Syntax: [no] ip access-group frag deny

Enabling ACL Filtering Based on VLAN Membership or VE Port Membership

Starting with release 02.3.03, you can apply an inbound ACL to specific VLAN members on a port (Layer 2 devices only) or to specific ports on a virtual interface (VE) (Layer 3 Devices only).

By default, this feature support is disabled. To enable it, enter the following commands at the Global CONFIG level of the CLI:

```
FESX424 Switch (config)# enable acl-per-port-per-vlan
FESX424 Switch (config)# write memory
FESX424 Switch (config)# exit
FESX424 Switch# reload
```

After entering the above commands, you can:

- Apply an ACL to specific VLAN members on a port – see Page 12-21
- Apply an ACL to a subset of ports on a VE – see Page 12-21

NOTE: You must save the configuration and reload the software to place the change into effect.

Syntax: [no] enable acl-per-port-per-vlan

Enter the **no** form of the command to disable this feature.

Applying an ACL to Specific VLAN Members on a Port (Layer 2 Devices Only)

When you bind an ACL to a port, the port filters all inbound traffic on the port. However, on a tagged port, there may be a need to treat packets for one VLAN differently from packets for another VLAN. Starting with release 02.3.03, you can configure a tagged port on a Layer 2 device to filter packets based on the packets' VLAN membership.

NOTE: Before you can bind an ACL to specific VLAN members on a port, you must first enable support for this feature. If this feature is not already enabled on your device, enable it as instructed in the section "Enabling ACL Filtering Based on VLAN Membership or VE Port Membership" on page 12-20.

To apply an ACL to a specific VLAN on a port, enter commands such as the following on a tagged port:

```
FESX424 Switch(config)# vlan 12 name vlan12
FESX424 Switch(config-vlan-12)# untag ethernet 5 to 8
FESX424 Switch(config-vlan-12)# tag ethernet 23 to 24
FESX424 Switch(config-vlan-12)#exit
FESX424 Switch(config)# access-list 10 deny host 209.157.22.26 log
FESX424 Switch(config)# access-list 10 deny 209.157.29.12 log
FESX424 Switch(config)# access-list 10 deny host IPHost1 log
FESX424 Switch(config)# access-list 10 permit
FESX424 Switch(config)# int e 23
FESX424 Switch(config-if-e1000-23))# per-vlan 12
FESX424 Switch(config-if-e1000-23-vlan-12))#ip access-group 10 in
```

The commands in this example configure port-based VLAN 12, and add ports e 5 – 8 as untagged ports and ports e 23 – 24 as tagged ports to the VLAN. The commands following the VLAN configuration commands configure ACL 10. Finally, the last three commands apply ACL 10 on VLAN 12 for which port e 23 is a member.

Syntax: per-vlan <VLAN ID>

Syntax: [no] ip access-group <ACL ID>

The <VLAN ID> parameter specifies the VLAN name or number to which you will bind the ACL.

The <ACL ID> parameter is the access list name or number.

Applying an ACL to a Subset of Ports on a Virtual Interface (Layer 3 Devices Only)

You can apply an ACL to a virtual routing interface. The virtual interface is used for routing between VLANs and contains all the ports within the VLAN. The ACL applies to all the ports on the virtual routing interface. Starting with release 02.3.03, you also can specify a subset of ports within the VLAN containing a specified virtual interface when assigning an ACL to that virtual interface.

Use this feature when you do not want the ACLs to apply to all the ports in the virtual interface's VLAN or when you want to streamline ACL performance for the VLAN.

NOTE: Before you can bind an ACL to specific ports on a virtual interface, you must first enable support for this feature. If this feature is not already enabled on your device, enable it as instructed in the section "Enabling ACL Filtering Based on VLAN Membership or VE Port Membership" on page 12-20.

To apply an ACL to a subset of ports within a virtual interface, enter commands such as the following:

```
FastIron SuperX Router(config)# vlan 10 name IP-subnet-vlan
FastIron SuperX Router(config-vlan-10)# untag ethernet 1/1 to 2/12
```

```
FastIron SuperX Router(config-vlan-10)# router-interface ve 1
FastIron SuperX Router(config-vlan-10)# exit
FastIron SuperX Router(config)# access-list 1 deny host 209.157.22.26 log
FastIron SuperX Router(config)# access-list 1 deny 209.157.29.12 log
FastIron SuperX Router(config)# access-list 1 deny host IPHost1 log
FastIron SuperX Router(config)# access-list 1 permit any
FastIron SuperX Router(config)# interface ve 1
FastIron SuperX Router(config-vif-1)# ip access-group 1 in ethernet 1/1 ethernet 1/
3 ethernet 2/1 to 2/4
```

The commands in this example configure port-based VLAN 10, add ports 1/1 – 2/12 to the VLAN, and add virtual routing interface 1 to the VLAN. The commands following the VLAN configuration commands configure ACL 1. Finally, the last two commands apply ACL 1 to a subset of the ports associated with virtual interface 1.

Syntax: [no] ip access-group <ACL ID> in ethernet <slotnum>/<portnum> [to <slotnum>/<portnum>]

The <ACL ID> parameter is the access list name or number.

The <slotnum> parameter applies on chassis devices only. It does not apply on FESX devices.

Filtering on IP Precedence and ToS Values

To configure an extended IP ACL that matches based on IP precedence, enter commands such as the following:

```
FESX424 Router(config)# access-list 103 deny tcp 209.157.21.0/24 209.157.22.0/24
precedence internet
FESX424 Router(config)# access-list 103 deny tcp 209.157.21.0/24 eq ftp
209.157.22.0/24 precedence 6
FESX424 Router(config)# access-list 103 permit ip any any
```

The first entry in this ACL denies TCP traffic from the 209.157.21.*x* network to the 209.157.22.x network, if the traffic has the IP precedence option “internet” (equivalent to “6”).

The second entry denies all FTP traffic from the 209.157.21.*x* network to the 209.157.22.x network, if the traffic has the IP precedence value “6” (equivalent to “internet”).

The third entry permits all packets that are not explicitly denied by the other entries. Without this entry, the ACL would deny all incoming or outgoing IP traffic on the ports to which you assign the ACL.

To configure an IP ACL that matches based on ToS, enter commands such as the following:

```
FESX424 Router(config)# access-list 104 deny tcp 209.157.21.0/24 209.157.22.0/24 tos
normal
FESX424 Router(config)# access-list 104 deny tcp 209.157.21.0/24 eq ftp 209.157.22.0/24
tos 13
FESX424 Router(config)# access-list 104 permit ip any any
```

The first entry in this IP ACL denies TCP traffic from the 209.157.21.*x* network to the 209.157.22.x network, if the traffic has the IP ToS option “normal” (equivalent to “0”).

The second entry denies all FTP traffic from the 209.157.21.*x* network to the 209.157.22.x network, if the traffic has the IP precedence value “13” (equivalent to “max-throughput”, “min-delay”, and “min-monetary-cost”).

The third entry permits all packets that are not explicitly denied by the other entries. Without this entry, the ACL would deny all incoming or outgoing IP traffic on the ports to which you assign the ACL.

QoS Options for IP ACLs

Quality of Service (QoS) options enable you to perform QoS for packets that match the ACLs. Using an ACL to perform QoS is an alternative to directly setting the internal forwarding priority based on incoming port, VLAN membership, and so on. (This method is described in "Assigning QoS Priorities to Traffic" on page 13-7.)

The following QoS ACL options are supported:

- **dscp-cos-mapping** – This option is similar to the **dscp-matching** command (described below). This option maps the DSCP value in incoming packets to a hardware table that provides mapping of each of the 0 – 63 DSCP values, and distributes them among eight traffic classes (internal priorities) and eight 802.1p priorities.

  By default, the Foundry device does the *802.1p to CoS* mapping. If you want to change the priority mapping to *DSCP to CoS* mapping, you must enter the following ACL statement:

  ```
  permit ip any any dscp-cos-mapping
  ```

- **dscp-marking** – Marks the DSCP value in the outgoing packet with the value you specify.
  - **internal-priority-marking** and **802.1p-priority-marking** – Supported with the DSCP marking option, these commands assign traffic that matches the ACL to a hardware forwarding queue (**internal-priority-marking**), and re-mark the packets that match the ACL with the 802.1p priority (**802.1p-priority-marking**).
- **dscp-matching** – Matches on the packet's DSCP value. This option does not change the packet's forwarding priority through the device or mark the packet.

Using an ACL to Map the DSCP Value (DSCP CoS Mapping)

The **dscp-cos-mapping** option on the FESX and FSX maps the DSCP value in incoming packets to a hardware table that provides mapping of each of the 0 – 63 DSCP values, and distributes them among eight traffic classes (internal priorities) and eight 802.1p priorities.

NOTE: The **dscp-cos-mapping** option overrides port-based priority settings.

By default, the Foundry device does the *802.1p to CoS* mapping. If you want to change the priority mapping to *DSCP to CoS* mapping, you must enter the following ACL statement:

```
permit ip any any dscp-cos-mapping
```

The complete CLI syntax for this feature is shown in "Configuring Extended Numbered ACLs" on page 12-8 and "Configuring Extended Named ACLs" on page 12-13. The following shows the syntax specific to the DSCP Cos mapping feature.

Syntax: ... [dscp-marking <dscp-value> **dscp-cos-mapping**]

OR

Syntax: ...[dscp-cos-mapping]

Using an IP ACL to Mark DSCP Values (DSCP Marking)

The **dscp-marking** option for extended ACLs allows you to configure an ACL that marks matching packets with a specified DSCP value. You also can use DSCP marking to assign traffic to a specific hardware forwarding queue (see "Using an ACL to Change the Forwarding Queue" on page 12-24).

For example, the following commands configure an ACL that marks all IP packets with DSCP value 5. The ACL is then applied to incoming packets on interface 7. Consequently, all inbound packets on interface 7 are marked with the specified DSCP value.

```
FESX424 Router(config)# access-list 120 permit ip any any dscp-marking 5 dscp-cos-
mapping
FESX424 Router(config)# interface 7
FESX424 Router(config-if-e1000-7)# ip access-group 120 in
```

Syntax: ...**dscp-marking** <dscp-value>

The **dscp-marking** <dscp-value> parameter maps a DSCP value to an internal forwarding priority. The DSCP value can be from 0 – 63.

Using an ACL to Change the Forwarding Queue

The **802.1p-priority-marking** <0 – 7> parameter re-marks the packets of the 802.1Q traffic that match the ACL with this new 802.1p priority, or marks the packets of the non-802.1Q traffic that match the ACL with this 802.1p priority, later at the outgoing 802.1Q interface.

The 802.1p priority mapping is shown in Table 12.2.

The **internal-priority-marking** <0 – 7> parameter assigns traffic that matches the ACL to a specific hardware forwarding queue (qosp0 – qosp7>.

NOTE: The **internal-priority-marking** parameter overrides port-based priority settings.

In addition to changing the internal forwarding priority, if the outgoing interface is an 802.1Q interface, this parameter maps the specified priority to its equivalent 802.1p (CoS) priority and marks the packet with the new 802.1p priority. Table 12.2 lists the default mappings of hardware forwarding queues to 802.1p priorities on the FESX and FSX.

Table 12.2: Default Mapping of Forwarding Queues to 802.1p Priorities

Forwarding Queue	qosp0	qosp1	qosp2	qosp3	qosp4	qosp5	qosp6	qosp7
802.1p	0	1	2	3	4	5	6	7

The complete CLI syntax for 802.1p priority marking and internal priority marking is shown in “Configuring Extended Numbered ACLs” on page 12-8 and “Configuring Extended Named ACLs” on page 12-13. The following shows the syntax specific to these features.

Syntax: ... **dscp-marking** <0 – 63> **802.1p-priority-marking** <0 – 7> **internal-priority-marking** <0 – 7>]

DSCP Matching

The **dscp-matching** option matches on the packet’s DSCP value. This option does not change the packet’s forwarding priority through the device or mark the packet.

To configure an ACL that matches on a packet with DSCP value 29, enter a command such as the following:

```
FESX424 Router(config)# access-list 112 permit ip 1.1.1.0 0.0.0.255 2.2.2.x
0.0.0.255 dscp-matching 29
```

The complete CLI syntax for this feature is shown in “Configuring Extended Numbered ACLs” on page 12-8 and “Configuring Extended Named ACLs” on page 12-13. The following shows the syntax specific to this feature.

Syntax: ...**dscp-matching** <0 – 63>

NOTE: For complete syntax information, see “Extended Numbered ACL Syntax” on page 12-8.

ACL-Based Rate Limiting

Software releases 02.3.03 and later provide support for IP ACL-based rate limiting of inbound traffic. ACL-based rate limiting provides the facility to limit the rate for IP traffic that matches the permit conditions in extended IP ACLs. This feature is available in the Layer 2 and Layer 3 code.

For more details, including configuration procedures, see the chapter “Traffic Policies” on page 15-1.

ACL Counting

Software releases 02.3.03 and later support ***ACL counting***, a mechanism for counting the number of packets and the number of bytes per packet to which ACL filters are applied.

Configuration procedures for ACL counting are in the chapter “Traffic Policies” on page 15-1.

Using ACLs to Control Multicast Features

You can use ACLs to control the following multicast features:

- Limit the number of multicast groups that are covered by a static rendezvous point (RP)
- Control which multicast groups for which candidate RPs sends advertisement messages to bootstrap routers
- Identify which multicast group packets will be forwarded or blocked on an interface

For configuration procedures, see the chapter “Configuring IP Multicast Protocols” on page 19-1

Displaying ACL Information

To display the number of Layer 4 CAM entries used by each ACL, enter the following command:

```
FESX424 Router(config)# show access-list all

Extended IP access list 100 (Total flows: N/A, Total packets: N/A, Total rule cam use: 3)
permit udp host 192.168.2.169 any (Flows: N/A, Packets: N/A, Rule cam use: 1)
permit icmp any any (Flows: N/A, Packets: N/A, Rule cam use: 1)
deny ip any any (Flows: N/A, Packets: N/A, Rule cam use: 1)
```

Syntax: show access-list <acl-num> | <acl-name> | all

The Rule cam use field lists the number of CAM entries used by the ACL or entry. The number of CAM entries listed for the ACL itself is the total of the CAM entries used by the ACL’s entries.

For flow-based ACLs, the Total flows and Flows fields list the number of Layer 4 session table flows in use for the ACL.

The Total packets and Packets fields apply only to flow-based ACLs.

Troubleshooting ACLs

Use the following methods to troubleshoot ACLs:

- To display the number of Layer 4 CAM entries being used by each ACL, enter the **show access-list** <acl-num> | <acl-name> | **all** command. See “Displaying ACL Information” on page 12-25.
- To determine whether the issue is specific to fragmentation, remove the Layer 4 information (TCP or UDP application ports) from the ACL, then reapply the ACL.

If you are using another feature that requires ACLs, either use the same ACL entries for filtering and for the other feature, or change to flow-based ACLs.

Chapter 13
Configuring Quality of Service

Quality of Service (QoS) features are used to prioritize the use of bandwidth in a switch. When QoS features are enabled, traffic is classified as it arrives at the switch, and processed through on the basis of configured priorities. Traffic can be dropped, prioritized for guaranteed delivery, or subject to limited delivery options as configured by a number of different mechanisms.

This chapter describes how QoS is implemented and configured in the FESX, FSX, and FWSX devices. This chapter contains the topics listed in Table 13.1.

Table 13.1: Chapter Contents

Classification

Classification is the process of selecting packets on which to perform QoS, reading the QoS information and assigning a priority to the packets. The classification process assigns a priority to packets as they enter the switch. These priorities can be determined on the basis of information contained within the packet or assigned to

the packet as it arrives at the switch. Once a packet or traffic flow is classified, it is mapped to a forwarding priority queue.

Packets on the FESX, FSX, and FWSX are classified in up to eight traffic classes with values between 0 and 7. Packets with higher priority classifications are given a precedence for forwarding.

Processing of Classified Traffic

The ***trust level*** in effect on an interface determines the type of QoS information the device uses for performing QoS. The Foundry device establishes the trust level based on the configuration of various features and if the traffic is switched or routed. The trust level can be one of the following:

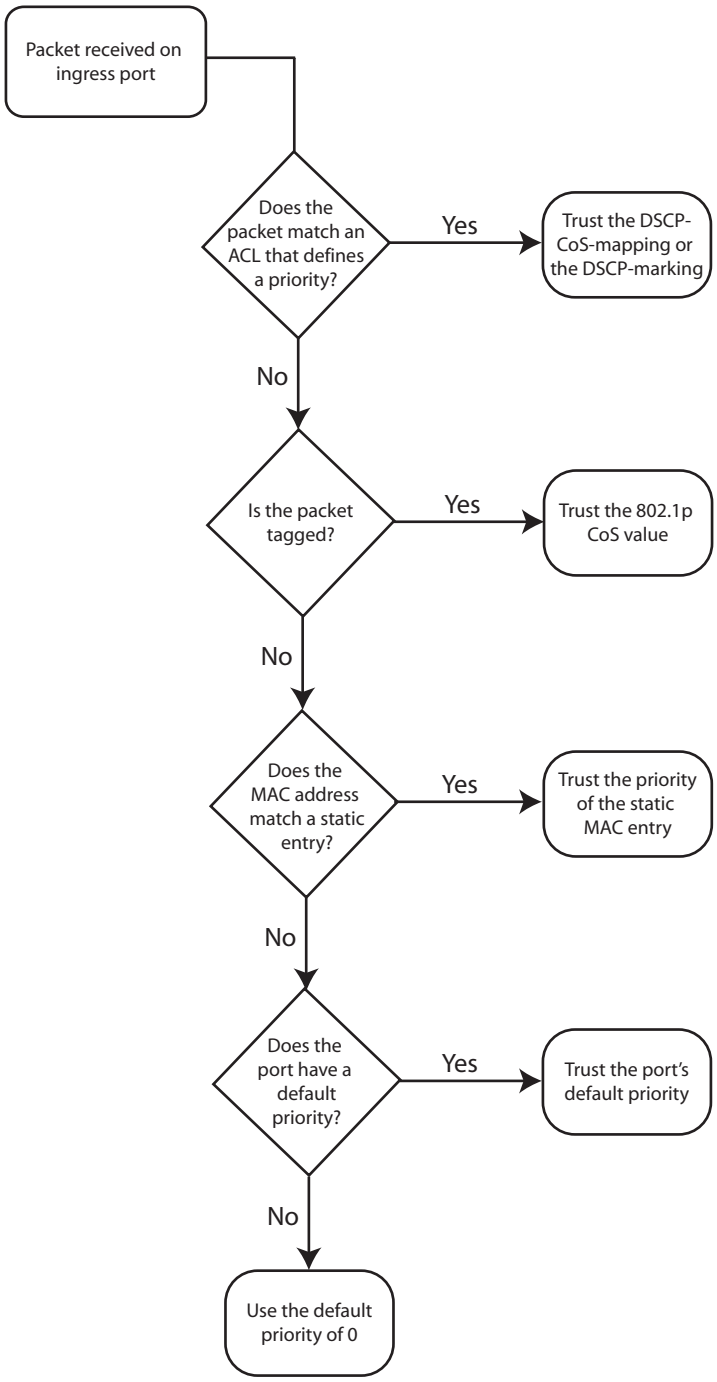
- Ingress port default priority
- Static MAC address
- Layer 2 Class of Service (CoS) value – This is the 802.1p priority value in the Ethernet frame. It can be a value from 0 – 7. The 802.1p priority is also called the Class of Service.
- Layer 3 Differentiated Service codepoint (DSCP) – This is the value in the six most significant bits of the IP packet header's 8-bit DSCP field. It can be a value from 0 – 63. These values are described in RFCs 2472 and 2475. The DSCP value is sometimes called the DiffServ value. The device automatically maps a packet's DSCP value to a hardware forwarding queue. See “Viewing QoS Settings” on page 13-15".
- ACL keyword – An ACL can also prioritize traffic and mark it before sending it along to the next hop. This is described in the ACL chapter in the section “QoS Options for IP ACLs” on page 12-23.

Given the variety of different criteria, there are multiple possibilities for traffic classification within a stream of network traffic. For this reason, the priority of packets must be resolved based on which criteria takes precedence. Precedence follows the scheme illustrated in Figure 13.1

Determining a Packet's Trust Level

Figure 13.1 illustrates how the Foundry device determines a packet's trust level.

Figure 13.1 Determining a Packet's Trust Level



As shown in the figure, the first criteria considered is whether the packet matches on an ACL that defines a priority. If this is not the case and the packet is tagged, the packet is classified with the 802.1p CoS value. If neither of these are true, the packet is next classified based on the static MAC address, ingress port default priority, or the default priority of zero (0).

Once a packet is classified by one of the procedures mentioned, it is mapped to an internal forwarding queue. There are eight queues designated as 0 to 7. The internal forwarding priority maps to one of these eight queues as shown in Table 13.2 through Table 13.5. The mapping between the internal priority and the forwarding queue cannot be changed.

Table 13.2 through Table 13.5 show the default QoS mappings which are used if the trust level for CoS or DSCP is enabled.

Table 13.2: Default QoS Mappings, Columns 0 to 15

DSCP value	0	1	2	3	4	5	6	7	8	9	10	11	12	12	14	15
802.1p (COS) Value	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
DSCP value	0	1	2	3	4	5	6	7	8	9	10	11	12	12	14	15
Internal Forwarding Priority	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1
Forwarding Queue	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1

Table 13.3: Default QoS Mappings, Columns 16 to 31

DSCP value	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
802.1p (COS) Value	2	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3
DSCP value	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
Internal Forwarding Priority	2	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3
Forwarding Queue	2	2	2	2	2	2	2	2	3	3	3	3	3	3	3	3

Table 13.4: Default QoS Mappings, Columns 32 to 47

DSCP value	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
802.1p (COS) Value	4	4	4	4	4	4	4	4	5	5	5	5	5	5	5	5
DSCP value	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
Internal Forwarding Priority	4	4	4	4	4	4	4	4	5	5	5	5	5	5	5	5
Forwarding Queue	4	4	4	4	4	4	4	4	5	5	5	5	5	5	5	5

Table 13.5: Default QoS Mappings, Columns 48 to 63

DSCP value	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
802.1p (COS) Value	6	6	6	6	6	6	6	6	7	7	7	7	7	7	7	7
DSCP value	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
Internal Forwarding Priority	6	6	6	6	6	6	6	6	7	7	7	7	7	7	7	7
Forwarding Queue	6	6	6	6	6	6	6	6	7	7	7	7	7	7	7	7

Mapping between DSCP value and Forwarding Queue cannot be changed. However, mapping between DSCP values and the other properties can be changed as follows:

- **DSCP to Internal Forwarding Priority Mapping** – You can change the mapping between the DSCP value and the Internal Forwarding priority value from the default values shown in Table 13.2 through Table 13.5. This mapping is used for COS marking and determining the internal priority when the trust level is DSCP. See “Changing the DSCP –> Internal Forwarding Priority Mappings” on page 13-10.
- **Internal Forwarding Priority to Forwarding Queue** – You can reassign an internal forwarding priority to a different hardware forwarding queue. See “Changing the Internal Forwarding Priority –> Hardware Forwarding Queue Mappings” on page 13-10.

QoS Queues

Foundry devices support the eight QoS queues (qosp0 – qosp7) listed in Table 13.6.

Table 13.6: QoS Queues

QoS Priority Level	QoS Queue
0	qosp0 (lowest priority queue)
1	qosp1
2	qosp2
3	qosp3
4	qosp4
5	qosp5
6	qosp6
7	qosp7 (highest priority queue)

The queue names listed above are the default names. If desired, you can rename the queues as instructed in “Renaming the Queues” on page 13-12.

Packets are classified and assigned to specific queues based on the criteria shown in Figure 13.1.

Assigning QoS Priorities to Traffic

By default, all traffic is in the best-effort queue (qosp0) and is honored on tagged ports on all FastIron family of switches. You can assign traffic to a higher queue based on the following:

- Incoming port (sometimes called the ingress port)
- Static MAC entry

The following sections describe how to change the priority for each of the items listed above.

Although it is possible for a packet to qualify for an adjusted QoS priority based on more than one of the criteria listed In the section above, the system always gives a packet the highest priority for which it qualifies. Thus, if a packet is entitled to the premium queue because of its IP source and destination addresses, but is entitled only to the high queue because of its incoming port, the system places the packet in the premium queue on the outgoing port.

When you apply a QoS priority to one of the items listed above, you specify a number from 0 – 7. The priority number specifies the IEEE 802.1 equivalent to one of the eight QoS queues on FESX, FSX, and FWSX devices. The numbers correspond to the queues as shown in Table 13.2.

Changing a Port's Priority

To change the QoS priority of port 1 to the premium queue (qosp7), enter the following commands:

```
FESX424 Router(config)# interface ethernet 1
FESX424 Router(config-if-e1000-1)# priority 7
```

The device will assign priority 7 to untagged switched traffic received on port 1.

Syntax: [no] priority <num>

The <num> parameter can be from 0 – 7 and specifies the IEEE 802.1 equivalent to one of eight QoS queues listed in Table 13.2.

Assigning Static MAC Entries to Priority Queues

By default, all MAC entries are in the best effort queue. When you configure a static MAC entry, you can assign the entry to a higher QoS level.

To configure a static MAC entry and assign the entry to the premium queue, enter commands such as the following:

```
FESX424 Router(config)# vlan 9
FESX424 Router(config-vlan-9)# static-mac-address 1145.1163.67FF ethernet 1/1
priority 7
FESX424 Router(config-vlan-9)# write memory
```

Syntax: [no] static-mac-address <mac-addr> ethernet [<slotnum>/]<portnum> [priority <num>] [host-type | router-type | fixed-host]

The <slotnum>/ parameter applies to the FSX only.

The **priority** <num> parameter can be from 0 – 7 and specifies the IEEE 802.1 equivalent to one of the eight QoS queues.

NOTE: The location of the **static-mac-address** command in the CLI depends on whether you configure port-based VLANs on the device. If the device does not have more than one port-based VLAN (VLAN 1, which is the default VLAN that contains all the ports), the **static-mac-address** command is at the global CONFIG level of the CLI. If the device has more than one port-based VLAN, then the **static-mac-address** command is not available at the global CONFIG level. In this case, the command is available at the configuration level for each port-based VLAN.

Marking

Marking is the process of changing the packet's QoS information (the 802.1p and DSCP information in a packet) for the next hop. For example, for traffic coming from a device that does not support DiffServ, you can change the packet's IP Precedence value into a DSCP value before forwarding the packet.

You can mark a packet's Layer 2 CoS value, its Layer 3 DSCP value, or both values. The Layer 2 CoS or DSCP value the device marks in the packet is the same value that results from mapping the packet's QoS value into a Layer 2 CoS or DSCP value.

Marking is optional and is disabled by default. Marking is performed using ACLs. When marking is not used, the device still performs the mappings listed in "Classification" for scheduling the packet, but leaves the packet's QoS values unchanged when the device forwards the packet.

For configuration syntax, rules, and examples of QoS marking, see "QoS Options for IP ACLs" on page 12-23.

Configuring DSCP-Based QoS

FastIron IronWare releases support basic DSCP-based QoS (also called Type of Service (ToS) based QoS) as described in this chapter. However, the FastIron family of switches do not support other advanced DSCP-based QoS features as described in the *Foundry Enterprise Configuration and Management Guide*.

Foundry FastIron IronWare releases also support marking of the DSCP value. FastIron devices can read Layer 3 Quality of Service (QoS) information in an IP packet and select a forwarding queue for the packet based on the information. The software interprets the value in the six most significant bits of the IP packet header's 8-bit ToS field as a Diffserv Control Point (DSCP) value, and maps that value to an internal forwarding priority.

The internal forwarding priorities are mapped to one of the eight forwarding queues (qosp0 – qosp7) on FESX, FSX, and FWSX devices. During a forwarding cycle, the device gives more preference to the higher numbered queues, so that more packets are forwarded from these queues. So for example, queue qosp7 receives the highest preference while queue qosp0, the best-effort queue, receives the lowest preference.

Application Notes

- DSCP-based QoS is not automatically honored for routed and switched traffic. The default is 802.1p to CoS mapping. To honor DSCP-based QoS, you must change the priority mapping to DSCP to CoS mapping. See "Using ACLs to Honor DSCP-based QoS" .

When DSCP marking is enabled on the FESX, FWSX, or FSX, the device changes the contents of the inbound packet's ToS field to match the DSCP-based QoS value. This is different than on the BigIron, which marks the outbound packet's ToS field.

Using ACLs to Honor DSCP-based QoS

FESX, FSX, and FWSX devices require the use of an ACL to honor DSCP-based QoS for routed traffic in the Layer 3 image, or for switched traffic in the Layer 2 image. To enable DSCP-based QoS on these devices, apply an ACL entry such as the following:

```
FESX424 Router(config)# access-list 101 permit ip any any dscp-cos-map
```

Configuring the QoS Mappings

You can optionally change the following QoS mappings:

- DSCP –> internal forwarding priority
- Internal forwarding priority –> hardware forwarding queue

The mappings are globally configurable and apply to all interfaces.

Default DSCP –> Internal Forwarding Priority Mappings

The DSCP values are described in RFCs 2474 and 2475. Table 13.7 list the default mappings of DSCP values to internal forwarding priority values.

Table 13.7: Default DSCP to Internal Forwarding Priority Mappings

Internal Forwarding Priority	DSCP Value
0 (lowest priority queue)	0 – 7
1	8 – 15
2	16 – 23
3	24 – 31
4	32 – 39
5	40 – 47
6	48 – 55
7 (highest priority queue)	56 – 63

Notice that DSCP values range from 0 – 63, whereas the internal forwarding priority values range from 0 – 7. Any DSCP value within a given range is mapped to the same internal forwarding priority value. For example, any DSCP value from 8 – 15 maps to priority 1.

After performing this mapping, the device maps the internal forwarding priority value to one of the hardware forwarding queues.

Table 13.8 list the default mappings of internal forwarding priority values to the hardware forwarding queues.

Table 13.8: Default Mappings of Internal Forwarding Priority Values

Internal Forwarding Priority	Forwarding Queues
0 (lowest priority queue)	qosp0
1	qosp1
2	qosp2
3	qosp3
4	qosp4
5	qosp5
6	qosp6
7 (highest priority queue)	qosp7

You can change the DSCP -> internal forwarding mappings. You also can change the internal forwarding priority -> hardware forwarding queue mappings.

Changing the DSCP –> Internal Forwarding Priority Mappings

To change the DSCP –> internal forwarding priority mappings for all the DSCP ranges, enter commands such as the following at the global CONFIG level of the CLI:

```
FESX424 Router(config)# qos-tos map dscp-priority 0 2 3 4 to 1
FESX424 Router(config)# qos-tos map dscp-priority 8 to 5
FESX424 Router(config)# qos-tos map dscp-priority 16 to 4
FESX424 Router(config)# qos-tos map dscp-priority 24 to 2
FESX424 Router(config)# qos-tos map dscp-priority 32 to 0
FESX424 Router(config)# qos-tos map dscp-priority 40 to 7
FESX424 Router(config)# qos-tos map dscp-priority 48 to 3
FESX424 Router(config)# qos-tos map dscp-priority 56 to 6
FESX424 Router(config)# ip rebind-acl all
```

These commands configure the mappings displayed in the DSCP to forwarding priority portion of the QoS information display. To read this part of the display, select the first part of the DSCP value from the d1 column and select the second part of the DSCP value from the d2 row. For example, to read the DSCP to forwarding priority mapping for DSCP value 24, select 2 from the d1 column and select 4 from the d2 row. The mappings that are changed by the command above are shown below in bold type.

```
FESX424 Router(config-if-e1000-1)# show qos-tos
```

...portions of table omitted for simplicity...

```
DSCP-Priority map: (dscp = d1d2)

     d2|  0   1   2   3   4   5   6   7   8   9
  d1   |
  -----+----------------------------------------
   0   |  1   0   1   1   1   0   0   0   5   1
   1   |  6   1   1   1   1   1   4   2   2   2
   2   |  2   2   2   2   2   3   3   3   3   3
   3   |  3   3   0   4   4   4   4   4   4   4
   4   |  7   5   5   5   5   5   5   5   3   6
   5   |  6   6   6   6   6   6   6   7   7   7
   6   |  7   7   7   7
```

Syntax: [no] qos-tos map dscp-priority <dscp-value> [<dscp-value> ...] to <priority>

The <dscp-value> [<dscp-value> ...] parameter specifies the DSCP value ranges you are remapping. You can specify up to seven DSCP values in the same command, to map to the same forwarding priority. The first command in the example above maps priority 1 to DSCP values 0, 2, 3, and 4.

The <priority> parameter specifies the internal forwarding priority.

Changing the Internal Forwarding Priority –> Hardware Forwarding Queue Mappings

To reassign an internal forwarding priority to a different hardware forwarding queue, enter commands such as the following at the global CONFIG level of the CLI:

```
FESX424 Router(config)# qos tagged-priority 2 qosp0
```

Syntax: [no] qos tagged-priority <num> <queue>

The <num> parameter can be from 0 – 7 and specifies the internal forwarding priority.

The <queue> parameter specifies the hardware forwarding queue to which you are reassigning the priority. The default queue names are as follows:

- qosp7
- qosp6
- qosp5
- qosp4
- qosp3
- qosp2
- qosp1
- qosp0

Scheduling

Scheduling is the process of mapping a packet to an internal forwarding queue based on its QoS information, and servicing the queues according to a mechanism.

This section describes the scheduling methods used on the FESX, FSX, and FWSX.

QoS Queuing Methods

The following QoS queuing methods are supported in all IronWare software releases for the FastIron FESX and FSX.

- ***Weighted Round Robin (WRR)*** – WRR ensures that all queues are serviced during each cycle. A weighted fair queuing algorithm is used to rotate service among the eight queues on FESX, FSX, and FWSX devices. The rotation is based on the weights you assign to each queue. This method rotates service among the queues, forwarding a specific number of packets in one queue before moving on to the next one.

  WRR is the default queuing method and uses a default set of queue weights.

  The number of packets serviced during each visit to a queue depends on the percentages you configure for the queues. The software automatically converts the percentages you specify into weights for the queues.

  NOTE: Queue cycles on the FESX, FSX, and FWSX are based on bytes. These devices service a given number of bytes (based on the weight) in each queue cycle. FES and BI/FI queue cycles are based on packets.

  The bytes-based scheme is more accurate compared to a packets-based scheme if packets vary greatly in size.

- ***Strict Priority(SP)*** – SP ensures service for high priority traffic. The software assigns the maximum weights to each queue, to cause the queuing mechanism to serve as many packets in one queue as possible before moving to a lower queue. This method biases the queuing mechanism to favor the higher queues over the lower queues.

  For example, strict queuing processes as many packets as possible in qosp3 before processing any packets in qosp2, then processes as many packets as possible in qosp2 before processing any packets in qosp1, and so on.

- ***Hybrid WRR and SP*** – Starting with software release 02.2.00, an additional configurable queueing mechanism combines both the strict priority and weighted round robin mechanisms. The combined method enables the Foundry device to give strict priority to delay-sensitive traffic such as VOIP traffic, and weighted round robin priority to other traffic types.

  By default, when you select the combined SP and WRR queueing method, the Foundry device assigns strict

priority to traffic in qosp7 and qosp6, and weighted round robin priority to traffic in qosp0 through qosp5. Thus, the Foundry device schedules traffic in queue 7 and queue 6 first, based on the strict priority queueing method. When there is no traffic in queue 7 and queue 6, the device schedules the other queues in round-robin fashion from the highest priority queue to the lowest priority queue.

By default, when you specify the combined SP and WRR queuing method, the system balances the traffic among the queues as shown in Table 2. If desired, you can change the default bandwidth values as instructed in the section “Changing the Bandwidth Allocations of the Hybrid WRR and SP Queues” on page 13-14.

Table 2: Default Bandwidth for Combined SP and WRR Queueing Methods

Queue	Default Bandwidth
qosp7	Strict priority (highest priority)
qosp6	Strict priority
qosp5	25%
qosp4	15%
qosp3	15%
qosp2	15%
qosp1	15%
qosp0	15% (lowest priority)

Selecting the QoS Queuing Method

FastIron devices use the weighted fair queuing method of packet prioritization by default. To change the method to strict priority or back to weighted fair queuing, enter the following command at the Global CONFIG level of the CLI:

```
FESX424 Router(config)# qos mechanism strict
```

To change the method back to weighted round robin, enter the following command:

```
FESX424 Router(config)# qos mechanism weighted
```

Syntax: [no] qos mechanism strict | weighted

NOTE: The following combined method is supported in releases 02.2.00 and later.

To change the queuing mechanism to the combined SP and WRR method, enter the following command at the Global CONFIG level of the CLI:

```
FESX424 Switch(config)#qos mechanism mixed-sp-wrr
```

Syntax: mechanism mixed-sp-wrr

Configuring the QoS Queues

Each of the queues has the following configurable parameters:

- The queue name
- The minimum percentage of a port’s outbound bandwidth guaranteed to the queue

Renaming the Queues

The default queue names on FESX, FSX, and FWSX devices are qosp7, qosp6, qosp5, qosp4, qosp3, qosp2, qosp1, and qosp0. You can change one or more of the names if desired.

To rename queue qosp3 to “92-octane”, enter the following commands:

```
FESX424 Router(config)# qos name qosp3 92-octane
```

Syntax: qos name <old-name> <new-name>

The <old-name> parameter specifies the name of the queue before the change.

The <new-name> parameter specifies the new name of the queue. You can specify an alphanumeric string up to 32 characters long.

Changing the Minimum Bandwidth Percentages of the WRR Queues

If you are using the weighted round robin mechanism instead of the strict mechanism, you can change the weights for each queue by changing the minimum percentage of bandwidth you want each queue to guarantee for its traffic.

By default, the eight QoS queues on FESX**,** FSX, and FWSX devices receive the following minimum guaranteed percentages of a port’s total bandwidth.

Table 13.9: Default Minimum Bandwidth Percentages on FESX, FSX, and FWSX Devices

Queue	Default Minimum Percentage of Bandwidth
qosp7	75%
qosp6	7%
qosp5	3%
qosp4	3%
qosp3	3%
qosp2	3%
qosp1	3%
qosp0	3%

When the queuing method is weighted round robin, the software internally translates the percentages into weights. The weight associated with each queue controls how many packets are processed for the queue at a given stage of a cycle through the weighted round robin algorithm.

NOTE: Queue cycles on the FESX, FSX, and FWSX are based on bytes. These devices service a given number of bytes (based on the weight) in each queue cycle. FES and BI/FI queue cycles are based on packets.

The bytes-based scheme is more accurate compared to a packets-based scheme if packets vary greatly in size.

The bandwidth allocated to each queue is based on the relative weights of the queues. You can change the bandwidth percentages allocated to the queues by changing the weights of the queues.

There is no minimum bandwidth requirement for a given queue. For example, queue qosp3 is not required to have at least 50% of the bandwidth.

Command Syntax

To change the bandwidth percentages for the queues, enter commands such as the following. Note that this example uses the default queue names.

```
FESX424 Switch(config)# qos profile qosp7 25 qosp6 15 qosp5 12 qosp4 12 qosp3 10
qosp2 10 qosp1 10 qosp0 6
Profile qosp7     : Priority7   bandwidth requested  25% calculated  25%
Profile qosp6     : Priority6   bandwidth requested  15% calculated  15%
Profile qosp5     : Priority5   bandwidth requested  12% calculated  12%
Profile qosp4     : Priority4   bandwidth requested  12% calculated  12%
Profile qosp3     : Priority3   bandwidth requested  10% calculated  10%
Profile qosp2     : Priority2   bandwidth requested  10% calculated  10%
Profile qosp1     : Priority1   bandwidth requested  10% calculated  10%
Profile qosp0     : Priority0   bandwidth requested   6% calculated   6%
```

Syntax: [no] qos profile <queue> <percentage> <queue> <percentage> <queue> <percentage> <queue> <percentage> <queue> <percentage> <queue> <percentage> <queue> <percentage> <queue> <percentage>

Each <queue> parameter specifies the name of a queue. You can specify the queues in any order on the command line, but you must specify each queue.

The <percentage> parameter specifies a number for the percentage of the device's outbound bandwidth that is allocated to the queue. The FESX, FSX, and FWSX QoS queues require a minimum bandwidth percentage of 3% for each priority. When jumbo frames are enabled, the minimum bandwidth requirement is 8%. If these minimum values are not met, QoS may not be accurate.

Configuration Notes

- The total of the percentages you enter must equal 100.
- The FESX, FSX, and FWSX do not adjust the bandwidth percentages you enter. BigIron QoS does adjust the bandwidth percentages to ensure that each queue has at least its required minimum bandwidth percentage.
- When sFlow is enabled, the FESX, FSX, and FWSX support seven priorities instead of eight. When sFlow is enabled, Priority 1 is not used. Any values assigned to queue 1 will be directed to queue 0.

Changing the Bandwidth Allocations of the Hybrid WRR and SP Queues

NOTE: This feature is supported in releases 02.2.00 and later.

To change the default bandwidth percentages for the queues when the device is configured to use the combined SP and WRR queuing mechanism, enter commands such as the following. Note that this example uses the default queue names.

```
FESX424 Switch(config)#qos profile qosp7 sp qosp6 sp qosp5 20 qosp4 16 qosp3 16
qosp2 16 qosp1 16 qosp0 16
```

Syntax: [no] qos profile <queue 7> sp <queue 6> sp | <percentage> <queue 5> <percentage> <queue 4> <percentage> <queue 3> <percentage> <queue 2> <percentage> <queue 1> <percentage> <queue 0> <percentage>]

Each **<queue *x*>** parameter specifies the name of a queue. You can specify the queues in any order on the command line, but you must specify each queue. Note that queue 7 supports strict priority only, queue 6 supports both strict priority and WRR queuing mechanisms, and queues 0 – 5 support the WRR queuing mechanism only.

The **sp** parameter configures strict priority as the queuing mechanism. Note that only queue 7 and queue 6 support this method.

The **<percentage>** parameter configures WRR as the queuing mechanism and specifies the percentage of the device's outbound bandwidth allocated to the queue. The queues require a minimum bandwidth percentage of

3% for each priority. When jumbo frames are enabled, the minimum bandwidth requirement is 8%. If these minimum values are not met, QoS may not be accurate.

NOTE: The total of the percentages must equal 100. The Foundry device does not adjust the bandwidth percentages you enter. In contrast, the BigIron QoS does adjust the bandwidth percentages to ensure that each queue has at least its required minimum bandwidth percentage.

NOTE: When sFlow is enabled, the Foundry device supports seven priorities instead of eight. When sFlow is enabled, Priority 1 is not used. Any values assigned to queue 1 will be directed to queue 0.

Viewing QoS Settings

To display the QoS settings for all the queues, enter the **show qos-profiles** command, as shown in the following examples.

The following shows an example display output on a FESX.

```
FESX424 Switch(config)# show qos-profiles all
bandwidth scheduling mechanism: weighted priority
Profile qosp7     : Priority7   bandwidth requested  25% calculated  25%
Profile qosp6     : Priority6   bandwidth requested  15% calculated  15%
Profile qosp5     : Priority5   bandwidth requested  12% calculated  12%
Profile qosp4     : Priority4   bandwidth requested  12% calculated  12%
Profile qosp3     : Priority3   bandwidth requested  10% calculated  10%
Profile qosp2     : Priority2   bandwidth requested  10% calculated  10%
Profile qosp1     : Priority1   bandwidth requested  10% calculated  10%
Profile qosp0     : Priority0   bandwidth requested   6% calculated   6%
```

Syntax: show qos-profiles all | <name>

The **all** parameter displays the settings for all eight queues.

The <name> parameter displays the settings for the specified queue.

Viewing DSCP-based QoS Settings

To display configuration information for DSCP-based QoS, enter the following command at any level of the CLI:

```
FastIron SuperX Switch(config)#show qos-tos
DSCP-->Traffic-Class map: (DSCP = d1d2: 00, 01...63)


     d2|  0   1   2   3   4   5   6   7   8   9
  d1   |
  -----+----------------------------------------
   0   |  0   0   0   0   0   0   0   0   1   1
   1   |  1   1   1   1   1   1   2   2   2   2
   2   |  2   2   2   2   3   3   3   3   3   3
   3   |  3   3   4   4   4   4   4   4   4   4
   4   |  5   5   5   5   5   5   5   5   6   6
   5   |  6   6   6   6   6   6   7   7   7   7
   6   |  7   7   7   7

Traffic-Class-->802.1p-Priority map (use to derive DSCP--802.1p-Priority):


Traffic | 802.1p
Class   | Priority
--------+---------
   0    |    0
   1    |    1
   2    |    2
   3    |    3
   4    |    4
   5    |    5
   6    |    6
   7    |    7
--------+---------
```

Syntax: show qos-tos

This command shows the following information.

Table 13.10: DSCP-based QoS Configuration Information

This Field...	Displays...
DSCP-Priority map	
d1 and d2	The DSCP to forwarding priority mappings that are currently in effect. **Note**: The example above shows the default mappings. If you change the mappings, the command displays the changed mappings

Table 13.10: DSCP-based QoS Configuration Information (Continued)

This Field...	Displays...
Traffic Class -> 802.1 Priority map	
Traffic Class and 802.1p Priority	The traffic class to 802.1p Priority mappings that are currently in effect. **Note**: The example above shows the default mappings. If you change the mappings, the command displays the changed mappings.

Chapter 14
Configuring Rate Limiting

This chapter describes how to configure rate limiting on Foundry FastIron devices using the CLI.

This chapter contains the following information:

Table 14.1: Chapter Contents

Overview

Fixed Rate Limiting allows you to specify the maximum number of bytes a given port can send or receive. The port drops bytes that exceed the limit you specify. You can configure a Fixed Rate Limiting policy on a port's inbound direction only. Fixed rate limiting applies to all traffic on the rate limited port.

When you specify the maximum number of bytes, you specify it in bits per second (bps). The Fixed Rate Limiting policy applies to one-second intervals and allows the port to send or receive the number of bytes you specify in the policy, but drops additional bytes. Unused bandwidth is not carried over from one interval to the next.

NOTE: Foundry recommends that you do not use Fixed Rate Limiting on ports that send or receive route control traffic or Spanning Tree Protocol (STP) control traffic. If the port drops control packets due to the Fixed Rate Limiting policy, routing or STP can be disrupted.

Rate Limiting in Hardware

Each FastIron device supports line-rate rate limiting in hardware. The device creates entries in Content Addressable Memory (CAM) for the rate limiting policies. The CAM entries enable the device to perform the rate limiting in hardware instead of sending the traffic to the CPU. The device sends the first packet in a given traffic

flow to the CPU, which creates a CAM entry for the traffic flow. A CAM entry consists of the source and destination addresses of the traffic. The device uses the CAM entry for rate limiting all the traffic within the same flow. A rate limiting CAM entry remains in the CAM for two minutes before aging out.

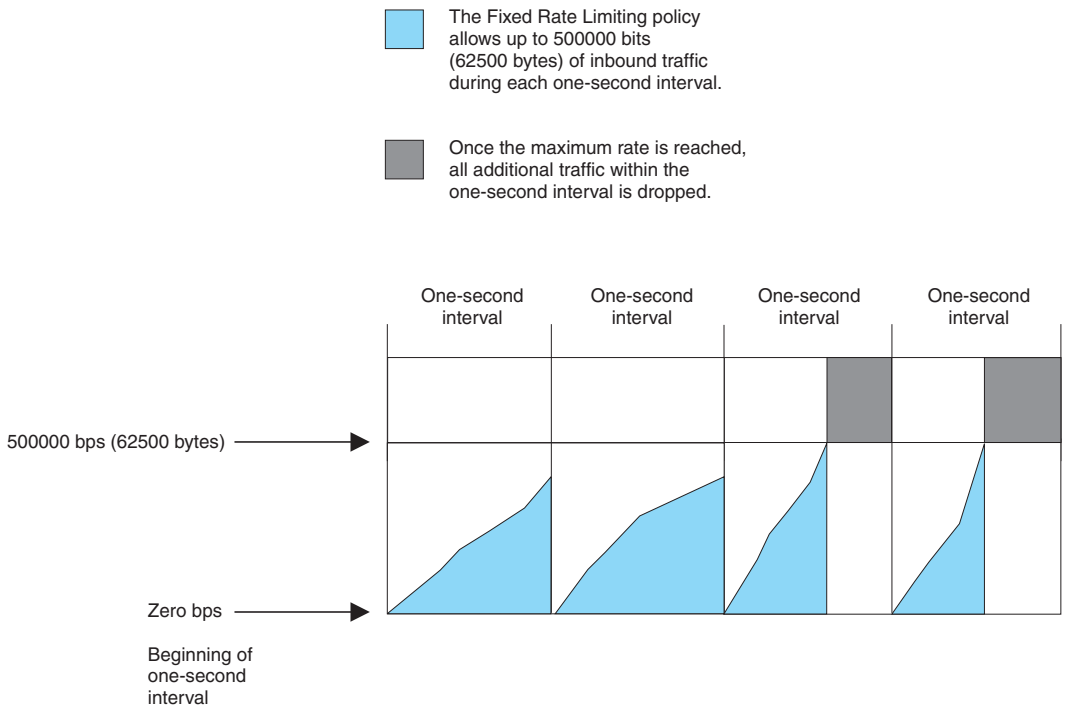
How Fixed Rate Limiting Works

Fixed Rate Limiting counts the number of bytes that a port either sends or receives, in one second intervals. The direction that the software monitors depends on the direction you specify when you configure the rate limit on the port. If the number of bytes exceeds the maximum number you specify when you configure the rate, the port drops all further packets for the rate-limited direction, for the duration of the one-second interval.

Once the one-second interval is complete, the port clears the counter and re-enables traffic.

Figure 14.1 shows an example of how Fixed Rate Limiting works. In this example, a Fixed Rate Limiting policy is applied to a port to limit the inbound traffic to 500000 bits (62500 bytes) a second. During the first two one-second intervals, the port receives less than 500000 bits in each interval. However, the port receives more than 500000 bits during the third and fourth one-second intervals, and consequently drops the excess traffic.

Figure 14.1 Fixed Rate Limiting



NOTE: The software counts the bytes by polling statistics counters for the port every 100 milliseconds, which provides 10 readings each second. Due to the polling interval, the Fixed Rate Limiting policy has an accuracy of within 10% of the port's line rate. It is therefore possible for the policy to sometimes allow more traffic than the limit you specify, but the extra traffic is never more than 10% of the port's line rate.

Configuration Notes

The following FastIron Ironware releases support fixed rate limiting for inbound traffic on individual ports (port-based rate limiting):

- FESX releases 01.1.00 and later
- All FWSX software releases
- All FSX software releases

- Rate limiting is available only on inbound ports on FastIron devices.
- Fixed rate limiting is not supported on 10-Gigabit Ethernet ports.

Configuring a Port-Based Rate Limiting Policy

To configure rate limiting on a port, enter commands such as the following:

```
FESX424 Router(config)# interface ethernet 24
FESX424 Router(config-if-e1000-24)# rate input fixed 500000
```

These commands configure a fixed rate limiting policy that allows port 24 to receive a maximum of 500000 bits per second (62500 bytes per second). If the port receives additional bytes during a given one-second interval, the port drops all inbound packets on the port until the next one-second interval starts.

Syntax: [no] rate-limit input fixed <average-rate>

The <average-rate> parameter specifies the maximum number of bits per second (bps) the port can receive. The minimum rate that can be configured on FESX, FSX, and FWSX devices is 64,000 bps.

Configuring an ACL-Based Rate Limiting Policy

Software releases 02.3.03 and later provide support for IP ACL-based rate limiting of inbound traffic. ACL-based rate limiting provides the facility to limit the rate for IP traffic that matches the permit conditions in extended IP ACLs. This feature is available in the Layer 2 and Layer 3 code.

To configure ACL-based rate limiting on an X-Series device, you create individual ***traffic policies***, then reference the traffic policies in one or more ACL entries (also called clauses or statements). The traffic policies become effective on ports to which the ACLs are bound.

Configuration procedures for ACL-based rate limiting are in the chapter “Traffic Policies” on page 15-1.

Optimizing Rate Limiting

By default, rate limiting is optimized for packets that are 256 bytes in size. This packet size includes 14 bytes of Layer 2 header (Ethernet II untagged) and 4 bytes of Layer 2 CRC.

To optimize rate limiting for all packet sizes, use the **payload-only** parameter. When this parameter is specified, the system excludes Layer 2 header and Layer 2 checksum (CRC) from the calculations, and the rate is accurate for all packet sizes and Layer 2 overhead (Layer 2 header + CRC). Layer 2 overhead for different encapsulations is as follows:

- Untagged Ethernet-II – 18 bytes
- Tagged Ethernet-II – 22 bytes
- LLC over Untagged Ethernet-II – 21 bytes
- LLC over Tagged Ethernet-II – 25 bytes
- LLC/SNAP over Untagged Ethernet-II – 26 bytes
- LLC/SNAP over Tagged Ethernet-II – 30 bytes

To optimize rate limiting, enter commands such as the following:

```
FESX424 Router(config)# interface ethernet 24
FESX424 Router(config-if-e1000-24)# rate input fixed 500000 payload-only
```

These commands configure a fixed rate limiting policy that allows port 24 to receive a maximum of 500000 bits per second. The payload-only parameter causes the device to exclude the Layer 2 header and Layer 2 checksum from the calculations.

NOTE: When you enable the **payload-only** parameter on the FESX, FSX, and FWSX devices, the configuration applies to all the other ports in the same port region. For example, if you enable the **payload-only** option on port 12 on a FESX424, the configuration applies to ports 1 through 12 since these ports are in the same port region.

Syntax: [no] rate-limit input fixed <average-rate> [payload-only]

The <average-rate> parameter specifies the maximum number of bits per second (bps) the port can receive. The minimum rate that can be configured on FESX, FSX, and FWSX devices is 64,000 bps. By default, rate limiting is optimized for packets that are 256 bytes in size.

Displaying the Fixed Rate Limiting Configuration

To display the fixed rate limiting configuration on the device, enter the following command:

```
FESX424 Switch(config-if-e1000-21)#show rate-limit fixed
Total rate-limited interface count: 11.
 Port    Configured Input Rate        Actual Input Rate             Mode
    1                1000000                  1000000      Payload-Only
    3               10000000                 10005000           Default
    7               10000000                 10000000      Payload-Only
    9                7500000                  7502000      Payload-Only
   11                8000000                  7999000           Default
   12                8000000                  7999000           Default
   13                8000000                  7999000           Default
   14                8000000                  7999000           Default
   15                8000000                  7999000           Default
   21                8000000                  8000000      Payload-Only
   25                7500000                  7502000           Default
```

Syntax: show rate-limit fixed

The command lists the ports on which fixed rate limiting is configured, and provides the information listed in Table 14.2 for each of the ports.

Table 14.2: CLI Display of Fixed Rate Limiting Information

This Field...	Displays...
Total rate-limited interface count	The total number of ports that are configured for Fixed Rate Limiting.
Port	The port number.
Configured Input Rate	The maximum rate requested for inbound traffic. The rate is measured in bits per second (bps).
Actual Input Rate	The actual maximum rate provided by the hardware. The rate is measured in bps.

Chapter 15
Traffic Policies

NOTE: Traffic policies are supported in FastIron X-Series devices running software release 02.3.03 or later.

X-Series devices use ***traffic policies*** to:

- Rate limit inbound traffic
- Count the packets and bytes per packet to which ACL permit or deny clauses are applied

This chapter describes how traffic policies are implemented and configured in the FESX, FSX, and FWSX devices. This chapter contains the topics listed in Table 15.1.

Table 15.1: Chapter Contents

About Traffic Policies

Traffic policies consist of policy names and policy definitions.

- ***Traffic policy name*** – This is a string of up to 8 alphanumeric characters that identifies individual traffic policy definitions.
- ***Traffic policy definition*** (***TPD***) – This is the command filter associated with a traffic policy name. A TPD can define any one of the following:
  - Rate limiting policy
  - ACL counting policy

- Combined rate limiting and ACL counting policy

The maximum number of supported active TPDs is a system-wide parameter and depends on the device you are configuring. The total number of active TPDs cannot exceed the system maximum. See “Maximum Number of Traffic Policies Supported on a Device” on page 15-3.

When you apply a traffic policy to an interface, you do so by adding a reference to the traffic policy in an ACL entry, instead of applying the individual traffic policy to the interface. The traffic policy becomes an ***active traffic policy*** or ***active TPD*** when you bind its associated ACL to an interface.

To configure traffic policies for ACL-based rate limiting, see “Configuring ACL-Based Fixed Rate Limiting” on page 15-4 and “Configuring ACL-Based Adaptive Rate Limiting” on page 15-5.

To configure traffic policies for ACL counting, see “Enabling ACL Counting” on page 15-8.

Configuration Notes and Feature Limitations

Note the following when configuring traffic policies:

- This feature is supported in the Layer 2 and Layer 3 code.
- This feature applies to IP ACLs only. X-Series devices do not support Layer 2 ACLs.
- Traffic policies are not supported on 10-Gigabit Ethernet interfaces.
- The maximum number of supported active TPDs is a system-wide parameter and depends on the device you are configuring. The total number of active TPDs cannot exceed the system maximum. See “Maximum Number of Traffic Policies Supported on a Device” on page 15-3.
- You can reference the same traffic policy in more than one ACL entry within an access list. For example, two or more ACL statements in ACL 101 can reference a TPD named TPD1.
- You can reference the same traffic policy in more than one access list. For example, ACLs 101 and 102 could both reference a TPD named TPD1.
- To modify or delete an active traffic policy, you must first unbind the ACL that references the traffic policy.
- When you define a TPD (when you enter the CLI command **traffic-policy**), explicit marking of CoS parameters, such as traffic class and 802.1p priority, are not available on the device. In the case of a TPD

defining rate limiting, the device re-marks CoS parameters based on the DSCP value in the packet header and the determined conformance level of the rate limited traffic, as shown in Table 15.2.

Table 15.2: CoS Parameters for Packets that use Rate Limiting Traffic Policies

If the packet's Conformance Level is...	and the packet's DSCP Value is...	the device sets the Traffic Class and 802.1p Priority to...
0 (Green) or 1 (Yellow)	0 – 7	0 (lowest priority queue)
	8 – 15	1
	16 – 23	2
	24 – 31	3
	32 – 39	4
	40 – 47	5
	48 – 55	6
	56 – 63	7 (highest priority queue)
2 (Red)	N/A	0 (lowest priority queue)

Maximum Number of Traffic Policies Supported on a Device

The maximum number of supported active traffic policies is a system-wide parameter and depends on the device you are configuring, as follows:

- By default, up to 1024 active traffic policies are supported on Layer 2 switches. This value is fixed on Layer 2 switches and cannot be modified.
- The number of active traffic policies supported on Layer 3 switches varies depending on the configuration and the available system memory. The default value and also the maximum number of traffic policies supported on Layer 3 switches is 50.

Setting the Maximum Number of Traffic Policies Supported on a Layer 3 Device

If desired you can adjust the maximum number of active traffic policies that a Layer 3 device will support. To do so, enter commands such as the following at the Global CONFIG level of the CLI:

```
FESX424 Switch(config)# system-max hw-traffic-conditioner 25
FESX424 Switch(config)# write memory
FESX424 Switch(config)# reload
```

NOTE: You must save the configuration and reload the software to place the change into effect.

Syntax: [no] system-max hw-traffic-conditioner <num>

<num> is a value from 0 to *n*, where 0 disables hardware resources for traffic policies, and *n* is a number up to 1024. The maximum number you can configure depends on the configuration and available memory on your device. If the configuration you enter causes the device to exceed the available memory, the device will reject the configuration and display a warning message on the console.

NOTE: Foundry does not recommend setting the system-max for traffic policies to 0 (zero), since this renders traffic policies ineffective.

ACL-Based Rate Limiting via Traffic Policies

Software release 02.3.03 provides support for IP ACL-based rate limiting of inbound traffic. ACL-based rate limiting provides the facility to limit the rate for IP traffic that matches the permit conditions in extended IP ACLs. This feature is available in the Layer 2 and Layer 3 code.

To configure ACL-based rate limiting on an X-Series device, you create individual ***traffic policies***, then reference the traffic policies in one or more ACL entries (also called clauses or statements). The traffic policies become effective on ports to which the ACLs are bound. See “About Traffic Policies” on page 15-1.

When you configure a traffic policy for rate limiting, the device automatically enables ***rate limit counting***, similar to the two-rate three-color marker (trTCM) mechanism described in RFC 2698 for adaptive rate limiting, and the single-rate three-color marker (srTCM) mechanism described in RFC 2697 for fixed rate limiting. This feature counts the number of bytes and trTCM or srTCM conformance level per packet to which rate limiting traffic policies are applied. See “ACL and Rate Limit Counting” on page 15-8.

You can configure ACL-based rate limiting on the following interface types:

- physical Ethernet interfaces
- virtual interfaces
- trunk ports
- specific VLAN members on a port (New in 02.3.03 – see “Applying an ACL to Specific VLAN Members on a Port (Layer 2 Devices Only)” on page 12-21
- a subset of ports on a virtual interface (New in 02.3.03 – see “Applying an ACL to a Subset of Ports on a Virtual Interface (Layer 3 Devices Only)” on page 12-21.)

Support for Fixed Rate Limiting and Adaptive Rate Limiting

X-Series devices support the following types of ACL-based rate limiting:

- Fixed Rate Limiting – Enforces a strict bandwidth limit. The device forwards traffic that is within the limit but either drops all traffic that exceeds the limit, or forwards all traffic that exceeds the limit at the lowest priority level, according to the action specified in the traffic policy.
- Adaptive Rate Limiting – Enforces a flexible bandwidth limit that allows for bursts above the limit. You can configure Adaptive Rate Limiting to forward, modify the IP precedence of and forward, or drop traffic based on whether the traffic is within the limit or exceeds the limit.

Configuring ACL-Based Fixed Rate Limiting

Use the procedures in this section to configure ACL-based fixed rate limiting. Before configuring this feature, see what to consider in “Configuration Notes and Feature Limitations” on page 15-2.

Fixed rate limiting enforces a strict bandwidth limit. The port forwards traffic that is within the limit. If the port receives more than the specified number of fragments in a one-second interval, the device either drops or forwards subsequent fragments in hardware, depending on the action you specify.

To implement the ACL-based fixed rate limiting feature, first create a traffic policy, then reference the policy in an extended ACL statement. Lastly, bind the ACL to an interface. Follow the steps below.

1. Create a traffic policy. Enter a command such as the following:

```
FESX424 Switch(config)# traffic-policy TPD1 rate-limit fixed 100 exceed-action drop
```

2. Create an extended ACL entry or modify an existing extended ACL entry that references the traffic policy. For example:

```
FESX424 Switch(config)# access-list 101 permit ip host 210.10.12.2 any traffic-policy TPD1
```


3. Bind the ACL to an interface.

```
FESX424 Switch(config)# int e 5
FESX424 Switch(config-if-e5)# ip access-group 101 in
FESX424 Switch(config-if-e5)# exit
```

The above commands configure a fixed rate limiting policy that allows port e5 to receive a maximum traffic rate of 100 kbps. If the port receives additional bits during a given one-second interval, the port drops the additional inbound packets that are received within that one-second interval.

Syntax: [no] traffic-policy <TPD name> rate-limit fixed <cir value> exceed-action <action> [count]

Syntax: access-list <num> permit | deny.... traffic policy <TPD name>

Syntax: [no] ip access-group <num> in | out

NOTES:

For brevity, some parameters were omitted from the above **access-list** syntax. For the complete CLI syntax, see the *Foundry Switch and Router Command Line Interface Reference*.

The software allows you to add a reference to a non-existent TPD in an ACL statement and to bind that ACL to an interface. The software does not issue a warning or error message for non-existent TPDs.

Use the **no** form of the command to delete a traffic policy definition. Note that you cannot delete a traffic policy definition if it is currently in use on a port. To delete a traffic policy, first unbind the associated ACL.

<TPD name> is the name of the traffic policy definition. This value can be 8 or fewer alphanumeric characters.

rate-limit fixed specifies that the traffic policy will enforce a strict bandwidth.

<cir value> is the committed information rate in kbps. This value can be from 64 – 1000000 Kbps.

exceed-action <action> specifies the action to be taken when packets exceed the configured cir value. See “Specifying the Action to be Taken for Packets that are Over the Limit” .

The **count** parameter is optional and enables ACL counting. See “ACL and Rate Limit Counting” on page 15-8.

Configuring ACL-Based Adaptive Rate Limiting

Use the procedures in this section to configure ACL-based adaptive rate limiting. Before configuring this feature, see what to consider in “Configuration Notes and Feature Limitations” on page 15-2.

Table 1 lists the configurable parameters for ACL-based adaptive rate limiting:

Table 1: ACL-Based Adaptive Rate Limiting Parameters

Parameter	Definition
Committed Information Rate (CIR)	The guaranteed kilobit rate of inbound traffic that is allowed on a port.
Committed Burst Size (CBS)	The number of bytes per second allowed in a burst before some packets will exceed the committed information rate. Larger bursts are more likely to exceed the rate limit. The CBS must be a value greater than zero (0). Foundry recommends that this value be equal to or greater than the size of the largest possible IP packet in a stream.
Peak Information Rate (PIR)	The peak maximum kilobit rate for inbound traffic on a port. The PIR must be equal to or greater than the CIR.

Table 1: ACL-Based Adaptive Rate Limiting Parameters (Continued)

Parameter	Definition
Peak Burst Size (PBS)	The number of bytes per second allowed in a burst before all packets will exceed the peak information rate. The PBS must be a value greater than zero (0). Foundry recommends that this value be equal to or greater than the size of the largest possible IP packet in the stream.

If a port receives more than the configured bit or byte rate in a one-second interval, the port will either drop or forward subsequent data in hardware, depending on the action you specify.

To implement the ACL-based adaptive rate limiting feature, first create a traffic policy then reference the policy in an extended ACL statement. Lastly, bind the ACL to an interface. Follow the steps below.

1. Create a traffic policy. Enter a command such as the following:

```
FESX424 Switch(config)# traffic-policy TPDAfour rate-limit adaptive cir 10000
cbs 1600 pir 20000 pbs 4000 exceed-action drop
```

2. Create a new extended ACL entry or modify an existing extended ACL entry that references the traffic policy. For example:

```
FESX424 Switch(config)# access-list 104 permit ip host 210.10.12.2 any traffic-
policy TPDAfour
```

3. Bind the ACL to an interface.

```
FESX424 Switch(config)# int e 7
FESX424 Switch(config-if-e7)# ip access-group 104 in
FESX424 Switch(config-if-e7)# exit
```

The above commands configure an adaptive rate limiting policy that enforces a guaranteed committed rate of 10000 kbps on port e7 and allows bursts of up to 1600 bytes. It also enforces a peak rate of 20000 kbps and allows bursts of 4000 bytes above the PIR limit. If the port receives additional bits during a given one-second interval, the port drops all packets on the port until the next one-second interval starts.

Syntax: [no] traffic-policy <TPD name> rate-limit adaptive cir <cir value> cbs <cbs value> pir <pir value> pbs <pbs value> exceed-action <action> [count]

Syntax: access-list <num> permit | deny.... traffic policy <TPD name>

Syntax: [no] ip access-group <num> in | out

NOTES:

For brevity, some parameters were omitted from the above **access-list** syntax. For the complete CLI syntax, see the *Foundry Switch and Router Command Line Interface Reference*.

The software allows you to add a reference to a non-existent TPD in an ACL statement and to bind that ACL to an interface. The software does not issue a warning or error message for non-existent TPDs.

Use the **no** form of the command to delete a traffic policy definition. Note that you cannot delete a traffic policy definition if it is currently in use on a port. To delete a traffic policy, first unbind the associated ACL.

<TPD name> is the name of the traffic policy definition. This value can be 8 or fewer alphanumeric characters.

rate-limit adaptive specifies that the policy will enforce a flexible bandwidth limit that allows for bursts above the limit.

<cir value> is the committed information rate in kbps. See Table 1.

<cbs value> is the committed burst size in bytes. See Table 1.

<pir value> is the peak information rate in kbps. See Table 1.

<pbs value> is the peak burst size in bytes. See Table 1.

exceed-action <action> specifies the action to be taken when packets exceed the configured values. See "Specifying the Action to be Taken for Packets that are Over the Limit" .

The **count** parameter is optional and enables ACL counting. See "ACL and Rate Limit Counting" on page 15-8.

Specifying the Action to be Taken for Packets that are Over the Limit

You can specify the action to be taken when packets exceed the configured cir value for fixed rate limiting, or the cir, cbs, pir, and pbs values for adaptive rate limiting. You can specify one of the following actions:

- drop packets that exceed the limit
- permit packets that exceed the limit and forward them at the lowest priority level

Dropping Packets that Exceed the Limit

This section shows some example configurations and provides the CLI syntax for configuring a port to drop packets that exceed the configured limit(s) for rate limiting.

EXAMPLE:

The following shows an example fixed rate limiting configuration.

```
FESX424 Switch(config)# traffic-policy TPD1 rate-limit fixed 10000 exceed-action
drop
```

The above command sets the fragment threshold at 10,000 per second. If the port receives more than 10,000 packet fragments in a one-second interval, the device drops the excess fragments.

Syntax: traffic-policy <TPD name> rate-limit fixed <cir value> exceed-action drop

EXAMPLE:

The following shows an example adaptive rate limiting configuration.

```
FESX424 Switch(config)# traffic-policy TPDAfour rate-limit adaptive cir 10000 cbs
1600 pir 20000 pbs 4000 exceed-action drop
```

The above commands configure an adaptive rate limiting policy that enforces a guaranteed committed rate of 10000 kbps on port e7 and allows bursts of up to 1600 bytes. It also enforces a peak rate of 20000 kbps and allows bursts of 4000 bytes above the PIR limit. If the port receives additional bits during a given one-second interval, the port drops all packets on the port until the next one-second interval starts.

Syntax: traffic-policy rate-limit adaptive cir <cir value> cbs <cbs value> pir <pir value> pbs <pbs value> exceed-action drop

Permitting Packets that Exceed the Limit

This section shows some example configurations and provides the CLI syntax for configuring a port to permit packets that exceed the configured limit for rate limiting.

EXAMPLE:

The following shows an example fixed rate limiting configuration.

```
FESX424 Switch(config)# traffic-policy TPD1 rate-limit fixed 10000 exceed-action
permit-at-low-pri
```

The above command sets the fragment threshold at 10,000 per second. If the port receives more than 10,000 packet fragments in a one-second interval, the device takes the specified action. The action specified with this command is to permit excess fragments and forward them at the lowest priority level.

Syntax: [no] traffic-policy <TPD name> rate-limit fixed <cir value> exceed-action permit-at-low-pri

EXAMPLE:

The following shows an example adaptive rate limiting configuration.

```
FESX424 Switch(config)# traffic-policy TPDAfour rate-limit adaptive cir 10000 cbs
1600 pir 20000 pbs 4000 exceed-action permit-at-low-pri
```

The above commands configure an adaptive rate limiting policy that enforces a guaranteed committed rate of 10000 kbps on port e7 and allows bursts of up to 1600 bytes. It also enforces a peak rate of 20000 kbps and allows bursts of 4000 bytes above the PIR limit. If the port receives additional bits during a given one-second interval, the port permits all packets on the port and forwards the packets at the lowest priority level.

Syntax: traffic-policy rate-limit adaptive cir <cir value> cbs <cbs value> pir <pir value> pbs <pbs value> exceed-action permit-at-low-pri

ACL and Rate Limit Counting

Software release 02.3.03 provides support for ***ACL counting*** and ***rate limit counting***.

- ***ACL counting*** enables the Foundry device to count the number of packets and the number of bytes per packet to which ACL filters are applied.
- ***Rate limit counting*** counts the number of bytes and conformance level per packet to which rate limiting traffic policies are applied. The device uses the counting method similar to the two-rate three-color marker (trTCM) mechanism described in RFC 2698 for adaptive rate limiting, and the single-rate three-color marker (srTCM) mechanism described in RFC 2697 for fixed rate limiting. Rate limit counting is automatically enabled when a traffic policy is enforced (active). You can view these counters using the **show** commands listed in “Viewing Traffic Policies” on page 15-11.

For more information about traffic policies, see “About Traffic Policies” on page 15-1.

This section provides the following procedures for ACL counting and rate limit counting:

Enabling ACL Counting

Use the procedures in this section to configure ACL counting. Before configuring this feature, see what to consider in “Configuration Notes and Feature Limitations” on page 15-2.

To enable ACL counting on an X-Series device, first create a ***traffic policy***, then reference the traffic policy in an extended ACL entry. Lastly, bind the ACL to an interface. The ACL counting policy becomes effective on ports to which the ACLs are bound.

You also can enable ACL counting when you create a traffic policy for rate limiting. See “Enabling ACL Counting with Rate Limiting Traffic Policies” on page 15-9.

To implement the ACL counting feature, perform the following steps:

1. Create a traffic policy. Enter a command such as the following:

   ```
   FESX424 Switch(config)# traffic-policy TPD5 count
   ```

2. Create an extended ACL entry or modify an existing extended ACL entry that references the traffic policy definition. For example:

   ```
   FESX424 Switch(config)# access-list 101 permit ip host 210.10.12.2 any traffic-
   policy TPD5
   ```

3. Bind the ACL to an interface.

   ```
   FESX424 Switch(config)# int e 4
   FESX424 Switch(config-if-e4)# ip access-group 101 in
   FESX424 Switch(config-if-e4)# exit
   ```

The above commands configure an ACL counting policy and apply it to port e4. Port e4 counts the number of packets and the number of bytes on the port that were permitted or denied by ACL filters.

Syntax: [no] traffic-policy <TPD name> count

Syntax: access-list <num> permit | deny.... traffic policy <TPD name>

Syntax: [no] ip access-group <num> in | out

NOTES:

For brevity, some parameters were omitted from the above **access-list** syntax. For the complete CLI syntax, see the *Foundry Switch and Router Command Line Interface Reference*.

The software allows you to add a reference to a non-existent TPD in an ACL statement and to bind that ACL to an interface. The software does not issue a warning or error message for non-existent TPDs.

Use the **no** form of the command to delete a traffic policy definition. Note that you cannot delete a traffic policy definition if it is currently in use on a port. To delete a traffic policy, first unbind the associated ACL.

<TPD name> is the name of the traffic policy definition. This value can be 8 alphanumeric characters or less.

Enabling ACL Counting with Rate Limiting Traffic Policies

The configuration example in the section “Enabling ACL Counting” shows how to enable ACL counting without having to configure parameters for rate limiting. You also can enable ACL counting while defining a rate limiting traffic policy, as illustrated in the following configuration examples.

EXAMPLE:

To enable ACL counting while defining traffic policies for fixed rate limiting, enter commands such as the following at the Global CONFIG Level of the CLI:

```
FESX424 Switch(config)# traffic-policy TPD1 rate-limit fixed 1000 count exceed-
action drop
FESX424 Switch(config)# traffic-policy TPD2 rate-limit fixed 10000 exceed-action
drop count
```

Syntax: [no] traffic-policy <TPD name> rate-limit fixed <cir value> exceed-action <action> count

EXAMPLE:

To enable ACL counting while defining traffic policies for adaptive rate limiting, enter commands such as the following at the Global CONFIG Level of the CLI:

```
traffic-policy TPDA4 rate-limit adaptive cir 10000 cbs 1600 pir 20000 pbs 4000 count
exceed-action drop
traffic-policy TPDA5 rate-limit adaptive cir 10000 cbs 1600 pir 20000 pbs 4000
exceed-action permit-at-low-pri count
```

Syntax: traffic-policy rate-limit adaptive cir <cir value> cbs <cbs value> pir <pir value> pbs <pbs value> exceed-action <action> count

Viewing ACL And Rate Limit Counters

When ACL counting is enabled on the Foundry device, you can use **show** commands to display the total packet count and byte count of the traffic filtered by ACL statements. The output of the show commands also display the rate limiting traffic counters, which are automatically enabled for active rate limiting traffic policies.

Use either the **show access-list accounting** command or the **show statistics traffic-policy** command to display ACL and traffic policy counters. The output of these commands are identical. The following shows an example output.

```
FESX424 Switch# show access-list accounting g_voip

Traffic Policy - g_voip:

General Counters:
Port Region#                    Byte Count              Packet Count
------------------   ---------------------   ----------------------
7 (4/1 - 4/12)                    85367040                    776064
All port regions                  84367040                    776064

Rate Limiting Counters:
Port Region#         Green Conformance  Yellow Conformance     Red Conformance
------------------  ------------------  ------------------  ------------------
7 (4/1 - 4/12)      329114195612139520   37533986897781760                   0
All port regions    329114195612139520   37533986897781760                   0
```

Syntax: show access-list accounting traffic-policy [<TPD name>]

OR

Syntax: show statistics traffic-policy [<TPD name>]

Table 2 defines the output of the **show access-list accounting** and **show statistics traffic-policy** commands.

Table 2: ACL and Rate Limit Counting Statistics

This Line...	Displays...
Traffic Policy	The name of the traffic policy.
General Counters:	
Port Region #	The port region to which the active traffic policy applies.
Byte Count	The number of bytes that were filtered (matched ACL clauses).
Packet Count	The number of packets that were filtered (matched ACL clauses).
Rate Limiting Counters:	
Port Region#	The port region to which the active traffic policy applies.
Green Conformance	The number of bytes that did not exceed the CIR packet rate.
Yellow Conformance	The number of bytes that exceeded the CIR packet rate.
Red Conformance	The number of bytes that exceeded the PIR packet rate.

Clearing ACL and Rate Limit Counters

The Foundry device keeps a running tally of the number of packets and the number of bytes per packet that are filtered by ACL statements and rate limiting traffic policies. You can clear these accumulated counters, essentially resetting them to zero. To do so, use either the **clear access-list account traffic-policy** or the **clear statistics traffic-policy** command.

To clear the counters for ACL counting and rate limit counting, enter commands such as the following:

```
FESX424 Switch(config)# clear access-list accounting traffic-policy CountOne
FESX424 Switch(config)# clear statistics traffic-policy CountTwo
```

Syntax: clear access-list accounting traffic-policy <TPD name>

OR

Syntax: clear statistics traffic-policy <TPD name>

where <TPD name> is the name of the traffic policy definition for which you want to clear traffic policy counters.

Viewing Traffic Policies

To view traffic policies that are currently defined on the Foundry device, enter the **show traffic-policy** command. An example display output is shown below. Table 3 defines the output.

```
FESX424 Switch# show traffic-policy t_voip

Traffic Policy - t_voip:

Metering Enabled, Parameters:
          Mode:  Adaptive Rate-Limiting
          cir: 100 kbps,     cbs: 2000 bytes,     pir: 200 kbps,     pbs: 4000 bytes

Counting Not Enabled
Number of References/Bindings:1
```

Syntax: show traffic-policy [<TPD name>]

To display all traffic policies, enter the **show traffic-policy** command without entering a TPD name.

Table 3: Traffic Policy Information

This Line...	Displays...
Traffic Policy	The name of the traffic policy.
Metering	Shows whether or not rate limiting was configured as part of the traffic policy. • Enabled – The traffic policy includes a rate limiting configuration. • Disabled – The traffic policy does not include a rate limiting configuration
Mode	If rate limiting is enabled, this field shows the type of metering enabled on the port: • Fixed Rate-Limiting • Adaptive Rate-Limiting
cir	The committed information rate, in kbps, for the adaptive rate-limiting policy.
cbs	The committed burst size, in bytes per second, for the adaptive rate-limiting policy.

Table 3: Traffic Policy Information (Continued)

This Line...	Displays...
pir	The peak information rate, in kbps, for the adaptive rate-limiting policy.
pbs	The peak burst size, in bytes per second, for the adaptive rate-limiting policy.
Counting	Shows whether or not ACL counting was configured as part of the traffic policy. • Enabled – The traffic policy includes an ACL counting configuration. • Disabled – The traffic policy does not include an ACL traffic counting configuration.
Number of References/ Bindings	The number of times this traffic policy is referenced in an ACL statement and the number of active bindings for this traffic policy.

Chapter 16
Configuring IP

This chapter describes the Internet Protocol (IP) parameters on Foundry Layer 2 Switches and Layer 3 Switches and how to configure them.

NOTE: References to chassis-based Layer 3 Switches apply to the FastIron SuperX Switch.

This chapter contains the following information:

Table 16.1: Chapter Contents

Basic Configuration

IP is enabled by default. Basic configuration consists of adding IP addresses and, for Layer 3 Switches, enabling a route exchange protocol, such as Routing Information Protocol (RIP).

- If you are configuring a Layer 3 Switch, see “Configuring IP Addresses” on page 16-17 to add IP addresses, then see one or more of the following to enable and configure the route exchange protocols:
  - “Configuring RIP” on page 17-1
  - “Configuring OSPF” on page 20-1
  - “Configuring BGP4” on page 21-1

- If you are configuring a Layer 2 Switch, see “Configuring the Management IP Address and Specifying the Default Gateway” on page 16-51 to add an IP address for management access through the network and to specify the default gateway.

The rest of this chapter describes IP and how to configure it in more detail. Use the information in this chapter if you need to change some of the IP parameters from their default values or you want to view configuration information or statistics.

Overview

Foundry Networks Layer 2 Switches and Layer 3 Switches support Internet Protocol (IP) version 4. IP support on Foundry Layer 2 Switches consists of basic services to support management access and access to a default gateway. IP support on Foundry Layer 3 Switches includes all of the following, in addition to a highly configurable implementation of basic IP services including Address Resolution Protocol (ARP), ICMP Router Discovery Protocol (IRDP), and Reverse ARP (RARP):

- Route exchange protocols
  - Routing Information Protocol (RIP)
  - Open Shortest Path First (OSPF)
  - Border Gateway Protocol version 4 (BGP4)
- Multicast protocols
  - Internet Group Membership Protocol (IGMP)
  - Protocol Independent Multicast Dense (PIM-DM)
  - Protocol Independent Multicast Sparse (PIM-SM)
  - Distance Vector Multicast Routing Protocol (DVMRP)
- Router redundancy protocols
  - Virtual Router Redundancy Protocol Extended (VRRPE)
  - Virtual Router Redundancy Protocol (VRRP)

IP Interfaces

Foundry Layer 3 Switches and Layer 2 Switches allow you to configure IP addresses. On Layer 3 Switches, IP addresses are associated with individual interfaces. On Layer 2 Switches, a single IP address serves as the management access address for the entire device.

All Foundry Layer 3 Switches and Layer 2 Switches support configuration and display of IP address in classical sub-net format (example: 192.168.1.1 255.255.255.0) and Classless Interdomain Routing (CIDR) format (example: 192.168.1.1/24). You can use either format when configuring IP address information. IP addresses are displayed in classical sub-net format by default but you can change the display format to CIDR. See “Changing the Network Mask Display to Prefix Format” on page 16-57.

Layer 3 Switches

Foundry Layer 3 Switches allow you to configure IP addresses on the following types of interfaces:

- Ethernet ports
- Virtual routing interfaces (used by VLANs to route among one another)
- Loopback interfaces

Each IP address on a Layer 3 Switch must be in a different sub-net. You can have only one interface that is in a given sub-net. For example, you can configure IP addresses 192.168.1.1/24 and 192.168.2.1/24 on the same Layer 3 Switch, but you cannot configure 192.168.1.1/24 and 192.168.1.2/24 on the same Layer 3 Switch.

You can configure multiple IP addresses on the same interface.

The number of IP addresses you can configure on an individual interface depends on the Layer 3 Switch model. To display the maximum number of IP addresses and other system parameters you can configure on a Layer 3 Switch, see the section “Displaying and Modifying System Parameter Default Settings” on page 4-8.

You can use any of the IP addresses you configure on the Layer 3 Switch for Telnet, Web management, or SNMP access.

Layer 2 Switches

You can configure an IP address on a Foundry Layer 2 Switch for management access to the Layer 2 Switch. An IP address is required for Telnet access, Web management access, and SNMP access.

You also can specify the default gateway for forwarding traffic to other sub-nets.

IP Packet Flow Through a Layer 3 Switch

Figure 16.1 shows how an IP packet moves through a Foundry Layer 3 Switch.

Figure 16.1 IP Packet flow through aFoundry Layer 3 Switch

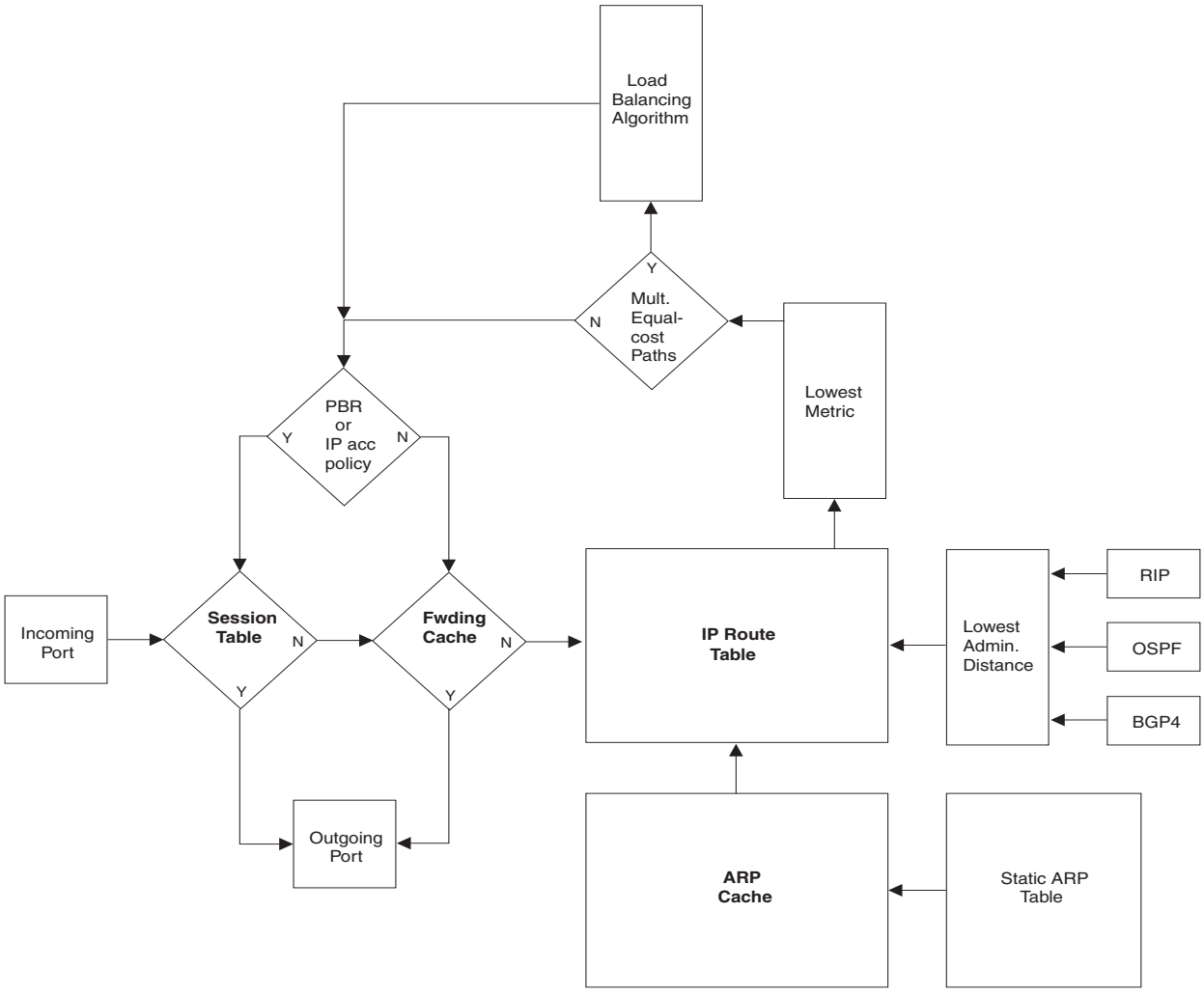


Figure 16.1 shows the following packet flow:

1. When the Layer 3 Switch receives an IP packet, the Layer 3 Switch checks for filters on the receiving interface.[1] If a deny filter on the interface denies the packet, the Layer 3 Switch discards the packet and performs no further processing, except generating a Syslog entry and SNMP message, if logging is enabled for the filter.
2. If the packet is not denied at the incoming interface, the Layer 3 Switch looks in the session table for an entry that has the same source IP address and TCP or UDP port as the packet. If the session table contains a matching entry, the Layer 3 Switch immediately forwards the packet, by addressing it to the destination IP address and TCP or UDP port listed in the session table entry and sending the packet to a queue on the outgoing port(s) listed in the session table. The Layer 3 Switch selects the queue based on the Quality of Service (QoS) level associated with the session table entry.
3. If the session table does not contain an entry that matches the packet's source address and TCP or UDP port, the Layer 3 Switch looks in the IP forwarding cache for an entry that matches the packet's destination IP address. If the forwarding cache contains a matching entry, the Layer 3 Switch forwards the packet to the IP address in the entry. The Layer 3 Switch sends the packet to a queue on the outgoing port(s) listed in the forwarding cache. The Layer 3 Switch selects the queue based on the Quality of Service (QoS) level associated with the forwarding cache entry.
4. If the IP forwarding cache does not have an entry for the packet, the Layer 3 Switch checks the IP route table for a route to the packet's destination. If the IP route table has a route, the Layer 3 Switch makes an entry in the session table or the forwarding cache, and sends the route to a queue on the outgoing port(s).
    - If the running-config contains an IP access policy for the packet, the software makes an entry in the session table. The Layer 3 Switch uses the new session table entry to forward subsequent packets from the same source to the same destination.
    - If the running-config does not contain an IP access policy for the packet, the software creates a new entry in the forwarding cache. The Layer 3 Switch uses the new cache entry to forward subsequent packets to the same destination.

The following sections describe the IP tables and caches:

- ARP cache and static ARP table
- IP route table
- IP forwarding cache
- IP session table

The software enables you to display these tables. You also can change the capacity of the tables on an individual basis if needed by changing the memory allocation for the table.

ARP Cache and Static ARP Table

The ARP cache contains entries that map IP addresses to MAC addresses. Generally, the entries are for devices that are directly attached to the Layer 3 Switch.

An exception is an ARP entry for an interface-based static IP route that goes to a destination that is one or more router hops away. For this type of entry, the MAC address is either the destination device's MAC address or the MAC address of the router interface that answered an ARP request on behalf of the device, using proxy ARP.

ARP Cache

The ARP cache can contain dynamic (learned) entries and static (user-configured) entries. The software places a dynamic entry in the ARP cache when the Layer 3 Switch learns a device's MAC address from an ARP request or ARP reply from the device.

1.The filter can be an Access Control List (ACL) or an IP access policy.

The software can learn an entry when the Layer 2 Switch or Layer 3 Switch receives an ARP request from another IP forwarding device or an ARP reply. Here is an example of a dynamic entry:

```
      IP Address          MAC Address         Type        Age     Port
1     207.95.6.102        0800.5afc.ea21      Dynamic     0         6
```

Each entry contains the destination device's IP address and MAC address.

Static ARP Table

In addition to the ARP cache, Layer 3 Switches have a static ARP table. Entries in the static ARP table are user-configured. You can add entries to the static ARP table regardless of whether the device the entry is for is connected to the Layer 3 Switch.

NOTE: The Layer 3 Switches have a static ARP table but Layer 2 Switches do not.

The software places an entry from the static ARP table into the ARP cache when the entry's interface comes up.

Here is an example of a static ARP entry:

```
  Index   IP Address          MAC Address         Port
  1       207.95.6.111        0800.093b.d210      1/1
```

Each entry lists the information you specified when you created the entry.

To display ARP entries, see the following:

- "Displaying the ARP Cache" on page 16-63 – Layer 3 Switch
- "Displaying the Static ARP Table" on page 16-65 – Layer 3 Switch only
- "Displaying ARP Entries" on page 16-74 – Layer 2 Switch

To configure other ARP parameters, see the following:

- "Configuring ARP Parameters" on page 16-25 – Layer 3 Switch only

To increase the size of the ARP cache and static ARP table, see the following:

- For dynamic entries, see the section "Displaying and Modifying System Parameter Default Settings" on page 4-8. The ip-arp parameter controls the ARP cache size.
- Static entries, "Changing the Maximum Number of Entries the Static ARP Table Can Hold" on page 16-28 – Layer 3 Switches only. The ip-static-arp parameter controls the static ARP table size.

IP Route Table

The IP route table contains paths to IP destinations.

NOTE: Layer 2 Switches do not have an IP route table. A Layer 2 Switch sends all packets addressed to another sub-net to the default gateway, which you specify when you configure the basic IP information on the Layer 2 Switch.

The IP route table can receive the paths from the following sources:

- A directly-connected destination, which means there are no router hops to the destination
- A static IP route, which is a user-configured route
- A route learned through RIP
- A route learned through OSPF
- A route learned through BGP4

The IP route table contains the best path to a destination.

- When the software receives paths from more than one of the sources listed above, the software compares the

administrative distance of each path and selects the path with the lowest administrative distance. The administrative distance is a protocol-independent value from 1 – 255.

- When the software receives two or more best paths from the same source and the paths have the same metric (cost), the software can load share traffic among the paths based on destination host or network address (based on the configuration and the Layer 3 Switch model).

Here is an example of an entry in the IP route table:

```
Destination       NetMask          Gateway          Port   Cost   Type
1.1.0.0           255.255.0.0      99.1.1.2         1/1    2      R
```

Each IP route table entry contains the destination's IP address and sub-net mask and the IP address of the next-hop router interface to the destination. Each entry also indicates the port attached to the destination or the next-hop to the destination, the route's IP metric (cost), and the type. The type indicates how the IP route table received the route.

To display the IP route table, see the following:

- "Displaying the IP Route Table" on page 16-67 – Layer 3 Switch only

To configure a static IP route, see the following:

- "Configuring Static Routes" on page 16-32 – Layer 3 Switch only

To clear a route from the IP route table, see the following:

- "Clearing IP Routes" on page 16-69 – Layer 3 Switch only

To increase the size of the IP route table for learned and static routes, see the section "Displaying and Modifying System Parameter Default Settings" on page 4-8.

- For learned routes, modify the ip-route parameter.
- For static routes, modify the ip-static-route parameter.

IP Forwarding Cache

The IP forwarding cache provides a fast-path mechanism for forwarding IP packets. The cache contains entries for IP destinations. When a Foundry Layer 3 Switch has completed processing and addressing for a packet and is ready to forward the packet, the device checks the IP forwarding cache for an entry to the packet's destination.

- If the cache contains an entry with the destination IP address, the device uses the information in the entry to forward the packet out the ports listed in the entry. The destination IP address is the address of the packet's final destination. The port numbers are the ports through which the destination can be reached.
- If the cache does not contain an entry and the traffic does not qualify for an entry in the session table instead, the software can create an entry in the forwarding cache.

Each entry in the IP forwarding cache has an age timer. If the entry remains unused for ten minutes, the software removes the entry. The age timer is not configurable.

Here is an example of an entry in the IP forwarding cache:

```
      IP Address       Next Hop         MAC              Type  Port  Vlan  Pri
1     192.168.1.11     DIRECT           0000.0000.0000   PU    n/a         0
```

Each IP forwarding cache entry contains the IP address of the destination, and the IP address and MAC address of the next-hop router interface to the destination. If the destination is actually an interface configured on the Layer 3 Switch itself, as shown here, then next-hop information indicates this. The port through which the destination is reached is also listed, as well as the VLAN and Layer 4 QoS priority associated with the destination if applicable.

To display the IP forwarding cache, see "Displaying the Forwarding Cache" on page 16-66.

NOTE: You cannot add static entries to the IP forwarding cache, although you can increase the number of entries the cache can contain. See the section “Displaying and Modifying System Parameter Default Settings” on page 4-8.

To increase the size of the IP forwarding cache, see the section “Displaying and Modifying System Parameter Default Settings” on page 4-8.

Layer 4 Session Table

The Layer 4 session provides a fast path for forwarding packets. A ***session*** is an entry that contains complete Layer 3 and Layer 4 information for a flow of traffic. Layer 3 information includes the source and destination IP addresses. Layer 4 information includes the source and destination TCP and UDP ports. For comparison, the IP forwarding cache contains the Layer 3 destination address but does not contain the other source and destination address information of a Layer 4 session table entry.

The Layer 2 Switch or Layer 3 Switch selects the session table instead of the IP forwarding table for fast-path forwarding for the following features:

- Layer 4 Quality-of-Service (QoS) policies
- IP access policies

To increase the size of the session table, see the section “Displaying and Modifying System Parameter Default Settings” on page 4-8. The ip-qos-session parameter controls the size of the session table.

IP Route Exchange Protocols

Foundry Layer 3 Switches support the following IP route exchange protocols:

- Routing Information Protocol (RIP)
- Open Shortest Path First (OSPF)
- Border Gateway Protocol version 4 (BGP4)

All these protocols provide routes to the IP route table. You can use one or more of these protocols, in any combination. The protocols are disabled by default. For configuration information, see the following:

- “Configuring RIP” on page 17-1
- “Configuring OSPF” on page 20-1
- “Configuring BGP4” on page 21-1

IP Multicast Protocols

Foundry Layer 3 Switches also support the following Internet Group Membership Protocol (IGMP) based IP multicast protocols:

- Protocol Independent Multicast – Dense mode (PIM-DM)
- Protocol Independent Multicast – Sparse mode (PIM-SM)
- Distance Vector Multicast Routing Protocol (DVMRP)

For configuration information, see “Configuring IP Multicast Protocols” on page 19-1.

NOTE: Foundry Layer 2 Switches support IGMP and can forward IP multicast packets. See the chapter “Configuring IP Multicast Traffic Reduction” .

IP Interface Redundancy Protocols

You can configure a Foundry Layer 3 Switch to back up an IP interface configured on another Foundry Layer 3 Switch. If the link for the backed up interface becomes unavailable, the other Layer 3 Switch can continue service for the interface. This feature is especially useful for providing a backup to a network's default gateway.

Foundry Layer 3 Switches support the following IP interface redundancy protocols:

- Virtual Router Redundancy Protocol (VRRP) – A standard router redundancy protocol based on RFC 2338. You can use VRRP to configure Foundry Layer 3 Switches and third-party routers to back up IP interfaces on other Foundry Layer 3 Switches or third-party routers.
- Virtual Router Redundancy Protocol Extended (VRRPE) – A Foundry extension to standard VRRP that adds additional features and overcomes limitations in standard VRRP. You can use VRRPE only on Foundry Layer 3 Switches.

For configuration information, see the following:

- Virtual Router Redundancy Protocol Extended (VRRPE) – see "Configuring VRRP and VRRPE" on page 22-1.
- Virtual Router Redundancy Protocol (VRRP) – see "Configuring VRRP and VRRPE" on page 22-1.

Access Control Lists and IP Access Policies

Foundry Layer 3 Switches provide two mechanisms for filtering IP traffic:

- Access Control Lists (ACLs)
- IP access policies

Both methods allow you to filter packets based on Layer 3 and Layer 4 source and destination information.

ACLs also provide great flexibility by providing the input to various other filtering mechanisms such as route maps, which are used by BGP4.

IP access policies allow you to configure QoS based on sessions (Layer 4 traffic flows).

Only one of these filtering mechanisms can be enabled on a Foundry device at a time. Foundry devices can store forwarding information for both methods of filtering in the session table.

For configuration information, see the following:

- "Rule-Based IP Access Control Lists (ACLs)" on page 12-1
- "Policies and Filters" on page C-1

Basic IP Parameters and Defaults – Layer 3 Switches

IP is enabled by default. The following IP-based protocols are all disabled by default:

- Routing protocols
  - Routing Information Protocol (RIP) – see "Configuring RIP" on page 17-1
  - Open Shortest Path First (OSPF) – see "Configuring OSPF" on page 20-1
  - Border Gateway Protocol version 4 (BGP4) – see "Configuring BGP4" on page 21-1
- Multicast protocols
  - Internet Group Membership Protocol (IGMP) – see "Changing Global IP Multicast Parameters" on page 19-3
  - Protocol Independent Multicast Dense (PIM-DM) – see "PIM Dense" on page 19-6
  - Protocol Independent Multicast Sparse (PIM-SM) – see "PIM Sparse" on page 19-13
  - Distance Vector Multicast Routing Protocol (DVMRP) – see "DVMRP Overview" on page 19-32

- Router redundancy protocols
  - Virtual Router Redundancy Protocol Extended (VRRPE) – see “Configuring VRRP and VRRPE” on page 22-1.
  - Virtual Router Redundancy Protocol (VRRP) – see “Configuring VRRP and VRRPE” on page 22-1.

The following tables list the Layer 3 Switch IP parameters, their default values, and where to find configuration information.

NOTE: For information about parameters in other protocols based on IP, such as RIP, OSPF, and so on, see the configuration chapters for those protocols.

When Parameter Changes Take Effect

Most IP parameters described in this chapter are dynamic. They take effect immediately, as soon as you enter the CLI command or select the Web management interface option. You can verify that a dynamic change has taken effect by displaying the running-config. To display the running-config, enter the **show running-config** or **write terminal** command at any CLI prompt. (You cannot display the running-config from the Web management interface.)

To save a configuration change permanently so that the change remains in effect following a system reset or software reload, save the change to the startup-config file.

- To save configuration changes to the startup-config file, enter the **write memory** command from the Privileged EXEC level of any configuration level of the CLI.
- To save the configuration changes using the Web management interface, select the Save link at the bottom of the dialog. Select Yes when prompted to save the configuration change to the startup-config file on the device’s flash memory. You also can access the dialog for saving configuration changes by clicking on Command in the tree view, then clicking on Save to Flash.

Changes to memory allocation require you to reload the software after you save the changes to the startup-config file. When reloading the software is required to complete a configuration change described in this chapter, the procedure that describes the configuration change includes a step for reloading the software.

IP Global Parameters – Layer 3 Switches

Table 16.2 lists the IP global parameters for Layer 3 Switches.

Table 16.2: IP Global Parameters – Layer 3 Switches

Parameter	Description	Default	See page...
IP state	The Internet Protocol, version 4	Enabled **Note**: You cannot disable IP.	n/a
IP address and mask notation	Format for displaying an IP address and its network mask information. You can enable one of the following: • Class-based format; example: 192.168.1.1 255.255.255.0 • Classless Interdomain Routing (CIDR) format; example: 192.168.1.1/24	Class-based **Note**: Changing this parameter affects the display of IP addresses, but you can enter addresses in either format regardless of the display setting.	16-57

Table 16.2: IP Global Parameters – Layer 3 Switches (Continued)

Parameter	Description	Default	See page...
Router ID	The value that routers use to identify themselves to other routers when exchanging route information. OSPF and BGP4 use router IDs to identify routers. RIP does not use the router ID.	The IP address configured on the lowest-numbered loopback interface. If no loopback interface is configured, then the lowest-numbered IP address configured on the device.	16-23
Maximum Transmission Unit (MTU)	The maximum length an Ethernet packet can be without being fragmented.	1500 bytes for Ethernet II encapsulation 1492 bytes for SNAP encapsulation	16-21
Address Resolution Protocol (ARP)	A standard IP mechanism that routers use to learn the Media Access Control (MAC) address of a device on the network. The router sends the IP address of a device in the ARP request and receives the device's MAC address in an ARP reply.	Enabled	16-25
ARP rate limiting	Lets you specify a maximum number of ARP packets the device will accept each second. If the device receives more ARP packets than you specify, the device drops additional ARP packets for the remainder of the one-second interval.	Disabled	16-26
ARP age	The amount of time the device keeps a MAC address learned through ARP in the device's ARP cache. The device resets the timer to zero each time the ARP entry is refreshed and removes the entry if the timer reaches the ARP age. **Note**: You also can change the ARP age on an individual interface basis. See Table 16.3 on page 16-13.	Ten minutes	16-27
Proxy ARP	An IP mechanism a router can use to answer an ARP request on behalf of a host, by replying with the router's own MAC address instead of the host's.	Disabled	16-27
Static ARP entries	An ARP entry you place in the static ARP table. Static entries do not age out.	No entries	16-28
Time to Live (TTL)	The maximum number of routers (hops) through which a packet can pass before being discarded. Each router decreases a packet's TTL by 1 before forwarding the packet. If decreasing the TTL causes the TTL to be 0, the router drops the packet instead of forwarding it.	64 hops	16-29

Table 16.2: IP Global Parameters – Layer 3 Switches (Continued)

Parameter	Description	Default	See page...
Directed broadcast forwarding	A directed broadcast is a packet containing all ones (or in some cases, all zeros) in the host portion of the destination IP address. When a router forwards such a broadcast, it sends a copy of the packet out each of its enabled IP interfaces. **Note**: You also can enable or disable this parameter on an individual interface basis. See Table 16.3 on page 16-13.	Disabled	16-29
Directed broadcast mode	The packet format the router treats as a directed broadcast. The following formats can be directed broadcast: • All ones in the host portion of the packet's destination address. • All zeroes in the host portion of the packet's destination address.	All ones **Note**: If you enable all-zeroes directed broadcasts, all-ones directed broadcasts remain enabled.	16-30
Source-routed packet forwarding	A source-routed packet contains a list of IP addresses through which the packet must pass to reach its destination.	Enabled	16-30
Internet Control Message Protocol (ICMP) messages	The Foundry Layer 3 Switch can send the following types of ICMP messages: • Echo messages (ping messages) • Destination Unreachable messages	Enabled	16-31
ICMP Router Discovery Protocol (IRDP)	An IP protocol a router can use to advertise the IP addresses of its router interfaces to directly attached hosts. You can enable or disable the protocol, and change the following protocol parameters: • Forwarding method (broadcast or multicast) • Hold time • Maximum advertisement interval • Minimum advertisement interval • Router preference level **Note**: You also can enable or disable IRDP and configure the parameters on an individual interface basis. See Table 16.3 on page 16-13.	Disabled	16-44
Reverse ARP (RARP)	An IP mechanism a host can use to request an IP address from a directly attached router when the host boots.	Enabled	16-45
Static RARP entries	An IP address you place in the RARP table for RARP requests from hosts. **Note**: You must enter the RARP entries manually. The Layer 3 Switch does not have a mechanism for learning or dynamically generating RARP entries.	No entries	16-46

Table 16.2: IP Global Parameters – Layer 3 Switches (Continued)

Parameter	Description	Default	See page...
Maximum BootP relay hops	The maximum number of hops away a BootP server can be located from a router and still be used by the router's clients for network booting.	Four	16-50
Domain name for Domain Name Server (DNS) resolver	A domain name (example: foundry.router.com) you can use in place of an IP address for certain operations such as IP pings, trace routes, and Telnet management connections to the router.	None configured	16-19
DNS default gateway addresses	A list of gateways attached to the router through which clients attached to the router can reach DNSs.	None configured	16-19
IP load sharing	AFoundry feature that enables the router to balance traffic to a specific destination across multiple equal-cost paths. Load sharing uses a simple round-robin mechanism and is based on destination address. **Note**: Load sharing is sometimes called Equal Cost Multi Path (ECMP).	Enabled	16-41
Maximum IP load sharing paths	The maximum number of equal-cost paths across which the Layer 3 Switch is allowed to distribute traffic.	Four	16-43
Origination of default routes	You can enable a router to originate default routes for the following route exchange protocols, on an individual protocol basis: • RIP • OSPF • BGP4	Disabled	17-8 20-32 21-26
Default network route	The router uses the default network route if the IP route table does not contain a route to the destination and also does not contain an explicit default route (0.0.0.0 0.0.0.0 or 0.0.0.0/0).	None configured	16-39
Static route	An IP route you place in the IP route table.	No entries	16-32
Source interface	The IP address the router uses as the source address for Telnet, RADIUS, or TACACS/TACACS+ packets originated by the router. The router can select the source address based on either of the following: • The lowest-numbered IP address on the interface the packet is sent on. • The lowest-numbered IP address on a specific interface. The address is used as the source for all packets of the specified type regardless of interface the packet is sent on.	The lowest-numbered IP address on the interface the packet is sent on.	16-24

IP Interface Parameters – Layer 3 Switches

Table 16.3 lists the interface-level IP parameters for Layer 3 Switches.

Table 16.3: IP Interface Parameters – Layer 3 Switches

Parameter	Description	Default	See page...
IP state	The Internet Protocol, version 4	Enabled **Note**: You cannot disable IP.	n/a
IP address	A Layer 3 network interface address **Note**: Layer 2 Switches have a single IP address used for management access to the entire device. Layer 3 Switches have separate IP addresses on individual interfaces.	None configured[a]	16-17
Encapsulation type	The format of the packets in which the router encapsulates IP datagrams. The encapsulation format can be one of the following: • Ethernet II • SNAP	Ethernet II	16-21
Maximum Transmission Unit (MTU)	The maximum length (number of bytes) of an encapsulated IP datagram the router can forward.	1500 for Ethernet II encapsulated packets 1492 for SNAP encapsulated packets	16-22
ARP age	Locally overrides the global setting. See Table 16.2 on page 16-9.	Ten minutes	16-27
Metric	A numeric cost the router adds to RIP routes learned on the interface. This parameter applies only to RIP routes.	1 (one)	17-4
Directed broadcast forwarding	Locally overrides the global setting. See Table 16.2 on page 16-9.	Disabled	16-29
ICMP Router Discovery Protocol (IRDP)	Locally overrides the global IRDP settings. See Table 16.2 on page 16-9.	Disabled	16-45
DHCP gateway stamp	The router can assist DHCP/BootP Discovery packets from one sub-net to reach DHCP/BootP servers on a different sub-net by placing the IP address of the router interface that receives the request in the request packet's Gateway field. You can override the default and specify the IP address to use for the Gateway field in the packets. **Note**: UDP broadcast forwarding for client DHCP/BootP requests (bootpc) must be enabled and you must configure an IP helper address (the server's IP address or a directed broadcast to the server's sub-net) on the port connected to the client.	The lowest-numbered IP address on the interface that receives the request	16-50

Table 16.3: IP Interface Parameters – Layer 3 Switches (Continued)

Parameter	Description	Default	See page...
UDP broadcast forwarding	The router can forward UDP broadcast packets for UDP applications such as BootP. By forwarding the UDP broadcasts, the router enables clients on one sub-net to find servers attached to other sub-nets. **Note**: To completely enable a client's UDP application request to find a server on another sub-net, you must configure an IP helper address consisting of the server's IP address or the directed broadcast address for the sub-net that contains the server. See the next row.	The router helps forward broadcasts for the following UDP application protocols: • bootps • dns • netbios-dgm • netbios-ns • tacacs • tftp • time	16-48
IP helper address	The IP address of a UDP application server (such as a BootP or DHCP server) or a directed broadcast address. IP helper addresses allow the router to forward requests for certain UDP applications from a client on one sub-net to a server on another sub-net.	None configured	16-49

a.Some devices have a factory default, such as 209.157.22.154, used for troubleshooting during installation. For Layer 3 Switches, the address is on module 1 port 1 (or 1/1). NetIron Internet Backbone routers do not have a default IP address.

Basic IP Parameters and Defaults – Layer 2 Switches

IP is enabled by default. The following tables list the Layer 2 Switch IP parameters, their default values, and where to find configuration information.

NOTE: Foundry Layer 2 Switches also provide IP multicast forwarding, which is enabled by default. For information about this feature, see the chapter "Configuring IP Multicast Traffic Reduction" on page 18-1.

IP Global Parameters – Layer 2 Switches

Table 16.4 lists the IP global parameters for Layer 2 Switches.

Table 16.4: IP Global Parameters – Layer 2 Switches

Parameter	Description	Default	See page...
IP address and mask notation	Format for displaying an IP address and its network mask information. You can enable one of the following: • Class-based format; example: 192.168.1.1 255.255.255.0 • Classless Interdomain Routing (CIDR) format; example: 192.168.1.1/24	Class-based **Note**: Changing this parameter affects the display of IP addresses, but you can enter addresses in either format regardless of the display setting.	16-57
IP address	A Layer 3 network interface address **Note**: Layer 2 Switches have a single IP address used for management access to the entire device. Layer 3 Switches have separate IP addresses on individual interfaces.	None configured[a]	16-51
Default gateway	The IP address of a locally attached router (or a router attached to the Layer 2 Switch by bridges or other Layer 2 Switches). The Layer 2 Switch and clients attached to it use the default gateway to communicate with devices on other sub-nets.	None configured	16-51
Address Resolution Protocol (ARP)	A standard IP mechanism that networking devices use to learn the Media Access Control (MAC) address of another device on the network. The Layer 2 Switch sends the IP address of a device in the ARP request and receives the device's MAC address in an ARP reply.	Enabled **Note**: You cannot disable ARP.	n/a
ARP age	The amount of time the device keeps a MAC address learned through ARP in the device's ARP cache. The device resets the timer to zero each time the ARP entry is refreshed and removes the entry if the timer reaches the ARP age.	Ten minutes **Note**: You cannot change the ARP age on Layer 2 Switches.	n/a

Table 16.4: IP Global Parameters – Layer 2 Switches (Continued)

Parameter	Description	Default	See page...
Time to Live (TTL)	The maximum number of routers (hops) through which a packet can pass before being discarded. Each router decreases a packet's TTL by 1 before forwarding the packet. If decreasing the TTL causes the TTL to be 0, the router drops the packet instead of forwarding it.	64 hops	16-53
Domain name for Domain Name Server (DNS) resolver	A domain name (example: foundry.router.com) you can use in place of an IP address for certain operations such as IP pings, trace routes, and Telnet management connections to the router.	None configured	16-51
DNS default gateway addresses	A list of gateways attached to the router through which clients attached to the router can reach DNSs.	None configured	16-51
Source interface	The IP address the Layer 2 Switch uses as the source address for Telnet, RADIUS, or TACACS/TACACS+ packets originated by the router. The Layer 2 Switch uses its management IP address as the source address for these packets.	The management IP address of the Layer 2 Switch. **Note**: This parameter is not configurable on Layer 2 Switches.	n/a
DHCP gateway stamp	The device can assist DHCP/BootP Discovery packets from one sub-net to reach DHCP/BootP servers on a different sub-net by placing the IP address of the router interface that forwards the packet in the packet's Gateway field. You can specify up to 32 gateway lists. A gateway list contains up to eight gateway IP addresses. You activate DHCP assistance by associating a gateway list with a port. When you configure multiple IP addresses in a gateway list, the Layer 2 Switch inserts the addresses into the DHCP Discovery packets in a round robin fashion.	None configured	16-56

a.Some devices have a factory default, such as 209.157.22.154, used for troubleshooting during installation. For Layer 3 Switches, the address is on port 1 (or 1/1). NetIron Internet Backbone routers do not have a default IP address.

Interface IP Parameters – Layer 2 Switches

Table 16.5 lists the interface-level IP parameters for Layer 2 Switches.

Table 16.5: Interface IP Parameters – Layer 2 Switches

Parameter	Description	Default	See page...
DHCP gateway stamp	You can configure a list of DHCP stamp addresses for a port. When the port receives a DHCP/BootP Discovery packet from a client, the port places the IP address(es) in the gateway list into the packet's Gateway field.	None configured	16-56

Configuring IP Parameters – Layer 3 Switches

The following sections describe how to configure IP parameters. Some parameters can be configured globally while others can be configured on individual interfaces. Some parameters can be configured globally and overridden for individual interfaces.

NOTE: This section describes how to configure IP parameters for Layer 3 Switches. For IP configuration information for Layer 2 Switches, see "Configuring IP Parameters – Layer 2 Switches" on page 16-51.

Configuring IP Addresses

You can configure an IP address on the following types of Layer 3 Switch interfaces:

- Ethernet port
- Virtual routing interface (also called a Virtual Ethernet or "VE")
- Loopback interface

By default, you can configure up to 24 IP addresses on each interface. On Stackable Layer 3 Switches, you can increase this amount to up to 64 IP sub-net addresses per port by increasing the size of the subnet-per-interface table. See the section "Displaying and Modifying System Parameter Default Settings" on page 4-8.

NOTE: Once you configure a virtual routing interface on a VLAN, you cannot configure Layer 3 interface parameters on individual ports in the VLAN. Instead, you must configure the parameters on the virtual routing interface itself.

Foundry devices support both classical IP network masks (Class A, B, and C sub-net masks, and so on) and Classless Interdomain Routing (CIDR) network prefix masks.

- To enter a classical network mask, enter the mask in IP address format. For example, enter "209.157.22.99 255.255.255.0" for an IP address with a Class-C sub-net mask.
- To enter a prefix network mask, enter a forward slash (/) and the number of bits in the mask immediately after the IP address. For example, enter "209.157.22.99/24" for an IP address that has a network mask with 24 significant bits (ones).

By default, the CLI displays network masks in classical IP address format (example: 255.255.255.0). You can change the display to prefix format. See "Changing the Network Mask Display to Prefix Format" on page 16-57.

Assigning an IP Address to an Ethernet Port

To assign an IP address to port 1/1, enter the following commands:

```
FastIron SuperX Router(config)# interface ethernet 1/1
```



```
FastIron SuperX Router(config-if-1/1)# ip address 192.45.6.1 255.255.255.0
```

NOTE: You also can enter the IP address and mask in CIDR format, as follows:

```
FastIron SuperX Router(config-if-1/1)# ip address 192.45.6.1/24
```

Syntax: [no] ip address <ip-addr> <ip-mask> [ospf-ignore | ospf-passive | secondary]

or

Syntax: [no] ip address <ip-addr>/<mask-bits> [ospf-ignore | ospf-passive | secondary]

The **ospf-ignore** | **ospf-passive** parameters modify the Layer 3 Switch defaults for adjacency formation and interface advertisement. Use one of these parameters if you are configuring multiple IP sub-net addresses on the interface but you want to prevent OSPF from running on some of the sub-nets.

- **ospf-passive** – This option disables adjacency formation with OSPF neighbors. By default, when OSPF is enabled on an interface, the software forms OSPF router adjacencies between each primary IP address on the interface and the OSPF neighbor attached to the interface.
- **ospf-ignore** – This option disables OSPF adjacency formation and also disables advertisement of the interface into OSPF. The sub-net is completely ignored by OSPF.

NOTE: The **ospf-passive** option disables adjacency formation but does not disable advertisement of the interface into OSPF. To disable advertisement in addition to disabling adjacency formation, you must use the **ospf-ignore** option.

Use the **secondary** parameter if you have already configured an IP address within the same sub-net on the interface.

NOTE: When you configure more than one address in the same sub-net, all but the first address are secondary addresses and do not form OSPF adjacencies.

Assigning an IP Address to a Loopback Interface

Loopback interfaces are always up, regardless of the states of physical interfaces. They can add stability to the network because they are not subject to route flap problems that can occur due to unstable links between a Layer 3 Switch and other devices. You can configure up to eight loopback interfaces on a Chassis Layer 3 Switch and up to four loopback interfaces on a Stackable Layer 3 Switch.

You can add up to 24 IP addresses to each loopback interface.

NOTE: If you configure the Foundry Layer 3 Switch to use a loopback interface to communicate with a BGP4 neighbor, you also must configure a loopback interface on the neighbor and configure the neighbor to use that loopback interface to communicate with the Foundry Layer 3 Switch. See “Adding a Loopback Interface” on page 21-11.

To add a loopback interface, enter commands such as those shown in the following example:

```
FESX424 Router(config-bgp-router)# exit
FESX424 Router(config)# int loopback 1
FESX424 Router(config-lbif-1)# ip address 10.0.0.1/24
```

Syntax: interface loopback <num>

The <num> parameter specifies the virtual interface number. You can specify from 1 to the maximum number of virtual interfaces supported on the device. To display the maximum number of virtual interfaces supported on the device, enter the **show default values** command. The maximum is listed in the System Parameters section, in the Current column of the virtual-interface row.

See the syntax description in “Assigning an IP Address to an Ethernet Port” on page 16-17.

Assigning an IP Address to a Virtual Interface

A virtual interface is a logical port associated with a Layer 3 Virtual LAN (VLAN) configured on a Layer 3 Switch. You can configure routing parameters on the virtual interface to enable the Layer 3 Switch to route protocol traffic from one Layer 3 VLAN to the other, without using an external router.[1]

You can configure IP routing interface parameters on a virtual interface. This section describes how to configure an IP address on a virtual interface. Other sections in this chapter that describe how to configure interface parameters also apply to virtual interfaces.

NOTE: The Layer 3 Switch uses the lowest MAC address on the device (the MAC address of port 1 or 1/1) as the MAC address for all ports within all virtual interfaces you configure on the device.

USING THE CLI

To add a virtual interface to a VLAN and configure an IP address on the interface, enter commands such as the following:

```
FESX424 Router(config)# vlan 2 name IP-Subnet_1.1.2.0/24
FESX424 Router(config-vlan-2)# untag e1 to 4
FESX424 Router(config-vlan-2)# router-interface ve1
FESX424 Router(config-vlan-2)# interface ve1
FESX424 Router(config-vif-1)# ip address 1.1.2.1/24
```

The first two commands in this example create a Layer 3 protocol-based VLAN name “IP-Subnet_1.1.2.0/24” and add a range of untagged ports to the VLAN. The **router-interface** command creates virtual interface 1 as the routing interface for the VLAN. The last two commands change to the interface configuration level for the virtual interface and assign an IP address to the interface.

Syntax: router-interface ve <num>

Syntax: interface ve <num>

See the syntax description in “Assigning an IP Address to an Ethernet Port” on page 16-17.

Deleting an IP Address

To delete an IP address, enter a command such as the following:

```
FESX424 Router(config-if-e1000-1)# no ip address 1.1.2.1
```

This command deletes IP address 1.1.2.1. You do not need to enter the subnet mask.

To delete all IP addresses from an interface, enter the following command:

```
FESX424 Router(config-if-e1000-1)# no ip address *
```

Syntax: no ip address <ip-addr> | *

Configuring Domain Name Server (DNS) Resolver

The Domain Name Server (DNS) resolver feature lets you use a host name to perform Telnet, ping, and traceroute commands. You can also define a DNS domain on a Foundry Layer 2 Switch or Layer 3 Switch and thereby recognize all hosts within that domain. After you define a domain name, the Foundry Layer 2 Switch or Layer 3 Switch automatically appends the appropriate domain to the host and forwards it to the domain name server.

For example, if the domain “newyork.com” is defined on a Foundry Layer 2 Switch or Layer 3 Switch and you want to initiate a ping to host “NYC01” on that domain, you need to reference only the host name in the command instead of the host name and its domain name. For example, you could enter either of the following commands to initiate the ping:

```
FESX424 Router# ping nyc01
FESX424 Router# ping nyc01.newyork.com
```

1.Foundry’s feature that allows routing between VLANs within the same device, without the need for external routers, is called Integrated Switch Routing (ISR).

Defining a DNS Entry

You can define up to four DNS servers for each DNS entry. The first entry serves as the primary default address. If a query to the primary address fails to be resolved after three attempts, the next gateway address is queried (also up to three times). This process continues for each defined gateway address until the query is resolved. The order in which the default gateway addresses are polled is the same as the order in which you enter them.

Suppose you want to define the domain name of newyork.com on a Layer 3 Switch and then define four possible default DNS gateway addresses. To do so, enter the following commands:

```
FESX424 Router(config)# ip dns domain-name newyork.com
FESX424 Router(config)# ip dns server-address 209.157.22.199 205.96.7.15 208.95.7.25
201.98.7.15
```

Syntax: ip dns server-address <ip-addr> [<ip-addr>] [<ip-addr>] [<ip-addr>]

In this example, the first IP address in the **ip dns server-address...** command becomes the primary gateway address and all others are secondary addresses. Because IP address 201.98.7.15 is the last address listed, it is also the last address consulted to resolve a query.

Using a DNS Name To Initiate a Trace Route

Suppose you want to trace the route from a Foundry Layer 3 Switch to a remote server identified as NYC02 on domain newyork.com. Because the newyork.com domain is already defined on the Layer 3 Switch, you need to enter only the host name, NYC02, as noted below.

```
FESX424 Router# traceroute nyc02
```

Syntax: traceroute <host-ip-addr> [maxttl <value>] [minttl <value>] [numeric] [timeout <value>] [source-ip <ip addr>]

The only required parameter is the IP address of the host at the other end of the route. See the *Foundry Switch and Router Command Line Interface Reference* for information about the parameters.

After you enter the command, a message indicating that the DNS query is in process and the current gateway address (IP address of the domain name server) being queried appear on the screen:

```
Type Control-c to abort
Sending DNS Query to 209.157.22.199
Tracing Route to IP node 209.157.22.80
To ABORT Trace Route, Please use stop-traceroute command.
 Traced route to target IP node 209.157.22.80:
   IP Address          Round Trip Time1    Round Trip Time2
  207.95.6.30         93 msec             121 msec
```

NOTE: In the above example, 209.157.22.199 is the IP address of the domain name server (default DNS gateway address), and 209.157.22.80 represents the IP address of the NYC02 host.

Configuring Packet Parameters

You can configure the following packet parameters on Layer 3 Switches. These parameters control how the Layer 3 Switch sends IP packets to other devices on an Ethernet network. The Layer 3 Switch always places IP packets into Ethernet packets to forward them on an Ethernet port.

- Encapsulation type – The format for the Layer 2 packets within which the Layer 3 Switch sends IP packets.
- Maximum Transmission Unit (MTU) – The maximum length of IP packet that a Layer 2 packet can contain. IP packets that are longer than the MTU are fragmented and sent in multiple Layer 2 packets. You can change the MTU globally or an individual ports.
    - Global MTU – The default MTU value depends on the encapsulation type on a port and is 1500 bytes for Ethernet II encapsulation and 1492 bytes for SNAP encapsulation.

- Port MTU – A port's default MTU depends on the encapsulation type enabled on the port.

Changing the Encapsulation Type

The Layer 3 Switch encapsulates IP packets into Layer 2 packets, to send the IP packets on the network. (A Layer 2 packet is also called a MAC layer packet or an Ethernet frame.) The source address of a Layer 2 packet is the MAC address of the Layer 3 Switch interface sending the packet. The destination address can be one of the following:

- The MAC address of the IP packet's destination. In this case, the destination device is directly connected to the Layer 3 Switch.
- The MAC address of the next-hop gateway toward the packet's destination.
- An Ethernet broadcast address.

The entire IP packet, including the source and destination address and other control information and the data, is placed in the data portion of the Layer 2 packet. Typically, an Ethernet network uses one of two different formats of Layer 2 packet:

- Ethernet II
- Ethernet SNAP (also called IEEE 802.3)

The control portions of these packets differ slightly. All IP devices on an Ethernet network must use the same format. Foundry Layer 3 Switches use Ethernet II by default. You can change the IP encapsulation to Ethernet SNAP on individual ports if needed.

NOTE: All devices connected to the Layer 3 Switch port must use the same encapsulation type.

To change the IP encapsulation type on interface 5 to Ethernet SNAP, enter the following commands:

```
FESX424 Router(config)# int e 5
FESX424 Router(config-if-e1000-5)# ip encapsulation snap
```

Syntax: ip encapsulation snap | ethernet_ii

Changing the Maximum Transmission Unit (MTU)

The Maximum Transmission Unit (MTU) is the maximum length of IP packet that a Layer 2 packet can contain. IP packets that are longer than the MTU are fragmented and sent in multiple Layer 2 packets. You can change the MTU globally or on individual ports.

The default MTU is 1500 bytes for Ethernet II packets and 1492 for Ethernet SNAP packets.

MTU Enhancements

Foundry devices contain the following enhancements to jumbo packet support:

- Hardware forwarding of Layer 3 jumbo packets – Layer 3 IP unicast jumbo packets received on a port that supports the frame's MTU size and forwarded to another port that also supports the frame's MTU size are forwarded in hardware. Previous releases support hardware forwarding of Layer 2 jumbo frames only.
- ICMP unreachable message if a frame is too large to be forwarded – If a jumbo packet has the Don't Fragment (DF) bit set, and the outbound interface does not support the packet's MTU size, the Foundry device sends an ICMP unreachable message to the device that sent the packet.

NOTE: These enhancements apply only to transit traffic forwarded through the Foundry device.

Configuration Considerations for Increasing the MTU

- When you increase the MTU size of a port, the increase uses system resources. Increase the MTU size only on the ports that need it. For example, if you have one port connected to a server that uses jumbo frames and two other ports connected to clients that can support the jumbo frames, increase the MTU only on those three ports. Leave the MTU size on the other ports at the default value (1500 bytes). Globally increase the MTU size only if needed.
- Use the same MTU size on all ports that will be supporting jumbo frames. If the device needs to fragment a

jumbo frame (and the frame does not have the DF bit set), the device fragments the frame into 1500-byte fragments, even if the outbound port has a larger MTU. For example, if a port has an MTU setting of 8000 and receives an 8000-byte frame, then must forward the frame onto a port with an MTU of 4000, the device does not fragment the 8000-byte frame into two 4000-byte frames. Instead, the device fragments the 8000-byte frame into six fragments (five 1500-byte fragments and a final, smaller fragment.)

Globally Changing the Maximum Transmission Unit (MTU)

The Maximum Transmission Unit (MTU) is the maximum size an IP packet can be when encapsulated in a Layer 2 packet. If an IP packet is larger than the MTU allowed by the Layer 2 packet, the Layer 3 Switch fragments the IP packet into multiple parts that will fit into the Layer 2 packets, and sends the parts of the fragmented IP packet separately, in different Layer 2 packets. The device that receives the multiple fragments of the IP packet reassembles the fragments into the original packet.

You can increase the MTU size to accommodate jumbo packet sizes up to 9216 bytes.

To globally enable jumbo support on all ports of an X-Series device, enter commands such as the following:

```
FESX424 Router(config)# jumbo
FESX424 Router(config)# write memory
FESX424 Router(config)# end
FESX424 Router# reload
```

Syntax: [no] jumbo

The above commands configure the Foundry device to forward Ethernet frames that are up to 9216 bytes long.

NOTE: You must save the configuration change and then reload the software to place the jumbo support into effect.

Changing the Maximum Transmission Unit on an Individual Port

By default, the maximum Ethernet MTU sizes are as follows:

- 1500 bytes – The maximum for Ethernet II encapsulation
- 1492 bytes – The maximum for SNAP encapsulation

When jumbo mode is enabled, the maximum Ethernet MTU sizes are as follows:

- 9216 bytes – The maximum for Ethernet II encapsulation
- 9216 bytes – The maximum for SNAP encapsulation

NOTE: If you set the MTU of a port to a value lower than the global MTU and from 576 – 1499, the port fragments the packets. However, if the port's MTU is exactly 1500 and this is larger than the global MTU, the port drops the packets.

NOTE: You must save the configuration change and then reload the software to place the jumbo support into effect.

To change the MTU for interface 1/5 to 1000, enter the following commands:

```
FastIron SuperX Router(config)# int e 1/5
FastIron SuperX Router(config-if-1/5)# ip mtu 1000
FastIron SuperX Router(config-if-1/5)# write memory
FastIron SuperX Router(config-if-1/5)# end
FastIron SuperX Router# reload
```

Syntax: [no] ip mtu <num>

The <num> parameter specifies the MTU. Ethernet II packets can hold IP packets from 576 – 1500 bytes long. If jumbo mode is enabled, Ethernet II packets can hold IP packets up to 9216 bytes long. Ethernet SNAP packets can hold IP packets from 576 – 1492 bytes long. If jumbo mode is enabled, SNAP packets can hold IP packets up to 9216 bytes long. The default MTU for Ethernet II packets is 1500. The default MTU for SNAP packets is 1492.

Path MTU Discovery (RFC 1191) Support

Foundry devices support the path MTU discovery method described in RFC 1191. When the Foundry device receives an IP packet that has its Don't Fragment (DF) bit set, and the packet size is greater than the MTU value of the outbound interface, then the Foundry device returns an ICMP Destination Unreachable message to the source of the packet, with the Code indicating "fragmentation needed and DF set". The ICMP Destination Unreachable message includes the MTU of the outbound interface. The source host can use this information to help determine the maximum MTU of a path to a destination.

RFC 1191 is supported on all interfaces.

Changing the Router ID

In most configurations, a Layer 3 Switch has multiple IP addresses, usually configured on different interfaces. As a result, a Layer 3 Switch's identity to other devices varies depending on the interface to which the other device is attached. Some routing protocols, including Open Shortest Path First (OSPF) and Border Gateway Protocol version 4 (BGP4), identify a Layer 3 Switch by just one of the IP addresses configured on the Layer 3 Switch, regardless of the interfaces that connect the Layer 3 Switches. This IP address is the router ID.

NOTE: Routing Information Protocol (RIP) does not use the router ID.

NOTE: If you change the router ID, all current BGP4 sessions are cleared.

By default, the router ID on a Foundry Layer 3 Switch is one of the following:

- If the router has loopback interfaces, the default router ID is the IP address configured on the lowest numbered loopback interface configured on the Layer 3 Switch. For example, if you configure loopback interfaces 1, 2, and 3 as follows, the default router ID is 9.9.9.9/24:
  - Loopback interface 1, 9.9.9.9/24
  - Loopback interface 2, 4.4.4.4/24
  - Loopback interface 3, 1.1.1.1/24
- If the device does not have any loopback interfaces, the default router ID is the lowest numbered IP interface configured on the device.

If you prefer, you can explicitly set the router ID to any valid IP address. The IP address cannot be in use on another device in the network.

NOTE: Foundry Layer 3 Switches use the same router ID for both OSPF and BGP4. If the router is already configured for OSPF, you may want to use the router ID that is already in use on the router rather than set a new one. To display the router ID, enter the **show ip** CLI command at any CLI level or select the IP->General links from the Configure tree in the Web management interface.

To change the router ID, enter a command such as the following:

```
FESX424 Router(config)# ip router-id 209.157.22.26
```

Syntax: ip router-id <ip-addr>

The <ip-addr> can be any valid, unique IP address.

NOTE: You can specify an IP address used for an interface on the Foundry Layer 3 Switch, but do not specify an IP address in use by another device.

Specifying a Single Source Interface for Telnet, TACACS/TACACS+, or RADIUS Packets

When the Layer 3 Switch originates a Telnet, TACACS/TACACS+, or RADIUS packet, the source address of the packet is the lowest-numbered IP address on the interface that sends the packet. You can configure the Layer 3 Switch to always the lowest-numbered IP address on a specific interface as the source addresses for these types of packets. When you configure the Layer 3 Switch to use a single source interface for all Telnet, TACACS/ TACACS+, or RADIUS packets, the Layer 3 Switch uses the same IP address as the source for all packets of the specified type, regardless of the port(s) that actually sends the packets.

Identifying a single source IP address for Telnet, TACACS/TACACS+, or RADIUS packets provides the following benefits:

- If your Telnet, TACACS/TACACS+, or RADIUS server is configured to accept packets only from specific IP addresses, you can use this feature to simplify configuration of the server by configuring the Foundry device to always send the packets from the same link or source address.
- If you specify a loopback interface as the single source for Telnet, TACACS/TACACS+, or RADIUS packets, servers can receive the packets regardless of the states of individual links. Thus, if a link to the server becomes unavailable but the client or server can be reached through another link, the client or server still receives the packets, and the packets still have the source IP address of the loopback interface.

The software contains separate CLI commands for specifying the source interface for Telnet, TACACS/TACACS+, or RADIUS packets. You can configure a source interface for one or more of these types of packets separately.

To specify an Ethernet or a loopback or virtual interface as the source for all TACACS/TACACS+ packets from the device, use the following CLI method. The software uses the lowest-numbered IP address configured on the port or interface as the source IP address for TACACS/TACACS+ packets originated by the device.

The following sections show the syntax for specifying a single source IP address for Telnet, TACACS/TACACS+, and RADIUS packets.

Telnet Packets

To specify the lowest-numbered IP address configured on a virtual interface as the device's source for all Telnet packets, enter commands such as the following:

```
FESX424 Router(config)# int loopback 2
FESX424 Router(config-lbif-2)# ip address 10.0.0.2/24
FESX424 Router(config-lbif-2)# exit
FESX424 Router(config)# ip telnet source-interface loopback 2
```

The commands in this example configure loopback interface 2, assign IP address 10.0.0.2/24 to the interface, then designate the interface as the source for all Telnet packets from the Layer 3 Switch.

Syntax: ip telnet source-interface ethernet [<slotnum>/]<portnum> | loopback <num> | ve <num>

The <num> parameter is a loopback interface or virtual interface number. If you specify an Ethernet port, the <portnum> is the port's number (including the slot number, if you are configuring a Chassis device).

The following commands configure an IP interface on an Ethernet port and designate the address port as the source for all Telnet packets from the Layer 3 Switch.

```
FastIron SuperX Router(config)# interface ethernet 1/4
FastIron SuperX Router(config-if-1/4)# ip address 209.157.22.110/24
FastIron SuperX Router(config-if-1/4)# exit
FastIron SuperX Router(config)# ip telnet source-interface ethernet 1/4
```


TACACS/TACACS+ Packets

To specify the lowest-numbered IP address configured on a virtual interface as the device's source for all TACACS/TACACS+ packets, enter commands such as the following:

```
FESX424 Router(config)# int ve 1
FESX424 Router(config-vif-1)# ip address 10.0.0.3/24
FESX424 Router(config-vif-1)# exit
FESX424 Router(config)# ip tacacs source-interface ve 1
```

The commands in this example configure virtual interface 1, assign IP address 10.0.0.3/24 to the interface, then designate the interface as the source for all TACACS/TACACS+ packets from the Layer 3 Switch.

Syntax: ip tacacs source-interface ethernet [<slotnum>/]<portnum> | loopback <num> | ve <num>

The <num> parameter is a loopback interface or virtual interface number. If you specify an Ethernet port, the <portnum> is the port's number (including the slot number, if you are configuring a Chassis device).

RADIUS Packets

To specify the lowest-numbered IP address configured on a virtual interface as the device's source for all RADIUS packets, enter commands such as the following:

```
FESX424 Router(config)# int ve 1
FESX424 Router(config-vif-1)# ip address 10.0.0.3/24
FESX424 Router(config-vif-1)# exit
FESX424 Router(config)# ip radius source-interface ve 1
```

The commands in this example configure virtual interface 1, assign IP address 10.0.0.3/24 to the interface, then designate the interface as the source for all RADIUS packets from the Layer 3 Switch.

Syntax: ip radius source-interface ethernet [<slotnum>/]<portnum> | loopback <num> | ve <num>

The <num> parameter is a loopback interface or virtual interface number. If you specify an Ethernet port, the <portnum> is the port's number (including the slot number, if you are configuring a Chassis device).

Configuring ARP Parameters

Address Resolution Protocol (ARP) is a standard IP protocol that enables an IP Layer 3 Switch to obtain the MAC address of another device's interface when the Layer 3 Switch knows the IP address of the interface. ARP is enabled by default and cannot be disabled.

NOTE: Foundry Layer 2 Switches also support ARP. The description in "How ARP Works" also applies to ARP on Foundry Layer 2 Switches. However, the configuration options described later in this section apply only to Layer 3 Switches, not to Layer 2 Switches.

How ARP Works

A Layer 3 Switch needs to know a destination's MAC address when forwarding traffic, because the Layer 3 Switch encapsulates the IP packet in a Layer 2 packet (MAC layer packet) and sends the Layer 2 packet to a MAC interface on a device directly attached to the Layer 3 Switch. The device can be the packet's final destination or the next-hop router toward the destination.

The Layer 3 Switch encapsulates IP packets in Layer 2 packets regardless of whether the ultimate destination is locally attached or is multiple router hops away. Since the Layer 3 Switch's IP route table and IP forwarding cache contain IP address information but not MAC address information, the Layer 3 Switch cannot forward IP packets based solely on the information in the route table or forwarding cache. The Layer 3 Switch needs to know the MAC address that corresponds with the IP address of either the packet's locally attached destination or the next-hop router that leads to the destination.

For example, to forward a packet whose destination is multiple router hops away, the Layer 3 Switch must send the packet to the next-hop router toward its destination, or to a default route or default network route if the IP route table does not contain a route to the packet's destination. In each case, the Layer 3 Switch must encapsulate the packet and address it to the MAC address of a locally attached device, the next-hop router toward the IP packet's destination.

To obtain the MAC address required for forwarding a datagram, the Layer 3 Switch does the following:

- First, the Layer 3 Switch looks in the ARP cache (not the static ARP table) for an entry that lists the MAC address for the IP address. The ARP cache maps IP addresses to MAC addresses. The cache also lists the port attached to the device and, if the entry is dynamic, the age of the entry. A dynamic ARP entry enters the cache when the Layer 3 Switch receives an ARP reply or receives an ARP request (which contains the sender's IP address and MAC address). A static entry enters the ARP cache from the static ARP table (which is a separate table) when the interface for the entry comes up.

  To ensure the accuracy of the ARP cache, each dynamic entry has its own age timer. The timer is reset to zero each time the Layer 3 Switch receives an ARP reply or ARP request containing the IP address and MAC address of the entry. If a dynamic entry reaches its maximum allowable age, the entry times out and the software removes the entry from the table. Static entries do not age out and can be removed only by you.

- If the ARP cache does not contain an entry for the destination IP address, the Layer 3 Switch broadcasts an ARP request out all its IP interfaces. The ARP request contains the IP address of the destination. If the device with the IP address is directly attached to the Layer 3 Switch, the device sends an ARP response containing its MAC address. The response is a unicast packet addressed directly to the Layer 3 Switch. The Layer 3 Switch places the information from the ARP response into the ARP cache.

  ARP requests contain the IP address and MAC address of the sender, so all devices that receive the request learn the MAC address and IP address of the sender and can update their own ARP caches accordingly.

  NOTE: The ARP request broadcast is a MAC broadcast, which means the broadcast goes only to devices that are directly attached to the Layer 3 Switch. A MAC broadcast is not routed to other networks. However, some routers, including Foundry Layer 3 Switches, can be configured to reply to ARP requests from one network on behalf of devices on another network. See "Enabling Proxy ARP" on page 16-27.

NOTE: If the router receives an ARP request packet that it is unable to deliver to the final destination because of the ARP timeout and no ARP response is received (the Layer 3 Switch knows of no route to the destination address), the router sends an ICMP Host Unreachable message to the source.

Rate Limiting ARP Packets

You can limit the number of ARP packets the Foundry device accepts during each second. By default, the software does not limit the number of ARP packets the device can receive. Since the device sends ARP packets to the CPU for processing, if a device in a busy network receives a high number of ARP packets in a short period of time, some CPU processing might be deferred while the CPU processes the ARP packets.

To prevent the CPU from becoming flooded by ARP packets in a busy network, you can restrict the number of ARP packets the device will accept each second. When you configure an ARP rate limit, the device accepts up to the maximum number of packets you specify, but drops additional ARP packets received during the one-second interval. When a new one-second interval starts, the counter restarts at zero, so the device again accepts up to the maximum number of ARP packets you specified, but drops additional packets received within the interval.

To limit the number of ARP packets the device will accept each second, enter a command such as the following at the global CONFIG level of the CLI:

```
FESX424 Router(config)# rate-limit-arp 100
```

This command configures the device to accept up to 100 ARP packets each second. If the device receives more than 100 ARP packets during a one-second interval, the device drops the additional ARP packets during the remainder of that one-second interval.

Syntax: [no] rate-limit-arp <num>

The <num> parameter specifies the number of ARP packets and can be from 0 – 100. If you specify 0, the device will not accept any ARP packets.

NOTE: If you want to change a previously configured the ARP rate limiting policy, you must remove the previously configured policy using the **no rate-limit-arp** <num> command before entering the new policy.

Changing the ARP Aging Period

When the Layer 3 Switch places an entry in the ARP cache, the Layer 3 Switch also starts an aging timer for the entry. The aging timer ensures that the ARP cache does not retain learned entries that are no longer valid. An entry can become invalid when the device with the MAC address of the entry is no longer on the network.

The ARP age affects dynamic (learned) entries only, not static entries. The default ARP age is ten minutes. On Layer 3 Switches, you can change the ARP age to a value from 0 – 240 minutes. You cannot change the ARP age on Layer 2 Switches. If you set the ARP age to zero, aging is disabled and entries do not age out.

To globally change the ARP aging parameter to 20 minutes, enter the following command:

```
FESX424 Router(config)# ip arp-age 20
```

Syntax: ip arp-age <num>

The <num> parameter specifies the number of minutes and can be from 0 – 240. The default is 10. If you specify 0, aging is disabled.

To override the globally configured IP ARP age on an individual interface, enter a command such as the following at the interface configuration level:

```
FastIron SuperX Router(config-if-e1000-1/1)# ip arp-age 30
```

Syntax: [no] ip arp-age <num>

The <num> parameter specifies the number of minutes and can be from 0 – 240. The default is the globally configured value, which is 10 minutes by default. If you specify 0, aging is disabled.

Enabling Proxy ARP

Proxy ARP allows a Layer 3 Switch to answer ARP requests from devices on one network on behalf of devices in another network. Since ARP requests are MAC-layer broadcasts, they reach only the devices that are directly connected to the sender of the ARP request. Thus, ARP requests do not cross routers.

For example, if Proxy ARP is enabled on a Layer 3 Switch connected to two sub-nets, 10.10.10.0/24 and 20.20.20.0/24, the Layer 3 Switch can respond to an ARP request from 10.10.10.69 for the MAC address of the device with IP address 20.20.20.69. In standard ARP, a request from a device in the 10.10.10.0/24 sub-net cannot reach a device in the 20.20.20.0 sub-net if the sub-nets are on different network cables, and thus is not answered.

NOTE: An ARP request from one sub-net can reach another sub-net when both sub-nets are on the same physical segment (Ethernet cable), since MAC-layer broadcasts reach all the devices on the segment.

Proxy ARP is disabled by default on Foundry Layer 3 Switches. The feature is not supported on Foundry Layer 2 Switches.

To enable IP proxy ARP, enter the following command:

```
FESX424 Router(config)# ip proxy-arp
```

To again disable IP proxy ARP, enter the following command:

```
FESX424 Router(config)# no ip proxy-arp
```

Syntax: [no] ip proxy-arp

Enabling Local Proxy ARP

Foundry devices support Proxy Address Resolution Protocol (***Proxy ARP***), a feature that enables router ports to respond to ARP requests for subnets it can reach. However, router ports will not respond to ARP requests for IP addresses in the same subnet as the incoming ports. Software release 02.3.03 resolves this issue with the

introduction of Local Proxy ARP per IP interface. ***Local Proxy ARP*** enables router ports to reply to ARP requests for IP addresses within the same subnet and to forward all traffic between hosts in the subnet.

When Local Proxy ARP is enabled on a router port, the port will respond to ARP requests for IP addresses within the same subnet, if it has ARP entries for the destination IP addresses in the ARP cache. If it does not have ARP entries for the IP addresses, the port will attempt to resolve them by broadcasting its own ARP requests.

Local Proxy ARP is disabled by default. To use Local Proxy ARP, Proxy ARP (CLI command **ip proxy-arp**) must be enabled globally on the Foundry device. You can enter the CLI command to enable Local Proxy ARP even though Proxy ARP is not enabled, however, the configuration will not take effect until you enable Proxy ARP.

Use the **show run** command to view the ports on which Local Proxy ARP is enabled.

To enable Local Proxy ARP, enter commands such as the following:

```
FESX424 Switch(config)# int e 4
FESX424 Switch(config-if-e1000-4)# ip local-proxy-arp
```

Syntax: [no] ip local-proxy-arp

Use the **no** form of the command to disable Local Proxy ARP.

Creating Static ARP Entries

Foundry Layer 3 Switches have a static ARP table, in addition to the regular ARP cache. The static ARP table contains entries that you configure.

Static entries are useful in cases where you want to pre-configure an entry for a device that is not connected to the Layer 3 Switch, or you want to prevent a particular entry from aging out. The software removes a dynamic entry from the ARP cache if the ARP aging interval expires before the entry is refreshed. Static entries do not age out, regardless of whether the Foundry device receives an ARP request from the device that has the entry's address.

NOTE: You cannot create static ARP entries on a Layer 2 Switch.

The maximum number of static ARP entries you can configure depends on the product. See "Changing the Maximum Number of Entries the Static ARP Table Can Hold" on page 16-28.

To display the ARP cache and static ARP table, see the following:

- To display the ARP table, see "Displaying the ARP Cache" on page 16-63.
- To display the static ARP table, see "Displaying the Static ARP Table" on page 16-65.

To create a static ARP entry, enter a command such as the following:

```
FESX424 Router(config)# arp 1 192.53.4.2 1245.7654.2348 e 1/2
```

Syntax: arp <num> <ip-addr> <mac-addr> ethernet [<slotnum>/]<portnum>

The <num> parameter specifies the entry number. You can specify a number from 1 up to the maximum number of static entries allowed on the device.

The <ip-addr> command specifies the IP address of the device that has the MAC address of the entry.

The <mac-addr> parameter specifies the MAC address of the entry.

The <slotnum> parameter is required on chassis devices.

The <portnum> command specifies the port number attached to the device that has the MAC address of the entry.

Changing the Maximum Number of Entries the Static ARP Table Can Hold

Table 16.6 on page 16-29 lists the default maximum and configurable maximum number of entries in the static ARP table that are supported on each type of Foundry Layer 3 Switch. If you need to change the maximum number of entries supported on a Layer 3 Switch, use either of the following methods.

NOTE: You must save the configuration to the startup-config file and reload the software after changing the static ARP table size to place the change into effect.

NOTE: The basic procedure for changing the static ARP table size is the same as the procedure for changing other configurable cache or table sizes. See the section "Displaying and Modifying System Parameter Default Settings" on page 4-8.

To increase the maximum number of entries in the static ARP table you can configure on a Foundry Layer 3 Switch, enter commands such as the following at the global CONFIG level of the CLI:

```
FESX424 Router(config)# system-max ip-static-arp 1000
FESX424 Router(config)# write memory
FESX424 Router(config)# end
FESX424 Router# reload
```

Syntax: system-max ip-static-arp <num>

The <num> parameter indicates the maximum number of static ARP entries and can be a number in one of the following ranges, depending on the device you are configuring. The table below lists the default maximum and range of configurable maximums for static ARP table entries supported on a Foundry Layer 3 Switch.

Table 16.6: Static ARP Entry Support

Default Maximum	Configurable Minimum	Configurable Maximum
512	512	1024

Configuring Forwarding Parameters

The following configurable parameters control the forwarding behavior of Foundry Layer 3 Switches:

- Time-To-Live (TTL) threshold
- Forwarding of directed broadcasts
- Forwarding of source-routed packets
- Ones-based and zero-based broadcasts

All these parameters are global and thus affect all IP interfaces configured on the Layer 3 Switch.

To configure these parameters, use the procedures in the following sections.

Changing the TTL Threshold

The TTL threshold prevents routing loops by specifying the maximum number of router hops an IP packet originated by the Layer 3 Switch can travel through. Each device capable of forwarding IP that receives the packet decrements (decreases) the packet's TTL by one. If a device receives a packet with a TTL of 1 and reduces the TTL to zero, the device drops the packet.

The default TTL is 64. You can change the TTL to a value from 1– 255.

To modify the TTL threshold to 25, enter the following commands:

```
FESX424 Router(config)# ip ttl 25
```

Syntax: ip ttl <1-255>

Enabling Forwarding of Directed Broadcasts

A directed broadcast is an IP broadcast to all devices within a single directly-attached network or sub-net. A net-directed broadcast goes to all devices on a given network. A sub-net-directed broadcast goes to all devices within a given sub-net.

NOTE: A less common type, the all-sub-nets broadcast, goes to all directly-attached sub-nets. Forwarding for this broadcast type also is supported, but most networks use IP multicasting instead of all-sub-net broadcasting.

Forwarding for all types of IP directed broadcasts is disabled by default. You can enable forwarding for all types if needed. You cannot enable forwarding for specific broadcast types.

To enable forwarding of IP directed broadcasts, enter the following command:

```
FESX424 Router(config)# ip directed-broadcast
```

Syntax: [no] ip directed-broadcast

Foundry software makes the forwarding decision based on the router's knowledge of the destination network prefix. Routers cannot determine that a message is unicast or directed broadcast apart from the destination network prefix. The decision to forward or not forward the message is by definition only possible in the last hop router.

To disable the directed broadcasts, enter the following command in the CONFIG mode:

```
FESX424 Router(config)# no ip directed-broadcast
```

To enable directed broadcasts on an individual interface instead of globally for all interfaces, enter commands such as the following:

```
FastIron SuperX Router(config)# interface ethernet 1/1
FastIron SuperX Router(config-if-1/1)# ip directed-broadcast
```

Syntax: [no] ip directed-broadcast

Disabling Forwarding of IP Source-Routed Packets

A source-routed packet specifies the exact router path for the packet. The packet specifies the path by listing the IP addresses of the router interfaces through which the packet must pass on its way to the destination. The Layer 3 Switch supports both types of IP source routing:

- Strict source routing – requires the packet to pass through only the listed routers. If the Layer 3 Switch receives a strict source-routed packet but cannot reach the next hop interface specified by the packet, the Layer 3 Switch discards the packet and sends an ICMP Source-Route-Failure message to the sender.

  NOTE: The Layer 3 Switch allows you to disable sending of the Source-Route-Failure messages. See “Disabling ICMP Messages” on page 16-31.

- Loose source routing – requires that the packet pass through all of the listed routers but also allows the packet to travel through other routers, which are not listed in the packet.

The Layer 3 Switch forwards both types of source-routed packets by default. To disable the feature, use either of the following methods. You cannot enable or disable strict or loose source routing separately.

To disable forwarding of IP source-routed packets, enter the following command:

```
FESX424 Router(config)# no ip source-route
```

Syntax: [no] ip source-route

To re-enable forwarding of source-routed packets, enter the following command:

```
FESX424 Router(config)# ip source-route
```

Enabling Support for Zero-Based IP Sub-Net Broadcasts

By default, the Layer 3 Switch treats IP packets with all ones in the host portion of the address as IP broadcast packets. For example, the Layer 3 Switch treats IP packets with 209.157.22.255/24 as the destination IP address as IP broadcast packets and forwards the packets to all IP hosts within the 209.157.22.x sub-net (except the host that sent the broadcast packet to the Layer 3 Switch).

Most IP hosts are configured to receive IP sub-net broadcast packets with all ones in the host portion of the address. However, some older IP hosts instead expect IP sub-net broadcast packets that have all zeros instead of

all ones in the host portion of the address. To accommodate this type of host, you can enable the Layer 3 Switch to treat IP packets with all zeros in the host portion of the destination IP address as broadcast packets.

NOTE: When you enable the Layer 3 Switch for zero-based sub-net broadcasts, the Layer 3 Switch still treats IP packets with all ones the host portion as IP sub-net broadcasts too. Thus, the Layer 3 Switch can be configured to support all ones only (the default) or all ones ***and*** all zeroes.

NOTE: This feature applies only to IP sub-net broadcasts, not to local network broadcasts. The local network broadcast address is still expected to be all ones.

To enable the Layer 3 Switch for zero-based IP sub-net broadcasts in addition to ones-based IP sub-net broadcasts, enter the following command.

```
FESX424 Router(config)# ip broadcast-zero
```

Syntax: [no] ip broadcast-zero

Disabling ICMP Messages

Foundry devices are enabled to reply to ICMP echo messages and send ICMP Destination Unreachable messages by default.

You can selectively disable the following types of Internet Control Message Protocol (ICMP) messages:

- Echo messages (ping messages) – The Layer 3 Switch replies to IP pings from other IP devices.
- Destination Unreachable messages – If the Layer 3 Switch receives an IP packet that it cannot deliver to its destination, the Layer 3 Switch discards the packet and sends a message back to the device that sent the packet to the Layer 3 Switch. The message informs the device that the destination cannot be reached by the Layer 3 Switch.

Disabling Replies to Broadcast Ping Requests

By default, Foundry devices are enabled to respond to broadcast ICMP echo packets, which are ping requests.

To disable response to broadcast ICMP echo packets (ping requests), enter the following command:

```
FESX424 Router(config)# no ip icmp echo broadcast-request
```

Syntax: [no] ip icmp echo broadcast-request

If you need to re-enable response to ping requests, enter the following command:

```
FESX424 Router(config)# ip icmp echo broadcast-request
```

Disabling ICMP Destination Unreachable Messages

By default, when a Foundry device receives an IP packet that the device cannot deliver, the device sends an ICMP Unreachable message back to the host that sent the packet. You can selectively disable a Foundry device's response to the following types of ICMP Unreachable messages:

- Administration – The packet was dropped by the Foundry device due to a filter or ACL configured on the device.
- Fragmentation-needed – The packet has the Don't Fragment bit set in the IP Flag field, but the Foundry device cannot forward the packet without fragmenting it.
- Host – The destination network or sub-net of the packet is directly connected to the Foundry device, but the host specified in the destination IP address of the packet is not on the network.
- Port – The destination host does not have the destination TCP or UDP port specified in the packet. In this case, the host sends the ICMP Port Unreachable message to the Foundry device, which in turn sends the message to the host that sent the packet.
- Protocol – The TCP or UDP protocol on the destination host is not running. This message is different from the Port Unreachable message, which indicates that the protocol is running on the host but the requested protocol port is unavailable.

- Source-route-failure – The device received a source-routed packet but cannot locate the next-hop IP address indicated in the packet's Source-Route option.

You can disable the Foundry device from sending these types of ICMP messages on an individual basis. To do so, use the following CLI method.

NOTE: Disabling an ICMP Unreachable message type does not change the Foundry device's ability to forward packets. Disabling ICMP Unreachable messages prevents the device from generating or forwarding the Unreachable messages.

To disable all ICMP Unreachable messages, enter the following command:

```
FESX424 Router(config)# no ip icmp unreachable
```

Syntax: [no] ip icmp unreachable [host | protocol | administration | fragmentation-needed | port | source-route-fail]

- If you enter the command without specifying a message type (as in the example above), all types of ICMP Unreachable messages listed above are disabled. If you want to disable only specific types of ICMP Unreachable messages, you can specify the message type. To disable more than one type of ICMP message, enter the **no ip icmp unreachable** command for each messages type.
- The **administration** parameter disables ICMP Unreachable (caused by Administration action) messages.
- The **fragmentation-needed** parameter disables ICMP Fragmentation-Needed But Don't-Fragment Bit Set messages.
- The **host** parameter disables ICMP Host Unreachable messages.
- The **port** parameter disables ICMP Port Unreachable messages.
- The **protocol** parameter disables ICMP Protocol Unreachable messages.
- The **source-route-fail** parameter disables ICMP Unreachable (caused by Source-Route-Failure) messages.

To disable ICMP Host Unreachable messages but leave the other types of ICMP Unreachable messages enabled, enter the following commands instead of the command shown above:

```
FESX424 Router(config)# no ip icmp unreachable host
```

If you have disabled all ICMP Unreachable message types but you want to re-enable certain types, for example ICMP Host Unreachable messages, you can do so by entering the following command:

```
FESX424 Router(config)# ip icmp unreachable host
```

Configuring Static Routes

The IP route table can receive routes from the following sources:

- Directly-connected networks – When you add an IP interface, the Layer 3 Switch automatically creates a route for the network the interface is in.
- RIP – If RIP is enabled, the Layer 3 Switch can learn about routes from the advertisements other RIP routers send to the Layer 3 Switch. If the route has a lower administrative distance than any other routes from different sources to the same destination, the Layer 3 Switch places the route in the IP route table.
- OSPF – See RIP, but substitute "OSPF" for "RIP".
- BGP4 – See RIP, but substitute "BGP4" for "RIP".
- Default network route – A statically configured default route that the Layer 3 Switch uses if other default routes to the destination are not available. See "Configuring a Default Network Route" on page 16-39.
- Statically configured route – You can add routes directly to the route table. When you add a route to the IP route table, you are creating a static IP route. This section describes how to add static routes to the IP route table.

Static Route Types

You can configure the following types of static IP routes:

- Standard – the static route consists of the destination network address and network mask, and the IP address of the next-hop gateway. You can configure multiple standard static routes with the same metric for load sharing or with different metrics to provide a primary route and backup routes.
- Interface-based – the static route consists of the destination network address and network mask, and the Layer 3 Switch interface through which you want the Layer 3 Switch to send traffic for the route. Typically, this type of static route is for directly attached destination networks.
- Null – the static route consists of the destination network address and network mask, and the "null0" parameter. Typically, the null route is configured as a backup route for discarding traffic if the primary route is unavailable.

Static IP Route Parameters

When you configure a static IP route, you must specify the following parameters:

- The IP address and network mask for the route's destination network.
- The route's path, which can be one of the following:
  - The IP address of a next-hop gateway
  - An Ethernet port
  - A virtual interface (a routing interface used by VLANs for routing Layer 3 protocol traffic among one another)
  - A "null" interface. The Layer 3 Switch drops traffic forwarded to the null interface.

You also can specify the following optional parameters:

- The route's metric – The value the Layer 3 Switch uses when comparing this route to other routes in the IP route table to the same destination. The metric applies only to routes that the Layer 3 Switch has already placed in the IP route table. The default metric for static IP routes is 1.
- The route's administrative distance – The value that the Layer 3 Switch uses to compare this route with routes from other route sources to the same destination before placing a route in the IP route table. This parameter does not apply to routes that are already in the IP route table. The default administrative distance for static IP routes is 1.

The default metric and administrative distance values ensure that the Layer 3 Switch always prefers static IP routes over routes from other sources to the same destination.

Multiple Static Routes to the Same Destination Provide Load Sharing and Redundancy

You can add multiple static routes for the same destination network to provide one or more of the following benefits:

- IP load balancing – When you add multiple IP static routes for the same destination to different next-hop gateways, and the routes each have the same metric and administrative distance, the Layer 3 Switch can load balance traffic to the routes' destination. For information about IP load balancing, see "Configuring IP Load Sharing" on page 16-41.
- Path redundancy – When you add multiple static IP routes for the same destination, but give the routes different metrics or administrative distances, the Layer 3 Switch uses the route with the lowest administrative distance by default, but uses another route to the same destination of the first route becomes unavailable.

See the following sections for examples and configuration information:

- "Configuring Load Balancing and Redundancy Using Multiple Static Routes to the Same Destination" on page 16-36
- "Configuring Standard Static IP Routes and Interface or Null Static Routes to the Same Destination" on page 16-37

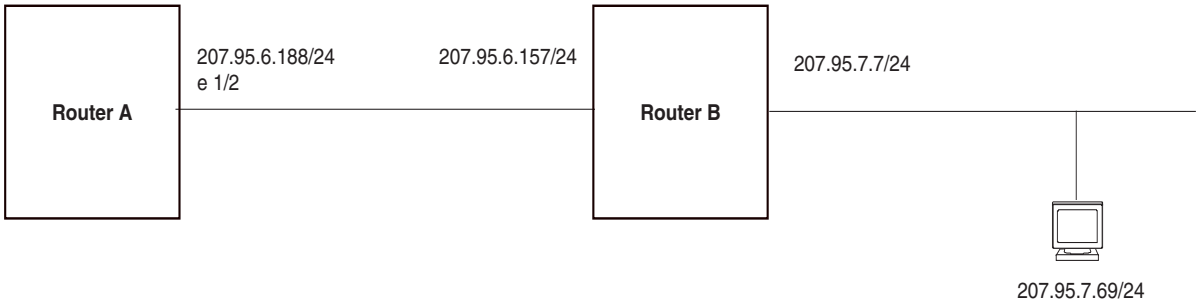
Static Route States Follow Port States

IP static routes remain in the IP route table only so long as the port or virtual interface used by the route is available. If the port or virtual routing interface becomes unavailable, the software removes the static route from the IP route table. If the port or virtual routing interface becomes available again later, the software adds the route back to the route table.

This feature allows the Layer 3 Switch to adjust to changes in network topology. The Layer 3 Switch does not continue trying to use routes on unavailable paths but instead uses routes only when their paths are available.

Figure 16.2 shows an example of a network containing a static route. The static route is configured on Router A, as shown in the CLI example following the figure.

Figure 16.2 Example of a static route



The following command configures a static route to 207.95.7.0, using 207.95.6.157 as the next-hop gateway.

```
FastIron SuperX Router(config)# ip route 207.95.7.0/24 207.95.6.157
```

When you configure a static IP route, you specify the destination address for the route and the next-hop gateway or Layer 3 Switch interface through which the Layer 3 Switch can reach the route. The Layer 3 Switch adds the route to the IP route table. In this case, Router A knows that 207.95.6.157 is reachable through port 1/2, and also assumes that local interfaces within that sub-net are on the same port. Router A deduces that IP interface 207.95.7.188 is also on port 1/2.

The software automatically removes a static IP route from the IP route table if the port used by that route becomes unavailable. When the port becomes available again, the software automatically re-adds the route to the IP route table.

Configuring a Static IP Route

To configure an IP static route with a destination address of 192.0.0.0 255.0.0.0 and a next-hop router IP address of 195.1.1.1, enter the following commands:

```
FastIron SuperX Router(config)# ip route 192.0.0.0 255.0.0.0 195.1.1.1
```

To configure a static IP route with an Ethernet port instead of a next-hop address, enter a command such as the following.

```
FastIron SuperX Router(config)# ip route 192.128.2.69 255.255.255.0 ethernet 4/1
```

The command in the example above configures a static IP route for destination network 192.128.2.69/24. Since an Ethernet port is specified instead of a gateway IP address as the next hop, the Layer 3 Switch always forwards traffic for the 192.128.2.69/24 network to port 4/1. The command in the following example configures an IP static route that uses virtual interface 3 as its next hop.

```
FastIron SuperX Router(config)# ip route 192.128.2.71 255.255.255.0 ve 3
```

The command in the following example configures an IP static route that uses port 2/2 as its next hop.

```
FastIron SuperX Router(config)# ip route 192.128.2.73 255.255.255.0 ethernet 2/2
```

Syntax: ip route <dest-ip-addr> <dest-mask>
<next-hop-ip-addr> |

ethernet [<slotnum>/]<portnum> | ve <num>
[<metric>] [distance <num>]

or

Syntax: ip route <dest-ip-addr>/<mask-bits>
<next-hop-ip-addr> |
ethernet [<slotnum>/]<portnum> | ve <num>
[<metric>] [distance <num>]

The <dest-ip-addr> is the route's destination. The <dest-mask> is the network mask for the route's destination IP address. Alternatively, you can specify the network mask information by entering a forward slash followed by the number of bits in the network mask. For example, you can enter 192.0.0.0 255.255.255.0 as 192.0.0.0/.24.

The <next-hop-ip-addr> is the IP address of the next-hop router (gateway) for the route.

If you do not want to specify a next-hop IP address, you can instead specify a port or interface number on the Layer 3 Switch. The <num> parameter is a virtual interface number. If you instead specify an Ethernet port, the <portnum> is the port's number (including the slot number, if you are configuring a Chassis device). In this case, the Layer 3 Switch forwards packets destined for the static route's destination network to the specified interface. Conceptually, this feature makes the destination network like a directly connected network, associated with a specific Layer 3 Switch interface.

NOTE: The port or virtual interface you use for the static route's next hop must have at least one IP address configured on it. The address does not need to be in the same sub-net as the destination network.

The <metric> parameter can be a number from 1 – 16. The default is 1.

NOTE: If you specify 16, RIP considers the metric to be infinite and thus also considers the route to be unreachable.

The **distance** <num> parameter specifies the administrative distance of the route. When comparing otherwise equal routes to a destination, the Layer 3 Switch prefers lower administrative distances over higher ones, so make sure you use a low value for your default route. The default is 1.

NOTE: The Layer 3 Switch will replace the static route if the it receives a route with a lower administrative distance. See "Changing Administrative Distances" on page 21-29 for a list of the default administrative distances for all types of routes.

NOTE: You can also assign the default router as the destination by entering 0.0.0.0 0.0.0.0 xxx.xxx.xxx.xxx.

Configuring a "Null" Route

You can configure the Layer 3 Switch to drop IP packets to a specific network or host address by configuring a "null" (sometimes called "null0") static route for the address. When the Layer 3 Switch receives a packet destined for the address, the Layer 3 Switch drops the packet instead of forwarding it.

To configure a null static route, use the following CLI method.

To configure a null static route to drop packets destined for network 209.157.22.x, enter the following commands.

```
FESX424 Router(config)# ip route 209.157.22.0 255.255.255.0 null0
FESX424 Router(config)# write memory
```

Syntax: ip route <ip-addr> <ip-mask> null0 [<metric>] [distance <num>]

or

Syntax: ip route <ip-addr>/<mask-bits> null0 [<metric>] [distance <num>]

To display the maximum value for your device, enter the **show default values** command. The maximum number of static IP routes the system can hold is listed in the ip-static-route row in the System Parameters section of the

display. To change the maximum value, use the **system-max ip-static-route** <num> command at the global CONFIG level.

The <ip-addr> parameter specifies the network or host address. The Layer 3 Switch will drop packets that contain this address in the destination field instead of forwarding them.

The <ip-mask> parameter specifies the network mask. Ones are significant bits and zeros allow any value. For example, the mask 255.255.255.0 matches on all hosts within the Class C sub-net address specified by <ip-addr>. Alternatively, you can specify the number of bits in the network mask. For example, you can enter 209.157.22.0/24 instead of 209.157.22.0 255.255.255.0.

The **null0** parameter indicates that this is a null route. You must specify this parameter to make this a null route.

The <metric> parameter adds a cost to the route. You can specify from 1 – 16. The default is 1.

The distance <num> parameter configures the administrative distance for the route. You can specify a value from 1 – 255. The default is 1. The value 255 makes the route unusable.

NOTE: The last two parameters are optional and do not affect the null route, unless you configure the administrative distance to be 255. In this case, the route is not used and the traffic might be forwarded instead of dropped.

Configuring Load Balancing and Redundancy Using Multiple Static Routes to the Same Destination

You can configure multiple static IP routes to the same destination, for the following benefits:

- IP load sharing – If you configure more than one static route to the same destination, and the routes have different next-hop gateways but have the same metrics, the Layer 3 Switch load balances among the routes using basic round-robin. For example, if you configure two static routes with the same metrics but to different gateways, the Layer 3 Switch alternates between the two routes. For information about IP load balancing, see “Configuring IP Load Sharing” on page 16-41.
- Backup Routes – If you configure multiple static IP routes to the same destination, but give the routes different next-hop gateways and different metrics, the Layer 3 Switch will always use the route with the lowest metric. If this route becomes unavailable, the Layer 3 Switch will fail over to the static route with the next-lowest metric, and so on.

NOTE: You also can bias the Layer 3 Switch to select one of the routes by configuring them with different administrative distances. However, make sure you do not give a static route a higher administrative distance than other types of routes, unless you want those other types to be preferred over the static route. For a list of the default administrative distances, see “Changing Administrative Distances” on page 21-29.

The steps for configuring the static routes are the same as described in the previous section. The following sections provide examples.

To configure multiple static IP routes, enter commands such as the following.

```
FESX424 Router(config)# ip route 192.128.2.69 255.255.255.0 209.157.22.1
FESX424 Router(config)# ip route 192.128.2.69 255.255.255.0 192.111.10.1
```

The commands in the example above configure two static IP routes. The routes go to different next-hop gateways but have the same metrics. These commands use the default metric value (1), so the metric is not specified. These static routes are used for load sharing among the next-hop gateways.

The following commands configure static IP routes to the same destination, but with different metrics. The route with the lowest metric is used by default. The other routes are backups in case the first route becomes unavailable. The Layer 3 Switch uses the route with the lowest metric if the route is available.

```
FESX424 Router(config)# ip route 192.128.2.69 255.255.255.0 209.157.22.1
FESX424 Router(config)# ip route 192.128.2.69 255.255.255.0 192.111.10.1 2
FESX424 Router(config)# ip route 192.128.2.69 255.255.255.0 201.1.1.1 3
```


In this example, each static route has a different metric. The metric is not specified for the first route, so the default (1) is used. A metric is specified for the second and third static IP routes. The second route has a metric of two and the third route has a metric of 3. Thus, the second route is used only of the first route (which has a metric of 1) becomes unavailable. Likewise, the third route is used only if the first and second routes (which have lower metrics) are both unavailable.

For complete syntax information, see “Configuring a Static IP Route” on page 16-34.

Configuring Standard Static IP Routes and Interface or Null Static Routes to the Same Destination

You can configure a null0 or interface-based static route to a destination and also configure a normal static route to the same destination, so long as the route metrics are different.

When the Layer 3 Switch has multiple routes to the same destination, the Layer 3 Switch always prefers the route with the lowest metric. Generally, when you configure a static route to a destination network, you assign the route a low metric so that the Layer 3 Switch prefers the static route over other routes to the destination.

This feature is especially useful for the following configurations. These are not the only allowed configurations but they are typical uses of this enhancement.

- When you want to ensure that if a given destination network is unavailable, the Layer 3 Switch drops (forwards to the null interface) traffic for that network instead of using alternate paths to route the traffic. In this case, assign the normal static route to the destination network a lower metric than the null route.
- When you want to use a specific interface by default to route traffic to a given destination network, but want to allow the Layer 3 Switch to use other interfaces to reach the destination network if the path that uses the default interface becomes unavailable. In this case, give the interface route a lower metric than the normal static route.

NOTE: You cannot add a null or interface-based static route to a network if there is already a static route of any type with the same metric you specify for the null or interface-based route.

Figure 16.3 shows an example of two static routes configured for the same destination network. In this example, one of the routes is a standard static route and has a metric of 1. The other static route is a null route and has a higher metric than the standard static route. The Layer 3 Switch always prefers the static route with the lower metric. In this example, the Layer 3 Switch always uses the standard static route for traffic to destination network 192.168.7.0/24, unless that route becomes unavailable, in which case the Layer 3 Switch sends traffic to the null route instead.

Figure 16.3 Standard and null static routes to the same destination network

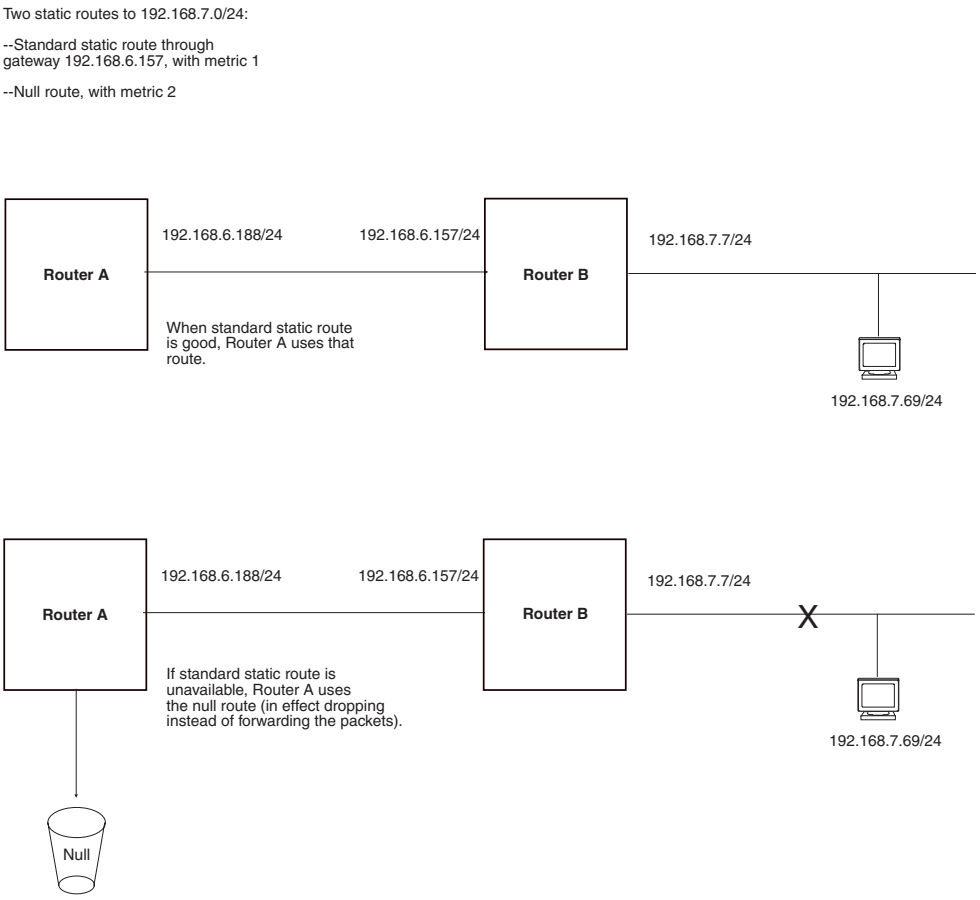
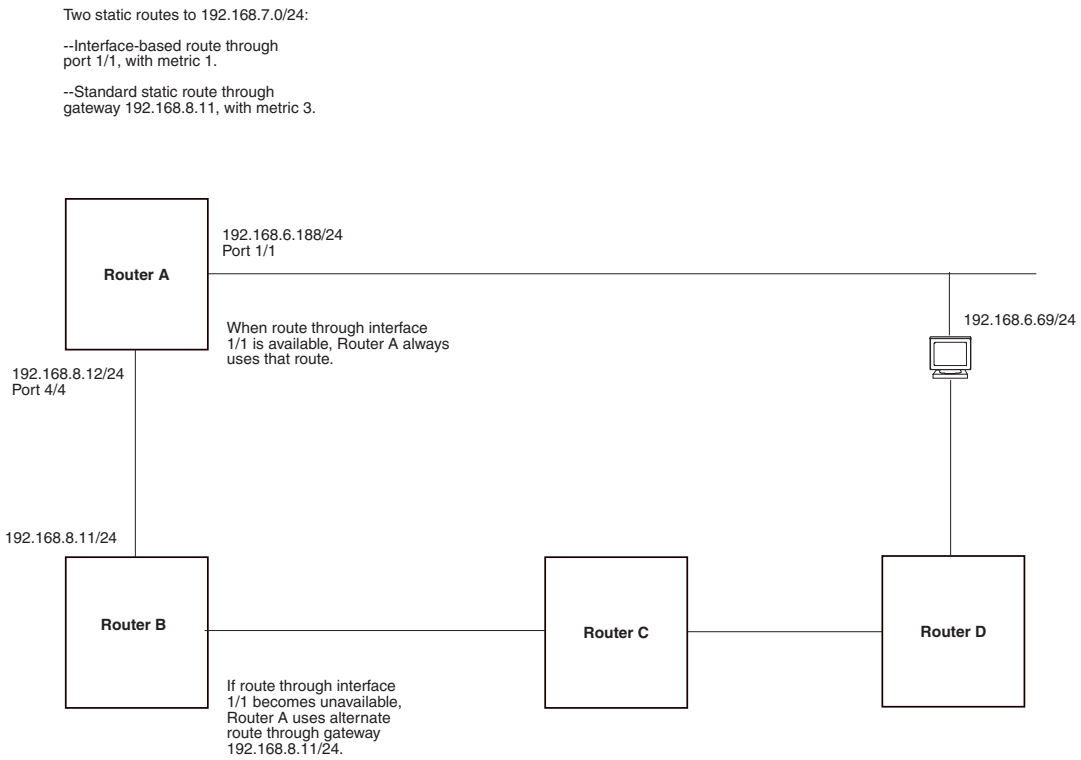


Figure 16.4 shows another example of two static routes. In this example, a standard static route and an interface-based static route are configured for destination network 192.168.6.0/24. The interface-based static route has a lower metric than the standard static route. As a result, the Layer 3 Switch always prefers the interface-based route when the route is available. However, if the interface-based route becomes unavailable, the Layer 3 Switch still forwards the traffic toward the destination using an alternate route through gateway 192.168.8.11/24.

Figure 16.4 Standard and interface routes to the same destination network



To configure a standard static IP route and a null route to the same network as shown in Figure 16.3 on page 16-38, enter commands such as the following:

```
FastIron SuperX Router(config)# ip route 192.168.7.0/24 192.168.6.157/24 1
FastIron SuperX Router(config)# ip route 192.168.7.0/24 null0 3
```

The first command configures a standard static route, which includes specification of the next-hop gateway. The command also gives the standard static route a metric of 1, which causes the Layer 3 Switch to always prefer this route when the route is available.

The second command configures another static route for the same destination network, but the second route is a null route. The metric for the null route is 3, which is higher than the metric for the standard static route. If the standard static route is unavailable, the software uses the null route.

For complete syntax information, see “Configuring a Static IP Route” on page 16-34.

To configure a standard static route and an interface-based route to the same destination, enter commands such as the following:

```
FastIron SuperX Router(config)# ip route 192.168.6.0/24 ethernet 1/1 1
FastIron SuperX Router(config)# ip route 192.168.6.0/24 192.168.8.11/24 3
```

The first command configured an interface-based static route through Ethernet port 1/1. The command assigns a metric of 1 to this route, causing the Layer 3 Switch to always prefer this route when it is available. If the route becomes unavailable, the Layer 3 Switch uses an alternate route through the next-hop gateway 192.168.8.11/24.

Configuring a Default Network Route

The Layer 3 Switch enables you to specify a candidate default route without the need to specify the next hop gateway. If the IP route table does not contain an explicit default route (for example, 0.0.0.0/0) or propagate an explicit default route through routing protocols, the software can use the default network route as a default route instead.

When the software uses the default network route, it also uses the default network route's next hop gateway as the gateway of last resort.

This feature is especially useful in environments where network topology changes can make the next hop gateway unreachable. This feature allows the Layer 3 Switch to perform default routing even if the default network route's default gateway changes.

The feature thus differs from standard default routes. When you configure a standard default route, you also specify the next hop gateway. If a topology change makes the gateway unreachable, the default route becomes unusable.

For example, if you configure 10.10.10.0/24 as a candidate default network route, if the IP route table does not contain an explicit default route (0.0.0.0/0), the software uses the default network route and automatically uses that route's next hop gateway as the default gateway. If a topology change occurs and as a result the default network route's next hop gateway changes, the software can still use the default network route. To configure a default network route, use the following CLI method.

If you configure more than one default network route, the Layer 3 Switch uses the following algorithm to select one of the routes:

1. Use the route with the lowest administrative distance.
2. If the administrative distances are equal:
    - Are the routes from different routing protocols (RIP, OSPF, or BGP4)? If so, use the route with the lowest IP address.
    - If the routes are from the same routing protocol, use the route with the best metric. The meaning of “best” metric depends on the routing protocol:
    - RIP – The metric is the number of hops (additional routers) to the destination. The best route is the route with the fewest hops.
    - OSPF – The metric is the path cost associated with the route. The path cost does not indicate the number of hops but is instead a numeric value associated with each route. The best route is the route with the lowest path cost.
    - BGP4 – The metric is the Multi-exit Discriminator (MED) associated with the route. The MED applies to routes that have multiple paths through the same AS. The best route is the route with the lowest MED.

Configuring a Default Network Route

You can configure up to four default network routes.

To configure a default network route, enter commands such as the following:

```
FESX424 Router(config)# ip default-network 209.157.22.0
FESX424 Router(config)# write memory
```

Syntax: ip default-network <ip-addr>

The <ip-addr> parameter specifies the network address.

To verify that the route is in the route table, enter the following command at any level of the CLI:

```
FESX424 Router(config)# show ip route

Total number of IP routes: 2
Start index: 1  B:BGP D:Connected  R:RIP  S:Static  O:OSPF *:Candidate default
      Destination       NetMask           Gateway           Port   Cost   Type
1     209.157.20.0      255.255.255.0     0.0.0.0           lb1    1      D
2     209.157.22.0      255.255.255.0     0.0.0.0           4/11   1      *D
```


This example shows two routes. Both of the routes are directly attached, as indicated in the Type column. However, one of the routes is shown as type "*D", with an asterisk (*). The asterisk indicates that this route is a candidate default network route.

Configuring IP Load Sharing

The IP route table can contain more than one path to a given destination. When this occurs, the Layer 3 Switch selects the path with the lowest cost as the path for forwarding traffic to the destination. If the IP route table contains more than one path to a destination and the paths each have the lowest cost, then the Layer 3 Switch uses ***IP load sharing*** to select a path to the destination.[1]

On X-Series devices, IP load sharing uses a hashing algorithm based on the source IP address, destination IP address, and protocol field in the IP header.

You can enable a Layer 3 Switch to load balance across up to six equal-cost paths. The default maximum number of equal-cost load sharing paths is four.

NOTE: IP load sharing is not based on source routing, only on next-hop routing.

NOTE: The term "path" refers to the next-hop router to a destination, not to the entire route to a destination. Thus, when the software compares multiple equal-cost paths, the software is comparing paths that use different next-hop routers, with equal costs, to the same destination.

In many contexts, the terms "route" and "path" mean the same thing. Most of the user documentation uses the term "route" throughout. The term "path" is used in this section to refer to an individual next-hop router to a destination, while the term "route" refers collectively to the multiple paths to the destination. Load sharing applies when the IP route table contains multiple, equal-cost paths to a destination.

NOTE: Foundry devices also perform load sharing among the ports in aggregate links. See "Trunk Group Load Sharing" on page 10-6.

How Multiple Equal-Cost Paths Enter the IP Route Table

IP load sharing applies to equal-cost paths in the IP route table. Routes that are eligible for load sharing can enter the table from any of the following sources:

- IP static routes
- Routes learned through RIP
- Routes learned through OSPF
- Routes learned through BGP4

Administrative Distance

The administrative distance is a unique value associated with each type (source) of IP route. Each path has an administrative distance. The administrative distance is not used when performing IP load sharing, but the administrative distance is used when evaluating multiple equal-cost paths to the same destination from different sources, such as RIP, OSPF and so on.

The value of the administrative distance is determined by the source of the route. The Layer 3 Switch is configured with a unique administrative distance value for each IP route source.

When the software receives multiple paths to the same destination and the paths are from different sources, the software compares the administrative distances of the paths and selects the path with the lowest distance. The software then places the path with the lowest administrative distance in the IP route table. For example, if the

1.IP load sharing is also called "Equal-Cost Multi-Path (ECMP)" load sharing or just "ECMP"

Layer 3 Switch has a path learned from OSPF and a path learned from RIP for a given destination, only the path with the lower administrative distance enters the IP route table.

Here are the default administrative distances on the Foundry Layer 3 Switch:

- Directly connected – 0 (this value is not configurable)
- Static IP route – 1 (applies to all static routes, including default routes and default network routes)
- Exterior Border Gateway Protocol (EBGP) – 20
- OSPF – 110
- RIP – 120
- Interior Gateway Protocol (IBGP) – 200
- Local BGP – 200
- Unknown – 255 (the router will not use this route)

Lower administrative distances are preferred over higher distances. For example, if the router receives routes for the same network from OSPF and from RIP, the router will prefer the OSPF route by default.

NOTE: You can change the administrative distances individually. See the configuration chapter for the route source for information.

Since the software selects only the path with the lowest administrative distance, and the administrative distance is determined by the path's source, IP load sharing does not apply to paths from different route sources. IP load sharing applies only when the IP route table contains multiple paths to the same destination, from the same IP route source.

IP load sharing does not apply to paths that come from different sources.

Path Cost

The cost parameter provides a common basis of comparison for selecting from among multiple paths to a given destination. Each path in the IP route table has a cost. When the IP route table contains multiple paths to a destination, the Layer 3 Switch chooses the path with the lowest cost. When the IP route table contains more than one path with the lowest cost to a destination, the Layer 3 Switch uses IP load sharing to select one of the lowest-cost paths.

The source of a path's cost value depends on the source of the path.

- IP static route – The value you assign to the metric parameter when you configure the route. The default metric is 1. See "Configuring Load Balancing and Redundancy Using Multiple Static Routes to the Same Destination" on page 16-36.
- RIP – The number of next-hop routers to the destination.
- OSPF – The Path Cost associated with the path. The paths can come from any combination of inter-area, intra-area, and external Link State Advertisements (LSAs).
- BGP4 – The path's Multi-Exit Discriminator (MED) value.

NOTE: If the path is redistributed between two or more of the above sources before entering the IP route table, the cost can increase during the redistribution due to settings in redistribution filters.

Static Route, OSPF, and BGP4 Load Sharing

IP load sharing and load sharing for static routes, OSPF routes, and BGP4 routes are individually configured. Multiple equal-cost paths for a destination can enter the IP route table only if the source of the paths is configured to support multiple equal-cost paths. For example, if BGP4 allows only one path with a given cost for a given destination, the BGP4 route table cannot contain equal-cost paths to the destination. Consequently, the IP route table will not receive multiple equal-cost paths from BGP4.

Table 16.7 lists the default and configurable maximum numbers of paths for each IP route source that can provide equal-cost paths to the IP route table. The table also lists where to find configuration information for the route source's load sharing parameters.

The load sharing state for all the route sources is based on the state of IP load sharing. Since IP load sharing is enabled by default on all Foundry Layer 3 Switches, load sharing for static IP routes, RIP routes, OSPF routes, and BGP4 routes also is enabled by default.

Table 16.7: Default Load Sharing Parameters for Route Sources

Route Source	Default Maximum Number of Paths	Maximum Number of Paths	See...
Static IP route	4[a]	6[a]	16-43
RIP	4[a]	6[a]	16-43
OSPF	4	6	16-43
BGP4	1	4	21-22

a.This value depends on the value for IP load sharing, and is not separately configurable.

How IP Load Sharing Works

When the Layer 3 Switch receives traffic for a destination and the IP route table contains multiple, equal-cost paths to that destination, the device checks the IP forwarding cache for a forwarding entry for the destination. The IP forwarding cache provides a fast path for forwarding IP traffic, including load-balanced traffic. The cache contains entries that associate a destination host or network with a path (next-hop router).

- If the IP forwarding sharing cache contains a forwarding entry for the destination, the device uses the entry to forward the traffic.
- If the IP load forwarding cache does not contain a forwarding entry for the destination, the software selects a path from among the available equal-cost paths to the destination, then creates a forwarding entry in the cache based on the calculation. Subsequent traffic for the same destination uses the forwarding entry.

Response to Path State Changes

If one of the load-balanced paths to a cached destination becomes unavailable, or the IP route table receives a new equal-cost path to a cached destination, the software removes the unavailable path from the IP route table. Then the software selects a new path.

Disabling or Re-Enabling Load Sharing

To disable IP load sharing, enter the following commands:

```
FESX424 Router(config)# no ip load-sharing
```

Syntax: [no] ip load-sharing

Changing the Maximum Number of Load Sharing Paths

By default, IP load sharing allows IP traffic to be balanced across up to four equal paths. You can change the maximum number of paths the Layer 3 Switch supports to a value from 2 – 6.

For optimal results, set the maximum number of paths to a value at least as high as the maximum number of equal-cost paths your network typically contains. For example, if the Layer 3 Switch you are configuring for IP load sharing has six next-hop routers, set the maximum paths value to six.

NOTE: If the setting for the maximum number of paths is lower than the actual number of equal-cost paths, the software does not use all the paths for load sharing.

To change the number of IP load sharing paths, enter a command such as the following:

```
FastIron SuperX Router(config)# ip load-sharing 6
```

Syntax: [no] ip load-sharing [<num>]

The <num> parameter specifies the number of paths and can be from 2 – 6.

Configuring IRDP

The ICMP Router Discovery Protocol (IRDP) is used by Foundry Layer 3 Switches to advertise the IP addresses of its router interfaces to directly attached hosts. IRDP is disabled by default. You can enable the feature on a global basis or on an individual port basis.

- If you enable the feature globally, all ports use the default values for the IRDP parameters.
- If you leave the feature disabled globally but enable it on individual ports, you also can configure the IRDP parameters on an individual port basis.

NOTE: You can configure IRDP parameters only an individual port basis. To do so, IRDP must be disabled globally and enabled only on individual ports. You cannot configure IRDP parameters if the feature is globally enabled.

When IRDP is enabled, the Layer 3 Switch periodically sends Router Advertisement messages out the IP interfaces on which the feature is enabled. The messages advertise the Layer 3 Switch's IP addresses to directly attached hosts who listen for the messages. In addition, hosts can be configured to query the Layer 3 Switch for the information by sending Router Solicitation messages.

Some types of hosts use the Router Solicitation messages to discover their default gateway. When IRDP is enabled on the Foundry Layer 3 Switch, the Layer 3 Switch responds to the Router Solicitation messages. Some clients interpret this response to mean that the Layer 3 Switch is the default gateway. If another router is actually the default gateway for these clients, leave IRDP disabled on the Foundry Layer 3 Switch.

IRDP uses the following parameters. If you enable IRDP on individual ports instead of enabling the feature globally, you can configure these parameters on an individual port basis.

- Packet type – The Layer 3 Switch can send Router Advertisement messages as IP broadcasts or as IP multicasts addressed to IP multicast group 224.0.0.1. The packet type is IP broadcast.
- Maximum message interval and minimum message interval – When IRDP is enabled, the Layer 3 Switch sends the Router Advertisement messages every 450 – 600 seconds by default. The time within this interval that the Layer 3 Switch selects is random for each message and is not affected by traffic loads or other network factors. The random interval minimizes the probability that a host will receive Router Advertisement messages from other routers at the same time. The interval on each IRDP-enabled Layer 3 Switch interface is independent of the interval on other IRDP-enabled interfaces. The default maximum message interval is 600 seconds. The default minimum message interval is 450 seconds.
- Hold time – Each Router Advertisement message contains a hold time value. This value specifies the maximum amount of time the host should consider an advertisement to be valid until a newer advertisement arrives. When a new advertisement arrives, the hold time is reset. The hold time is always longer than the maximum advertisement interval. Therefore, if the hold time for an advertisement expires, the host can reasonably conclude that the router interface that sent the advertisement is no longer available. The default hold time is three times the maximum message interval.
- Preference – If a host receives multiple Router Advertisement messages from different routers, the host selects the router that sent the message with the highest preference as the default gateway. The preference can be a number from -4294967296 to 4294967295. The default is 0.

Enabling IRDP Globally

To globally enable IRDP, enter the following command:

```
FastIron SuperX Router(config)# ip irdp
```

This command enables IRDP on the IP interfaces on all ports. Each port uses the default values for the IRDP parameters. The parameters are not configurable when IRDP is globally enabled.

Enabling IRDP on an Individual Port

To enable IRDP on an individual interface and change IRDP parameters, enter commands such as the following:

```
FastIron SuperX Router(config)# interface ethernet 1/3
FastIron SuperX Router(config-if-1/3)# ip irdp maxadvertinterval 400
```

This example shows how to enable IRDP on a specific port and change the maximum advertisement interval for Router Advertisement messages to 400 seconds.

NOTE: To enable IRDP on individual ports, you must leave the feature globally disabled.

Syntax: [no] ip irdp [broadcast | multicast] [holdtime <seconds>] [maxadvertinterval <seconds>] [minadvertinterval <seconds>] [preference <number>]

The **broadcast | multicast** parameter specifies the packet type the Layer 3 Switch uses to send Router Advertisement.

- **broadcast** – The Layer 3 Switch sends Router Advertisement as IP broadcasts. This is the default.
- **multicast** – The Layer 3 Switch sends Router Advertisement as multicast packets addressed to IP multicast group 224.0.0.1.

The **holdtime** <seconds> parameter specifies how long a host that receives a Router Advertisement from the Layer 3 Switch should consider the advertisement to be valid. When a host receives a new Router Advertisement message from the Layer 3 Switch, the host resets the hold time for the Layer 3 Switch to the hold time specified in the new advertisement. If the hold time of an advertisement expires, the host discards the advertisement, concluding that the router interface that sent the advertisement is no longer available. The value must be greater than the value of the **maxadvertinterval** parameter and cannot be greater than 9000. The default is three times the value of the **maxadvertinterval** parameter.

The **maxadvertinterval** parameter specifies the maximum amount of time the Layer 3 Switch waits between sending Router Advertisements. You can specify a value from 1 to the current value of the **holdtime** parameter. The default is 600 seconds.

The **minadvertinterval** parameter specifies the minimum amount of time the Layer 3 Switch can wait between sending Router Advertisements. The default is three-fourths (0.75) the value of the **maxadvertinterval** parameter. If you change the **maxadvertinterval** parameter, the software automatically adjusts the **minadvertinterval** parameter to be three-fourths the new value of the **maxadvertinterval** parameter. If you want to override the automatically configured value, you can specify an interval from 1 to the current value of the **maxadvertinterval** parameter.

The **preference** <number> parameter specifies the IRDP preference level of this Layer 3 Switch. If a host receives Router Advertisements from multiple routers, the host selects the router interface that sent the message with the highest interval as the host's default gateway. The valid range is -4294967296 to 4294967295. The default is 0.

Configuring RARP

The Reverse Address Resolution Protocol (RARP) provides a simple mechanism for directly-attached IP hosts to boot over the network. RARP allows an IP host that does not have a means of storing its IP address across power cycles or software reloads to query a directly-attached router for an IP address.

RARP is enabled by default. However, you must create a RARP entry for each host that will use the Layer 3 Switch for booting. A RARP entry consists of the following information:

- The entry number – the entry's sequence number in the RARP table.

- The MAC address of the boot client.
- The IP address you want the Layer 3 Switch to give to the client.

When a client sends a RARP broadcast requesting an IP address, the Layer 3 Switch responds to the request by looking in the RARP table for an entry that contains the client's MAC address:

- If the RARP table contains an entry for the client, the Layer 3 Switch sends a unicast response to the client that contains the IP address associated with the client's MAC address in the RARP table.
- If the RARP table does not contain an entry for the client, the Layer 3 Switch silently discards the RARP request and does not reply to the client.

How RARP Differs from BootP/DHCP

RARP and BootP/DHCP are different methods for providing IP addresses to IP hosts when they boot. These methods differ in the following ways:

- Location of configured host addresses
  - RARP requires static configuration of the host IP addresses on the Layer 3 Switch. The Layer 3 Switch replies directly to a host's request by sending an IP address you have configured in the RARP table.
  - The Layer 3 Switch forwards BootP and DHCP requests to a third-party BootP/DHCP server that contains the IP addresses and other host configuration information.
- Connection of host to boot source (Layer 3 Switch or BootP/DHCP server):
  - RARP requires the IP host to be directly attached to the Layer 3 Switch.
  - An IP host and the BootP/DHCP server can be on different networks and on different routers, so long as the routers are configured to forward ("help") the host's boot request to the boot server.
  - You can centrally configure other host parameters on the BootP/DHCP server, in addition to the IP address, and supply those parameters to the host along with its IP address.

To configure the Layer 3 Switch to forward BootP/DHCP requests when boot clients and the boot servers are on different sub-nets on different Layer 3 Switch interfaces, see "Configuring BootP/DHCP Forwarding Parameters" on page 16-49.

Disabling RARP

RARP is enabled by default. To disable RARP, enter the following command at the global CONFIG level:

```
FESX424 Router(config)# no ip rarp
```

Syntax: [no] ip rarp

To re-enable RARP, enter the following command:

```
FESX424 Router(config)# ip rarp
```

Creating Static RARP Entries

You must configure the RARP entries for the RARP table. The Layer 3 Switch can send an IP address in reply to a client's RARP request only if create a RARP entry for that client.

To assign a static IP RARP entry for static routes on a Foundry router, enter a command such as the following:

```
FESX424 Router(config)# rarp 1 1245.7654.2348 192.53.4.2
```

This command creates a RARP entry for a client with MAC address 1245.7654.2348. When the Layer 3 Switch receives a RARP request from this client, the Layer 3 Switch replies to the request by sending IP address 192.53.4.2 to the client.

Syntax: rarp <number> <mac-addr>.<ip-addr>

The <number> parameter identifies the RARP entry number. You can specify an unused number from 1 to the maximum number of RARP entries supported on the device. To determine the maximum number of entries supported on the device, see the section "Displaying and Modifying System Parameter Default Settings" on page 4-8.

The <mac-addr> parameter specifies the MAC address of the RARP client.

The <ip-addr> parameter specifies the IP address the Layer 3 Switch will give the client in response to the client's RARP request.

Changing the Maximum Number of Static RARP Entries Supported

The number of RARP entries the Layer 3 Switch supports depends on how much memory the Layer 3 Switch has. To determine how many RARP entries your Layer 3 Switch can have, display the system default information using the procedure in the section "Displaying and Modifying System Parameter Default Settings" on page 4-8.

If your Layer 3 Switch allows you to increase the maximum number of RARP entries, you can use a procedure in the same section to do so.

NOTE: You must save the configuration to the startup-config file and reload the software after changing the RARP cache size to place the change into effect.

Configuring UDP Broadcast and IP Helper Parameters

Some applications rely on client requests sent as limited IP broadcasts addressed to the UDP's application port. If a server for the application receives such a broadcast, the server can reply to the client. Routers do not forward subnet directed broadcasts, so the client and server must be on the same network for the broadcast to reach the server. If the client and server are on different networks (on opposite sides of a router), the client's request cannot reach the server.

You can configure the Layer 3 Switch to forward clients' requests to UDP application servers. To do so:

- Enable forwarding support for the UDP application port, if forwarding support is not already enabled.
- Configure a helper adders on the interface connected to the clients. Specify the helper address to be the IP address of the application server or the subnet directed broadcast address for the IP sub-net the server is in. A helper address is associated with a specific interface and applies only to client requests received on that interface. The Layer 3 Switch forwards client requests for any of the application ports the Layer 3 Switch is enabled to forward to the helper address.

Forwarding support for the following application ports is enabled by default.

- bootps (port 67)
- dns (port 53)
- tftp (port 69)
- time (port 37)
- netbios-ns (port 137)
- netbios-dgm (port 138)
- tacacs (port 65)

NOTE: The application names are the names for these applications that the Layer 3 Switch software recognizes, and might not match the names for these applications on some third-party devices. The numbers listed in parentheses are the UDP port numbers for the applications. The numbers come from RFC 1340.

NOTE: As shown above, forwarding support for BootP/DHCP is enabled by default. If you are configuring the Layer 3 Switch to forward BootP/DHCP requests, see "Configuring BootP/DHCP Forwarding Parameters" on page 16-49.

You can enable forwarding for other applications by specifying the application port number.

You also can disable forwarding for an application.

NOTE: If you disable forwarding for a UDP application, forwarding of client requests received as broadcasts to helper addresses is disabled. Disabling forwarding of an application does not disable other support for the application. For example, if you disable forwarding of Telnet requests to helper addresses, other Telnet support on the Layer 3 Switch is not also disabled.

Enabling Forwarding for a UDP Application

If you want the Layer 3 Switch to forward client requests for UDP applications that the Layer 3 Switch does not forward by default, you can enable forwarding support for the port. To enable forwarding support for a UDP application, use the following method. You also can disable forwarding for an application using this method.

NOTE: You also must configure a helper address on the interface that is connected to the clients for the application. The Layer 3 Switch cannot forward the requests unless you configure the helper address. See "Configuring an IP Helper Address" on page 16-50.

To enable the forwarding of SNMP trap broadcasts, enter the following command:

```
FastIron SuperX Router(config)# ip forward-protocol udp snmp-trap
```

Syntax: [no] ip forward-protocol udp <udp-port-name> | <udp-port-num>

The <udp-port-name> parameter can have one of the following values. For reference, the corresponding port numbers from RFC 1340 are shown in parentheses. If you specify an application name, enter the name only, not the parentheses or the port number shown here.

- bootpc (port 68)
- bootps (port 67)
- discard (port 9)
- dns (port 53)
- dnsix (port 90)
- echo (port 7)
- mobile-ip (port 434)
- netbios-dgm (port 138)
- netbios-ns (port 137)
- ntp (port 123)
- tacacs (port 65)
- talk (port 517)
- time (port 37)
- tftp (port 69)

In addition, you can specify any UDP application by using the application's UDP port number.

The <udp-port-num> parameter specifies the UDP application port number. If the application you want to enable is not listed above, enter the application port number. You also can list the port number for any of the applications listed above.

To disable forwarding for an application, enter a command such as the following:

```
FastIron SuperX Router(config)# no ip forward-protocol udp snmp
```

This command disables forwarding of SNMP requests to the helper addresses configured on Layer 3 Switch interfaces.

Configuring an IP Helper Address

To forward a client's broadcast request for a UDP application when the client and server are on different networks, you must configure a helper address on the interface connected to the client. Specify the server's IP address or the subnet directed broadcast address of the IP sub-net the server is in as the helper address.

You can configure up to 16 helper addresses on each interface. You can configure a helper address on an Ethernet port or a virtual interface.

To configure a helper address on interface 2 on chassis module 1, enter the following commands:

```
FastIron SuperX Router(config)# interface e 1/2
FastIron SuperX Router(config-if-1/2)# ip helper-address 1 207.95.7.6
```

The commands in this example change the CLI to the configuration level for port 1/2, then add a helper address for server 207.95.7.6 to the port. If the port receives a client request for any of the applications that the Layer 3 Switch is enabled to forward, the Layer 3 Switch forwards the client's request to the server.

Syntax: ip helper-address <num> <ip-addr>

The <num> parameter specifies the helper address number and can be from 1 – 16.

The <ip-addr> command specifies the server's IP address or the subnet directed broadcast address of the IP sub-net the server is in.

Configuring BootP/DHCP Forwarding Parameters

A host on an IP network can use BootP/DHCP to obtain its IP address from a BootP/DHCP server. To obtain the address, the client sends a BootP/DHCP request. The request is a subnet directed broadcast and is addressed to UDP port 67. A limited IP broadcast is addressed to IP address 255.255.255.255 and is not forwarded by the Foundry Layer 3 Switch or other IP routers.

When the BootP/DHCP client and server are on the same network, the server receives the broadcast request and replies to the client. However, when the client and server are on different networks, the server does not receive the client's request, because the Layer 3 Switch does not forward the request.

You can configure the Layer 3 Switch to forward BootP/DHCP requests. To do so, configure a helper address on the interface that receives the client requests, and specify the BootP/DHCP server's IP address as the address you are helping the BootP/DHCP requests to reach. Instead of the server's IP address, you can specify the subnet directed broadcast address of the IP sub-net the server is in.

BootP/DHCP Forwarding Parameters

The following parameters control the Layer 3 Switch's forwarding of BootP/DHCP requests:

- Helper address – The BootP/DHCP server's IP address. You must configure the helper address on the interface that receives the BootP/DHCP requests from the client. The Layer 3 Switch cannot forward a request to the server unless you configure a helper address for the server.
- Gateway address – The Layer 3 Switch places the IP address of the interface that received the BootP/DHCP request in the request packet's Gateway Address field (sometimes called the Router ID field). When the server responds to the request, the server sends the response as a unicast packet to the IP address in the Gateway Address field. (If the client and server are directly attached, the Gateway ID field is empty and the server replies to the client using a unicast or broadcast packet, depending on the server.)

  By default, the Layer 3 Switch uses the lowest-numbered IP address on the interface that receives the request as the Gateway address. You can override the default by specifying the IP address you want the Layer 3 Switch to use.
- Hop Count – Each router that forwards a BootP/DHCP packet increments the hop count by 1. Routers also discard a forwarded BootP/DHCP request instead of forwarding the request if the hop count is greater than the maximum number of BootP/DHCP hops allows by the router. By default, a Foundry Layer 3 Switch forwards a BootP/DHCP request if its hop count is four or less, but discards the request if the hop count is greater than four. You can change the maximum number of hops the Layer 3 Switch will allow to a value from 1 – 15.

NOTE: The BootP/DHCP hop count is not the TTL parameter.

Configuring an IP Helper Address

The procedure for configuring a helper address for BootP/DHCP requests is the same as the procedure for configuring a helper address for other types of UDP broadcasts. See “Configuring an IP Helper Address” on page 16-49.

Changing the IP Address Used for Stamping BootP/DHCP Requests

When the Layer 3 Switch forwards a BootP/DHCP request, the Layer 3 Switch “stamps” the Gateway Address field. The default value the Layer 3 Switch uses to stamp the packet is the lowest-numbered IP address configured on the interface that received the request. If you want the Layer 3 Switch to use a different IP address to stamp requests received on the interface, use either of the following methods to specify the address.

The BootP/DHCP stamp address is an interface parameter. Change the parameter on the interface that is connected to the BootP/DHCP client.

To change the IP address used for stamping BootP/DHCP requests received on interface 1/1, enter commands such as the following:

```
FastIron SuperX Router(config)# int e 1/1
FastIron SuperX Router(config-if-1/1)# ip bootp-gateway 109.157.22.26
```

These commands change the CLI to the configuration level for port 1/1, then change the BootP/DHCP stamp address for requests received on port 1/1 to 192.157.22.26. The Layer 3 Switch will place this IP address in the Gateway Address field of BootP/DHCP requests that the Layer 3 Switch receives on port 1/1 and forwards to the BootP/DHCP server.

Syntax: ip bootp-gateway <ip-addr>

Changing the Maximum Number of Hops to a BootP Relay Server

Each BootP/DHCP request includes a field Hop Count field. The Hop Count field indicates how many routers the request has passed through. When the Layer 3 Switch receives a BootP/DHCP request, the Layer 3 Switch looks at the value in the Hop Count field.

- If the hop count value is equal to or less than the maximum hop count the Layer 3 Switch allows, the Layer 3 Switch increments the hop count by one and forwards the request.
- If the hop count is greater than the maximum hop count the Layer 3 Switch allows, the Layer 3 Switch discards the request.

To change the maximum number of hops the Layer 3 Switch allows for forwarded BootP/DHCP requests, use either of the following methods.

NOTE: The BootP/DHCP hop count is not the TTL parameter.

To modify the maximum number of BootP/DHCP hops, enter the following command:

```
FESX424 Router(config)# bootp-relay-max-hops 10
```

This command allows the Layer 3 Switch to forward BootP/DHCP requests that have passed through up to ten previous hops before reaching the Layer 3 Switch.

Syntax: bootp-relay-max-hops <1-15>

Configuring IP Parameters – Layer 2 Switches

The following sections describe how to configure IP parameters on a Foundry Layer 2 Switch.

NOTE: This section describes how to configure IP parameters for Layer 2 Switches. For IP configuration information for Layer 3 Switches, see "Configuring IP Parameters – Layer 3 Switches" on page 16-17.

Configuring the Management IP Address and Specifying the Default Gateway

To manage a Layer 2 Switch using Telnet or Secure Shell (SSH) CLI connections or the Web management interface, you must configure an IP address for the Layer 2 Switch. Optionally, you also can specify the default gateway.

Foundry devices support both classical IP network masks (Class A, B, and C sub-net masks, and so on) and Classless Interdomain Routing (CIDR) network prefix masks.

- To enter a classical network mask, enter the mask in IP address format. For example, enter "209.157.22.99 255.255.255.0" for an IP address with a Class-C sub-net mask.
- To enter a prefix network mask, enter a forward slash (/) and the number of bits in the mask immediately after the IP address. For example, enter "209.157.22.99/24" for an IP address that has a network mask with 24 significant bits (ones).

By default, the CLI displays network masks in classical IP address format (example: 255.255.255.0). You can change the display to prefix format. See "Changing the Network Mask Display to Prefix Format" on page 16-57.

To assign an IP address to a Foundry Layer 2 Switch, enter a command such as the following at the global CONFIG level:

```
FESX424 Switch(config)# ip address 192.45.6.110 255.255.255.0
```

Syntax: ip address <ip-addr> <ip-mask>

or

Syntax: ip address <ip-addr>/<mask-bits>

NOTE: You also can enter the IP address and mask in CIDR format, as follows:

```
FESX424 Switch(config)# ip address 192.45.6.1/24
```

To specify the Layer 2 Switch's default gateway, enter a command such as the following:

```
FESX424 Switch(config)# ip default-gateway 192.45.6.1 255.255.255.0
```

Syntax: ip default-gateway <ip-addr>

or

Syntax: ip default-gateway <ip-addr>/<mask-bits>

Configuring Domain Name Server (DNS) Resolver

The Domain Name Server (DNS) resolver feature lets you use a host name to perform Telnet, ping, and traceroute commands. You can also define a DNS domain on a Foundry Layer 2 Switch or Layer 3 Switch and thereby recognize all hosts within that domain. After you define a domain name, the Foundry Layer 2 Switch or Layer 3 Switch automatically appends the appropriate domain to the host and forwards it to the domain name server.

For example, if the domain "newyork.com" is defined on a Foundry Layer 2 Switch or Layer 3 Switch and you want to initiate a ping to host "NYC01" on that domain, you need to reference only the host name in the command instead of the host name and its domain name. For example, you could enter either of the following commands to initiate the ping:

```
FESX424 Switch# ping nyc01
FESX424 Switch# ping nyc01.newyork.com
```


Defining a DNS Entry

You can define up to four DNS servers for each DNS entry. The first entry serves as the primary default address. If a query to the primary address fails to be resolved after three attempts, the next gateway address is queried (also up to three times). This process continues for each defined gateway address until the query is resolved. The order in which the default gateway addresses are polled is the same as the order in which you enter them.

Suppose you want to define the domain name of newyork.com on a Layer 2 Switch and then define four possible default DNS gateway addresses. To do so, enter the following commands:

```
FESX424 Switch(config)# ip dns domain-name newyork.com
FESX424 Switch(config)# ip dns server-address 209.157.22.199 205.96.7.15 208.95.7.25
201.98.7.15
```

Syntax: ip dns server-address <ip-addr> [<ip-addr>] [<ip-addr>] [<ip-addr>]

In this example, the first IP address in the **ip dns server-address...** command becomes the primary gateway address and all others are secondary addresses. Because IP address 201.98.7.15 is the last address listed, it is also the last address consulted to resolve a query.

Using a DNS Name To Initiate a Trace Route

EXAMPLE:

Suppose you want to trace the route from a Foundry Layer 2 Switch to a remote server identified as NYC02 on domain newyork.com. Because the newyork.com domain is already defined on the Layer 2 Switch, you need to enter only the host name, NYC02, as noted below.

```
FESX424 Switch# traceroute nyc02
```

Syntax: traceroute <host-ip-addr> [maxttl <value>] [minttl <value>] [numeric] [timeout <value>] [source-ip <ip addr>]

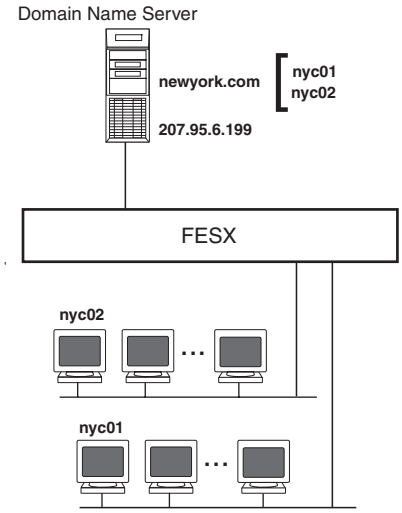
The only required parameter is the IP address of the host at the other end of the route. See the *Foundry Switch and Router Command Line Interface Reference* for information about the parameters.

After you enter the command, a message indicating that the DNS query is in process and the current gateway address (IP address of the domain name server) being queried appear on the screen:

```
Type Control-c to abort
Sending DNS Query to 209.157.22.199
Tracing Route to IP node 209.157.22.80
To ABORT Trace Route, Please use stop-traceroute command.
 Traced route to target IP node 209.157.22.80:
   IP Address          Round Trip Time1    Round Trip Time2
  207.95.6.30         93 msec             121 msec
```

NOTE: In the above example, 209.157.22.199 is the IP address of the domain name server (default DNS gateway address), and 209.157.22.80 represents the IP address of the NYC02 host.

Figure 16.5 Querying a host on the newyork.com domain



Changing the TTL Threshold

The TTL threshold prevents routing loops by specifying the maximum number of router hops an IP packet originated by the Layer 2 Switch can travel through. Each device capable of forwarding IP that receives the packet decrements (decreases) the packet's TTL by one. If a router receives a packet with a TTL of 1 and reduces the TTL to zero, the router drops the packet.

The default TTL is 64. You can change the TTL to a value from 1 – 255.

To modify the TTL threshold to 25, enter the following commands:

```
FESX424 Switch(config)# ip ttl 25
FESX424 Switch(config)# exit
```

Syntax: ip ttl <1-255>

Configuring DHCP Assist

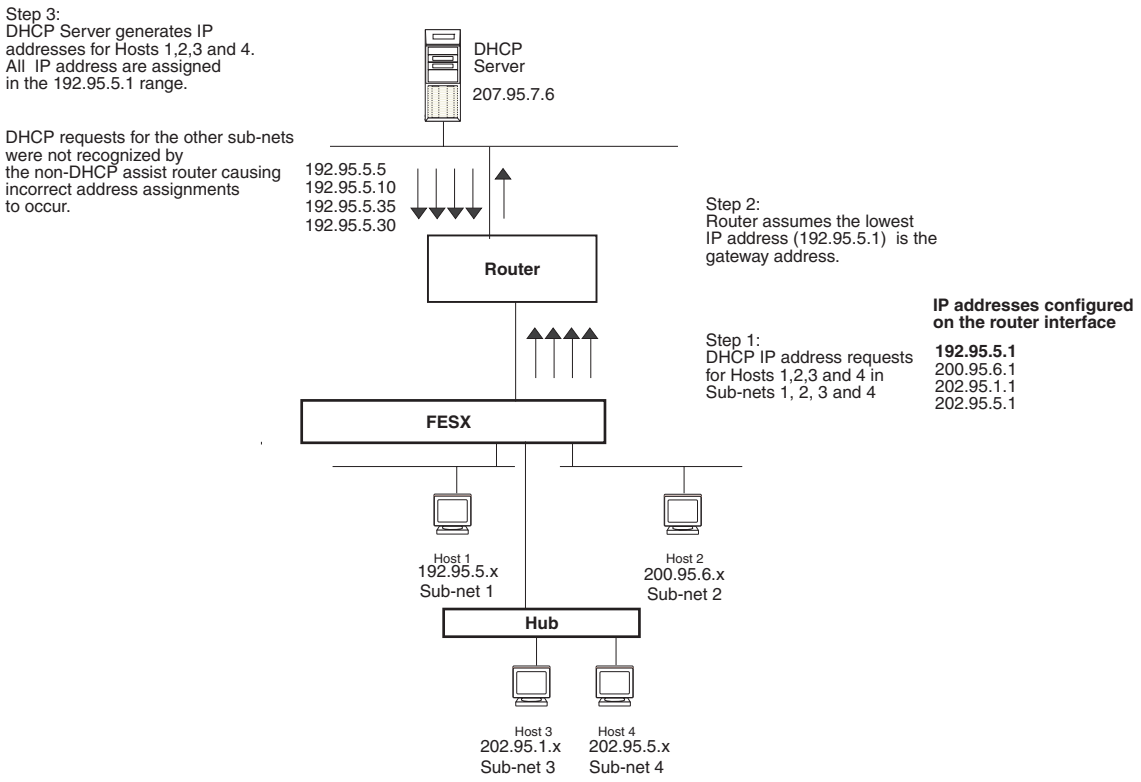
DHCP Assist allows a Foundry Layer 2 Switch to assist a router that is performing multi-netting on its interfaces as part of its DHCP relay function.

DHCP Assist ensures that a DHCP server that manages multiple IP sub-nets can readily recognize the requester's IP sub-net, even when that server is not on the client's local LAN segment. The Foundry Layer 2 Switch does so by stamping each request with its IP gateway address in the DHCP discovery packet.

NOTE: Foundry Layer 3 Switches provide BootP/DHCP assistance by default on an individual port basis. See "Changing the IP Address Used for Stamping BootP/DHCP Requests" on page 16-50.

By allowing multiple sub-net DHCP requests to be sent on the same wire, you can reduce the number of router ports required to support secondary addressing as well as reduce the number of DHCP servers required, by allowing a server to manage multiple sub-net address assignments.

Figure 16.6 DHCP requests in a network without DHCP Assist on the Layer 2 Switch



In a network operating without DHCP Assist, hosts can be assigned IP addresses from the wrong sub-net range because a router with multiple sub-nets configured on an interface cannot distinguish among DHCP discovery packets received from different sub-nets.

For example, in Figure 16.6 a host from each of the four sub-nets supported on a Layer 2 Switch requests an IP address from the DHCP server. These requests are sent transparently to the router. Because the router is unable to determine the origin of each packet by sub-net, it assumes the lowest IP address or the 'primary address' is the gateway for all ports on the Layer 2 Switch and stamps the request with that address.

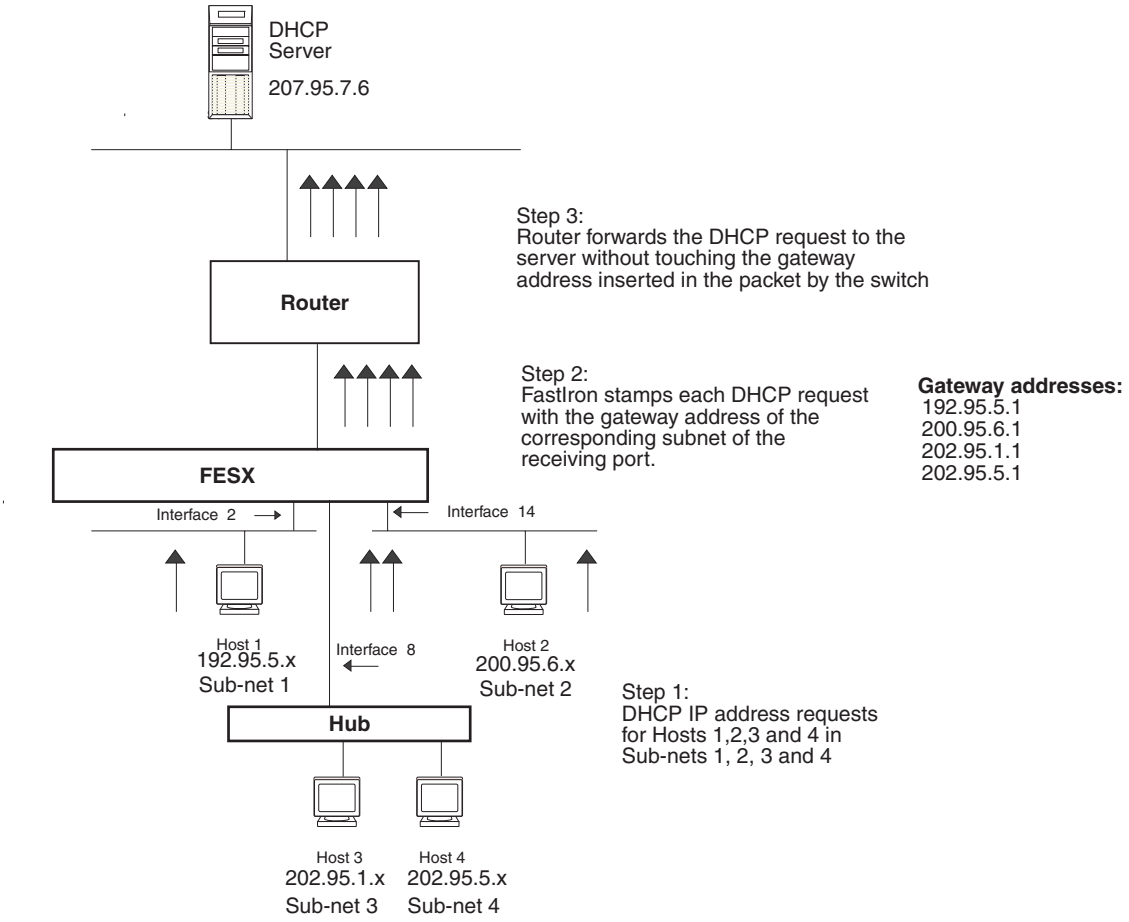
When the DHCP request is received at the server, it assigns all IP addresses within that range only.

With DHCP Assist enabled on a Foundry Layer 2 Switch, correct assignments are made because the Layer 2 Switch provides the stamping service.

How DHCP Assist Works

Upon initiation of a DHCP session, the client sends out a DHCP discovery packet for an address from the DHCP server as seen in Figure 16.7. When the DHCP discovery packet is received at a Foundry Layer 2 Switch with the DHCP Assist feature enabled, the gateway address configured on the receiving interface is inserted into the packet. This address insertion is also referred to as stamping.

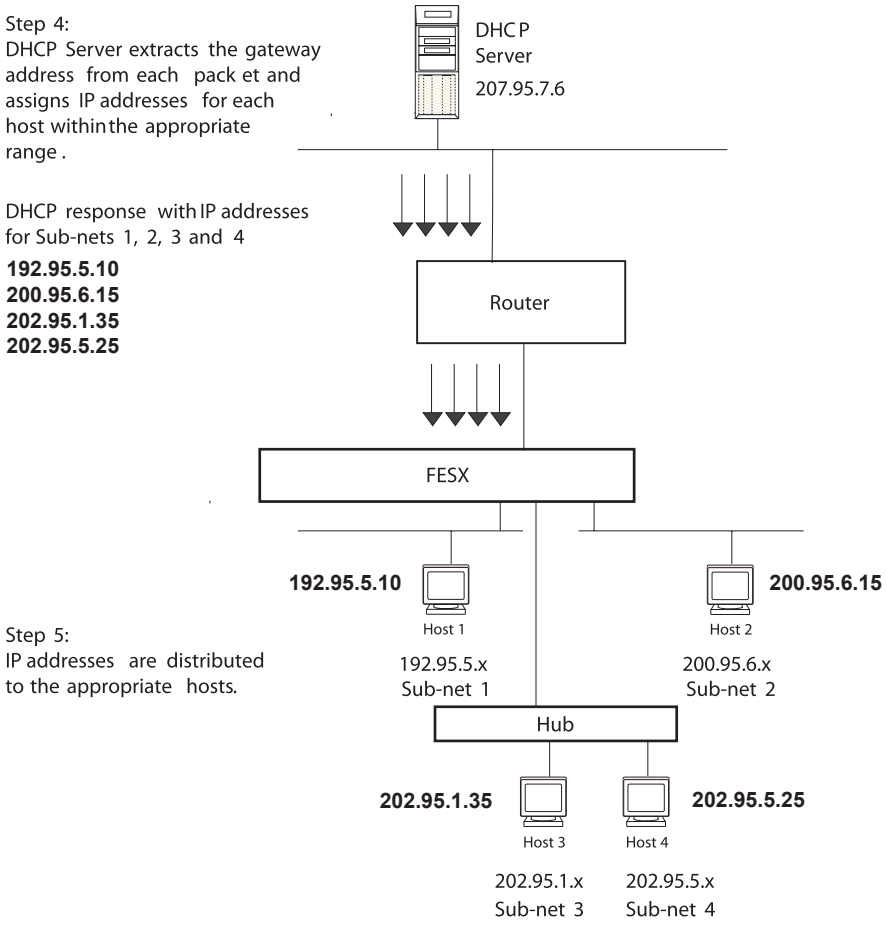
Figure 16.7 DHCP requests in a network with DHCP Assist operating on a FastIron



When the stamped DHCP discovery packet is then received at the router, it is forwarded to the DHCP server. The DHCP server then extracts the gateway address from each request and assigns an available IP address within the corresponding IP sub-net (Figure 16.8). The IP address is then forwarded back to the workstation that originated the request.

NOTE: The DHCP relay function of the connecting router needs to be turned on.

Figure 16.8 DHCP offers are forwarded back toward the requestors



Configuring DHCP Assist

You can associate a gateway list with a port. You must configure a gateway list when DHCP Assist is enabled on a Foundry Layer 2 Switch. The gateway list contains a gateway address for each sub-net that will be requesting addresses from a DHCP server. The list allows the stamping process to occur. Each gateway address defined on the Layer 2 Switch corresponds to an IP address of the Foundry router interface or other router involved.

Up to eight addresses can be defined for each gateway list in support of ports that are multi-homed. When multiple IP addresses are configured for a gateway list, the Layer 2 Switch inserts the addresses into the discovery packet in a round robin fashion.

Up to 32 gateway lists can be defined for each Layer 2 Switch.

EXAMPLE:

To create the configuration indicated in Figure 16.7 and Figure 16.8:

```
FESX424 Switch(config)# dhcp-gateway-list 1 192.95.5.1
FESX424 Switch(config)# dhcp-gateway-list 2 200.95.6.1
FESX424 Switch(config)# dhcp-gateway-list 3 202.95.1.1 202.95.5.1
FESX424 Switch(config)# int e 2
FESX424 Switch(config-if-e1000-2)# dhcp-gateway-list 1
FESX424 Switch(config-if-e1000-2)# int e8
FESX424 Switch(config-if-e1000-8)# dhcp-gateway-list 3
FESX424 Switch(config-if-e1000-8)# int e 14
FESX424 Switch(config-if-e1000-14)# dhcp-gateway-list 2
```


Syntax: dhcp-gateway-list <num> <ip-addr>

Displaying IP Configuration Information and Statistics

The following sections describe IP display options for Layer 3 Switches and Layer 2 Switches.

- To display IP information on a Layer 3 Switch, see “Displaying IP Information – Layer 3 Switches” on page 16-57.
- To display IP information on a Layer 2 Switch, see “Displaying IP Information – Layer 2 Switches” on page 16-73.

Changing the Network Mask Display to Prefix Format

By default, the CLI displays network masks in classical IP address format (example: 255.255.255.0). You can change the displays to prefix format (example: /18) on a Layer 3 Switch or Layer 2 Switch using the following CLI method.

NOTE: This option does not affect how information is displayed in the Web management interface.

To enable CIDR format for displaying network masks, entering the following command at the global CONFIG level of the CLI:

```
FESX424 Router(config)# ip show-subnet-length
```

Syntax: [no] ip show-subnet-length

Displaying IP Information – Layer 3 Switches

You can display the following IP configuration information statistics on Layer 3 Switches:

- Global IP parameter settings and IP access policies – see “Displaying Global IP Configuration Information” on page 16-58.
- CPU utilization statistics – see “Displaying CPU Utilization Statistics” on page 16-60.
- IP interfaces – see “Displaying IP Interface Information” on page 16-62.
- ARP entries – see “Displaying ARP Entries” on page 16-63.
- Static ARP entries – see “Displaying ARP Entries” on page 16-63.
- IP forwarding cache – see “Displaying the Forwarding Cache” on page 16-66.
- IP route table – see “Displaying the IP Route Table” on page 16-67.
- IP traffic statistics – see “Displaying IP Traffic Statistics” on page 16-70.

The sections below describe how to display this information.

In addition to the information described below, you can display the following IP information. This information is described in other parts of this guide.

- RIP information – see “Displaying RIP Filters” on page 17-10.
- OSPF information – see “Displaying OSPF Information” on page 20-37.
- BGP4 information – see “Displaying BGP4 Information” on page 21-65.
- DVMRP information – see the “Show Commands” chapter in the *Foundry Switch and Router Command Line Interface Reference*.
- PIM information – see the “Show Commands” chapter in the *Foundry Switch and Router Command Line Interface Reference*.
- VRRP or VRRPE information – see “Displaying VRRP and VRRPE Information” on page 22-19.

Displaying Global IP Configuration Information

To display IP configuration information, enter the following command at any CLI level:

```
FESX424 Router> show ip

Global Settings
  ttl: 64, arp-age: 10, bootp-relay-max-hops: 4
  router-id : 207.95.11.128
  enabled : UDP-Broadcast-Forwarding  IRDP  Proxy-ARP  RARP  OSPF
  disabled: BGP4 Load-Sharing  RIP  DVMRP  FSRP  VRRP

Static Routes
  Index   IP Address        Subnet Mask       Next Hop Router   Metric Distance
  1       0.0.0.0           0.0.0.0           209.157.23.2      1      1

Policies
  Index   Action   Source           Destination      Protocol   Port  Operator
  1       deny     209.157.22.34    209.157.22.26    tcp        http  =
  64      permit   any              any
```

Syntax: show ip

NOTE: This command has additional options, which are explained in other sections in this guide, including the sections below this one.

This display shows the following information.

Table 16.8: CLI Display of Global IP Configuration Information – Layer 3 Switch

This Field...	Displays...
Global settings	
ttl	The Time-To-Live (TTL) for IP packets. The TTL specifies the maximum number of router hops a packet can travel before reaching the Foundry router. If the packet's TTL value is higher than the value specified in this field, the Foundry router drops the packet. To change the maximum TTL, see "Changing the TTL Threshold" on page 16-29.
arp-age	The ARP aging period. This parameter specifies how many minutes an inactive ARP entry remains in the ARP cache before the router ages out the entry. To change the ARP aging period, see "Changing the ARP Aging Period" on page 16-27.
bootp-relay-max-hops	The maximum number of hops away a BootP server can be located from the Foundry router and still be used by the router's clients for network booting. To change this value, see "Changing the Maximum Number of Hops to a BootP Relay Server" on page 16-50.
router-id	The 32-bit number that uniquely identifies the Foundry router. By default, the router ID is the numerically lowest IP interface configured on the router. To change the router ID, see "Changing the Router ID" on page 16-23.
enabled	The IP-related protocols that are enabled on the router.
disabled	The IP-related protocols that are disabled on the router.
Static routes	
Index	The row number of this entry in the IP route table.
IP Address	The IP address of the route's destination.
Subnet Mask	The network mask for the IP address.
Next Hop Router	The IP address of the router interface to which the Foundry router sends packets for the route.
Metric	The cost of the route. Usually, the metric represents the number of hops to the destination.
Distance	The administrative distance of the route. The default administrative distance for static IP routes in Foundry routers is 1. To list the default administrative distances for all types of routes or to change the administrative distance of a static route, see "Changing Administrative Distances" on page 21-29.
Policies	

Table 16.8: CLI Display of Global IP Configuration Information – Layer 3 Switch (Continued)

This Field...	Displays...
Index	The policy number. This is the number you assigned the policy when you configured it.
Action	The action the router takes if a packet matches the comparison values in the policy. The action can be one of the following: • deny – The router drops packets that match this policy. • permit – The router forwards packets that match this policy.
Source	The source IP address the policy matches.
Destination	The destination IP address the policy matches.
Protocol	The IP protocol the policy matches. The protocol can be one of the following: • ICMP • IGMP • IGRP • OSPF • TCP • UDP
Port	The Layer 4 TCP or UDP port the policy checks for in packets. The port can be displayed by its number or, for port types the router recognizes, by the well-known name. For example, TCP port 80 can be displayed as HTTP. **Note**: This field applies only if the IP protocol is TCP or UDP.
Operator	The comparison operator for TCP or UDP port names or numbers. **Note**: This field applies only if the IP protocol is TCP or UDP.

Displaying CPU Utilization Statistics

You can display CPU utilization statistics for IP protocols using the **show process cpu** command.

The **show process cpu** command includes CPU utilization statistics for ACL, 802.1x, and L2VLAN. L2VLAN contains any packet transmitted to a VLAN by the CPU, including unknown unicast, multicast, broadcast, and CPU forwarded Layer 2 traffic.

To display CPU utilization statistics for the previous one-second, one-minute, five-minute, and fifteen-minute intervals, enter the following command at any level of the CLI:

```
FESX424 Router# show process cpu
Process Name    5Sec(%)    1Min(%)    5Min(%)    15Min(%)    Runtime(ms)
ACL               0.00       0.00       0.00       0.00              0
ARP               0.01       0.01       0.01       0.01            714
BGP               0.00       0.00       0.00       0.00              0
DOT1X             0.00       0.00       0.00       0.00              0
GVRP              0.00       0.00       0.00       0.00              0
ICMP              0.00       0.00       0.00       0.00            161
IP                0.00       0.00       0.00       0.00            229
L2VLAN            0.01       0.00       0.00       0.01            673
OSPF              0.00       0.00       0.00       0.00              0
RIP               0.00       0.00       0.00       0.00              9
STP               0.00       0.00       0.00       0.00              7
VRRP              0.00       0.00       0.00       0.00              0
```

If the software has been running less than 15 minutes (the maximum interval for utilization statistics), the command indicates how long the software has been running. Here is an example:

```
FESX424 Router# show process cpu
The system has only been up for 6 seconds.
Process Name    5Sec(%)    1Min(%)    5Min(%)    15Min(%)    Runtime(ms)
ACL               0.00       0.00       0.00       0.00              0
ARP               0.01       0.01       0.01       0.01            714
BGP               0.00       0.00       0.00       0.00              0
DOT1X             0.00       0.00       0.00       0.00              0
GVRP              0.00       0.00       0.00       0.00              0
ICMP              0.00       0.00       0.00       0.00            161
IP                0.00       0.00       0.00       0.00            229
L2VLAN            0.01       0.00       0.00       0.01            673
OSPF              0.00       0.00       0.00       0.00              0
RIP               0.00       0.00       0.00       0.00              9
STP               0.00       0.00       0.00       0.00              7
VRRP              0.00       0.00       0.00       0.00              0
```

To display utilization statistics for a specific number of seconds, enter a command such as the following:

```
FESX424 Router# show process cpu 2
Statistics for last 1 sec and 80 ms
Process Name    Sec(%)    Time(ms)
ACL                 0       0.00
ARP                 1       0.01
BGP                 0       0.00
DOT1X               0       0.00
GVRP                0       0.00
ICMP                0       0.00
IP                  0       0.00
L2VLAN              1       0.01
OSPF                0       0.00
RIP                 0       0.00
STP                 0       0.00
VRRP                0       0.00
```

When you specify how many seconds' worth of statistics you want to display, the software selects the sample that most closely matches the number of seconds you specified. In this example, statistics are requested for the previous two seconds. The closest sample available is actually for the previous 1 second plus 80 milliseconds.

Syntax: show process cpu [<num>]

The <num> parameter specifies the number of seconds and can be from 1 – 900. If you use this parameter, the command lists the usage statistics only for the specified number of seconds. If you do not use this parameter, the command lists the usage statistics for the previous one-second, one-minute, five-minute, and fifteen-minute intervals.

Displaying IP Interface Information

To display IP interface information, enter the following command at any CLI level:

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

Interface       IP-Address      OK?  Method    Status                Protocol
Ethernet 1/1    207.95.6.173    YES  NVRAM     up                    up
Ethernet 1/2    3.3.3.3         YES  manual    up                    up
Loopback 1      1.2.3.4         YES  NVRAM     down                  down
```

Syntax: show ip interface [ethernet [<slotnum>/]<portnum>] | [loopback <num>] | [ve <num>]

This display shows the following information.

Table 16.9: CLI Display of Interface IP Configuration Information

This Field...	Displays...
Interface	The type and the slot and port number of the interface.
IP-Address	The IP address of the interface. **Note**: If an "s" is listed following the address, this is a secondary address. When the address was configured, the interface already had an IP address in the same sub-net, so the software required the "secondary" option before the software could add the interface.

Table 16.9: CLI Display of Interface IP Configuration Information (Continued)

This Field...	Displays...
OK?	Whether the IP address has been configured on the interface.
Method	Whether the IP address has been saved in NVRAM. If you have set the IP address for the interface in the CLI or Web Management interface, but have not saved the configuration, the entry for the interface in the Method field is “manual”.
Status	The link status of the interface. If you have disabled the interface with the **disable** command, the entry in the Status field will be “administratively down”. Otherwise, the entry in the Status field will be either “up” or “down”.
Protocol	Whether the interface can provide two-way communication. If the IP address is configured, and the link status of the interface is up, the entry in the protocol field will be “up”. Otherwise the entry in the protocol field will be “down”.

To display detailed IP information for a specific interface, enter a command such as the following:

```
FastIron SuperX Router# show ip interface ethernet 1/1
Interface Ethernet 1/1
  port state: UP
  ip address: 192.168.9.51      subnet mask: 255.255.255.0
  encapsulation: ETHERNET, mtu: 1500, metric: 1
  directed-broadcast-forwarding: disabled
  proxy-arp: disabled
  ip arp-age:  10 minutes
  Ip Flow switching is disabled
  No Helper Addresses are configured.
  No inbound ip access-list is set
  No outgoing ip access-list is set
```

Displaying ARP Entries

You can display the ARP cache and the static ARP table. The ARP cache contains entries for devices attached to the Layer 3 Switch. The static ARP table contains the user-configured ARP entries. An entry in the static ARP table enters the ARP cache when the entry’s interface comes up.

The tables require separate display commands or Web management options.

Displaying the ARP Cache

To display the contents of the ARP cache, enter the following command at any CLI level:

```
FESX424 Router# show arp

Total number of ARP entries: 5
     IP Address          MAC Address          Type        Age      Port
1     207.95.6.102        0800.5afc.ea21       Dynamic     0          6
2     207.95.6.18         00a0.24d2.04ed       Dynamic     3          6
3     207.95.6.54         00a0.24ab.cd2b       Dynamic     0          6
4     207.95.6.101        0800.207c.a7fa       Dynamic     0          6
5     207.95.6.211        00c0.2638.ac9c       Dynamic     0          6
```

Syntax: show arp [ethernet [<slotnum>/]<portnum> | mac-address <xxxx.xxxx.xxxx> [<mask>] | <ip-addr> [<ip-mask>]] [<num>]

The <slotnum> parameter is required on chassis devices.

The <portnum> parameter lets you restrict the display to entries for a specific port.

The **mac-address** <xxxx.xxxx.xxxx> parameter lets you restrict the display to entries for a specific MAC address.

The <mask> parameter lets you specify a mask for the **mac-address** <xxxx.xxxx.xxxx> parameter, to display entries for multiple MAC addresses. Specify the MAC address mask as "f"s and "0"s, where "f"s are significant bits.

The <ip-addr> and <ip-mask> parameters let you restrict the display to entries for a specific IP address and network mask. Specify the IP address masks in standard decimal mask format (for example, 255.255.0.0).

NOTE: The <ip-mask> parameter and <mask> parameter perform different operations. The <ip-mask> parameter specifies the network mask for a specific IP address, whereas the <mask> parameter provides a filter for displaying multiple MAC addresses that have specific values in common.

The <num> parameter lets you display the table beginning with a specific entry number.

NOTE: The entry numbers in the ARP cache are not related to the entry numbers for static ARP table entries.

This display shows the following information. The number in the left column of the CLI display is the row number of the entry in the ARP cache. This number is not related to the number you assign to static MAC entries in the static ARP table.

Table 16.10: CLI Display of ARP Cache

This Field...	Displays...
IP Address	The IP address of the device.
MAC Address	The MAC address of the device.
Type	The type, which can be one of the following: • Dynamic – The Layer 3 Switch learned the entry from an incoming packet. • Static – The Layer 3 Switch loaded the entry from the static ARP table when the device for the entry was connected to the Layer 3 Switch.
Age	The number of minutes the entry has remained unused. If this value reaches the ARP aging period, the entry is removed from the table. To display the ARP aging period, see "Displaying Global IP Configuration Information" on page 16-58. To change the ARP aging interval, see "Changing the ARP Aging Period" on page 16-27. **Note**: Static entries do not age out.
Port	The port on which the entry was learned.

Displaying the Static ARP Table

To display the static ARP table instead of the ARP cache, enter the following command at any CLI level:

```
FastIron SuperX Router# show ip static-arp

Static ARP table size: 512, configurable from 512 to 1024
  Index    IP Address          MAC Address         Port
  1        207.95.6.111        0800.093b.d210      1/1
  3        207.95.6.123        0800.093b.d211      1/1
```

This example shows two static entries. Note that since you specify an entry's index number when you create the entry, it is possible for the range of index numbers to have gaps, as shown in this example.

NOTE: The entry number you assign to a static ARP entry is not related to the entry numbers in the ARP cache.

Syntax: show ip static-arp [ethernet [<slotnum>/]<portnum> | mac-address <xxxx.xxxx.xxxx> [<mask>] | <ip-addr> [<ip-mask>]] [<num>]

The <slotnum> parameter is required on chassis devices.

The <portnum> parameter lets you restrict the display to entries for a specific port.

The **mac-address** <xxxx.xxxx.xxxx> parameter lets you restrict the display to entries for a specific MAC address.

The <mask> parameter lets you specify a mask for the **mac-address** <xxxx.xxxx.xxxx> parameter, to display entries for multiple MAC addresses. Specify the MAC address mask as "f"s and "0"s, where "f"s are significant bits.

The <ip-addr> and <ip-mask> parameters let you restrict the display to entries for a specific IP address and network mask. Specify the IP address masks in standard decimal mask format (for example, 255.255.0.0).

NOTE: The <ip-mask> parameter and <mask> parameter perform different operations. The <ip-mask> parameter specifies the network mask for a specific IP address, whereas the <mask> parameter provides a filter for displaying multiple MAC addresses that have specific values in common.

The <num> parameter lets you display the table beginning with a specific entry number.

Table 16.11: CLI Display of Static ARP Table

This Field...	Displays...
Static ARP table size	The maximum number of static entries that can be configured on the device using the current memory allocation. The range of valid memory allocations for static ARP entries is listed after the current allocation. To change the memory allocation for static ARP entries, see "Changing the Maximum Number of Entries the Static ARP Table Can Hold" on page 16-28.
Index	The number of this entry in the table. You specify the entry number when you create the entry.
IP Address	The IP address of the device.
MAC Address	The MAC address of the device.
Port	The port attached to the device the entry is for.

Displaying the Forwarding Cache

To display the IP forwarding cache, enter the following command at any CLI level:

```
FESX424 Router> show ip cache

Total number of cache entries: 3
D:Dynamic  P:Permanent  F:Forward  U:Us  C:Complex Filter
W:Wait ARP  I:ICMP Deny  K:Drop  R:Fragment  S:Snap Encap
      IP Address       Next Hop         MAC              Type  Port  Vlan  Pri
1     192.168.1.11     DIRECT           0000.0000.0000   PU    n/a         0
2     192.168.1.255    DIRECT           0000.0000.0000   PU    n/a         0
3     255.255.255.255  DIRECT           0000.0000.0000   PU    n/a         0
```

Syntax: show ip cache [<ip-addr>] | [<num>]

The <ip-addr> parameter displays the cache entry for the specified IP address.

The <num> parameter displays the cache beginning with the row following the number you enter. For example, to begin displaying the cache at row 10, enter the following command: **show ip cache 9**.

The **show ip cache** command displays the following information.

Table 16.12: CLI Display of IP Forwarding Cache – Layer 3 Switch

This Field...	Displays...
IP Address	The IP address of the destination.
Next Hop	The IP address of the next-hop router to the destination. This field contains either an IP address or the value DIRECT. DIRECT means the destination is either directly attached or the destination is an address on this Foundry device. For example, the next hop for loopback addresses and broadcast addresses is shown as DIRECT.
MAC	The MAC address of the destination. **Note**: If the entry is type U (indicating that the destination is this Foundry device), the address consists of zeroes.
Type	The type of host entry, which can be one or more of the following: • D – Dynamic • P – Permanent • F – Forward • U – Us • C – Complex Filter • W – Wait ARP • I – ICMP Deny • K – Drop • R – Fragment • S – Snap Encap

Table 16.12: CLI Display of IP Forwarding Cache – Layer 3 Switch (Continued)

This Field...	Displays...
Port	The port through which this device reaches the destination. For destinations that are located on this device, the port number is shown as “n/a”.
VLAN	Indicates the VLAN(s) the listed port is in.
Pri	The QoS priority of the port or VLAN.

Displaying the IP Route Table

To display the IP route table, enter the following command at any CLI level:

```
FastIron SuperX Router> show ip route

Total number of IP routes: 514
Start index: 1  B:BGP D:Connected  R:RIP  S:Static  O:OSPF *:Candidate default

Destination       NetMask           Gateway           Port   Cost   Type
1.1.0.0           255.255.0.0       99.1.1.2          1/1    2      R
1.2.0.0           255.255.0.0       99.1.1.2          1/1    2      R
1.3.0.0           255.255.0.0       99.1.1.2          1/1    2      R
1.4.0.0           255.255.0.0       99.1.1.2          1/1    2      R
1.5.0.0           255.255.0.0       99.1.1.2          1/1    2      R
1.6.0.0           255.255.0.0       99.1.1.2          1/1    2      R
1.7.0.0           255.255.0.0       99.1.1.2          1/1    2      R
1.8.0.0           255.255.0.0       99.1.1.2          1/1    2      R
1.9.0.0           255.255.0.0       99.1.1.2          1/1    2      R
1.10.0.0          255.255.0.0       99.1.1.2          1/1    2      S
```

Syntax: show ip route [<ip-addr> [<ip-mask>] [longer] [none-bgp]] | <num> | bgp | direct | ospf | rip | static

The <ip-addr> parameter displays the route to the specified IP address.

The <ip-mask> parameter lets you specify a network mask or, if you prefer CIDR format, the number of bits in the network mask. If you use CIDR format, enter a forward slash immediately after the IP address, then enter the number of mask bits (for example: 209.157.22.0/24 for 209.157.22.0 255.255.255.0).

The **longer** parameter applies only when you specify an IP address and mask. This option displays only the routes for the specified IP address and mask. See the example below.

The **none-bgp** parameter displays only the routes that did not come from BGP4.

The <num> option display the route table entry whose row number corresponds to the number you specify. For example, if you want to display the tenth row in the table, enter “10”.

The **bgp** option displays the BGP4 routes.

The **direct** option displays only the IP routes that are directly attached to the Layer 3 Switch.

The **ospf** option displays the OSPF routes.

The **rip** option displays the RIP routes.

The **static** option displays only the static IP routes.

The default routes are displayed first.

Here is an example of how to use the **direct** option. To display only the IP routes that go to devices directly attached to the Layer 3 Switch:

```
FastIron SuperX Router(config)# show ip route direct
Start index: 1  B:BGP D:Connected  R:RIP  S:Static  O:OSPF *:Candidate default

      Destination       NetMask           Gateway           Port   Cost   Type
      209.157.22.0      255.255.255.0     0.0.0.0           4/11   1      D
```

Notice that the route displayed in this example has “D” in the Type field, indicating the route is to a directly connected device.

Here is an example of how to use the **static** option. To display only the static IP routes:

```
FastIron SuperX Router(config)# show ip route static
Start index: 1  B:BGP D:Connected  R:RIP  S:Static  O:OSPF *:Candidate default

      Destination       NetMask           Gateway           Port   Cost   Type
      192.144.33.11     255.255.255.0     209.157.22.12     1/1    2      S
```

Notice that the route displayed in this example has “S” in the Type field, indicating the route is static.

Here is an example of how to use the **longer** option. To display only the routes for a specified IP address and mask, enter a command such as the following:

```
FastIron SuperX Router(config)# show ip route 209.159.0.0/16 longer

Starting index: 1 B:BGP D:Directly-Connected R:RIP S:Static O:OSPF
Destination NetMask Gateway Port Cost Type

52 209.159.38.0 255.255.255.0 207.95.6.101 1/1 1 S
53 209.159.39.0 255.255.255.0 207.95.6.101 1/1 1 S
54 209.159.40.0 255.255.255.0 207.95.6.101 1/1 1 S
55 209.159.41.0 255.255.255.0 207.95.6.101 1/1 1 S
56 209.159.42.0 255.255.255.0 207.95.6.101 1/1 1 S
57 209.159.43.0 255.255.255.0 207.95.6.101 1/1 1 S
58 209.159.44.0 255.255.255.0 207.95.6.101 1/1 1 S
59 209.159.45.0 255.255.255.0 207.95.6.101 1/1 1 S
60 209.159.46.0 255.255.255.0 207.95.6.101 1/1 1 S
```

This example shows all the routes for networks beginning with 209.159. The mask value and **longer** parameter specify the range of network addresses to be displayed. In this example, all routes within the range 209.159.0.0 – 209.159.255.255 are listed.

The **summary** option displays a summary of the information in the IP route table. The following is an example of the output from this command:

EXAMPLE:

```
FastIron SuperX Router# show ip route summary

IP Routing Table - 35 entries:
  6 connected, 28 static, 0 RIP, 1 OSPF, 0 BGP, 0 ISIS, 0 MPLS
  Number of prefixes:
  /0: 1 /16: 27 /22: 1 /24: 5 /32: 1
```


Syntax: show ip route summary

In this example, the IP route table contains 35 entries. Of these entries, 6 are directly connected devices, 28 are static routes, and 1 route was calculated through OSPF. One of the routes has a zero-bit mask (this is the default route), 27 have a 22-bit mask, 5 have a 24-bit mask, and 1 has a 32-bit mask.

The following table lists the information displayed by the **show ip route** command.

Table 16.13: CLI Display of IP Route Table

This Field...	Displays...
Destination	The destination network of the route.
NetMask	The network mask of the destination address.
Gateway	The next-hop router.
Port	The port through which this router sends packets to reach the route's destination.
Cost	The route's cost.

Type	The route type, which can be one of the following:
	• B – The route was learned from BGP.
	• D – The destination is directly connected to this Layer 3 Switch.
	• R – The route was learned from RIP.
	• S – The route is a static route.
	• * – The route is a candidate default route.
	• O – The route is an OSPF route. Unless you use the **ospf** option to display the route table, "O" is used for all OSPF routes. If you do use the **ospf** option, the following type codes are used:
	  • O – OSPF intra area route (within the same area).
	  • IA – The route is an OSPF inter area route (a route that passes from one area into another).
	  • E1 – The route is an OSPF external type 1 route.
	  • E2 – The route is an OSPF external type 2 route.

Clearing IP Routes

If needed, you can clear the entire route table or specific individual routes.

To clear all routes from the IP route table:

```
FESX424 Router# clear ip route
```

To clear route 209.157.22.0/24 from the IP routing table:

```
FESX424 Router# clear ip route 209.157.22.0/24
```

Syntax: clear ip route [<ip-addr> <ip-mask>]

or

Syntax: clear ip route [<ip-addr>/<mask-bits>]

Displaying IP Traffic Statistics

To display IP traffic statistics, enter the following command at any CLI level:

```
FESX424 Router> show ip traffic

IP Statistics

  139 received, 145 sent, 0 forwarded
  0 filtered, 0 fragmented, 0 reassembled, 0 bad header
  0 no route, 0 unknown proto, 0 no buffer, 0 other errors

ICMP Statistics
Received:
  0 total, 0 errors, 0 unreachable, 0 time exceed
  0 parameter, 0 source quench, 0 redirect, 0 echo,
  0 echo reply, 0 timestamp, 0 timestamp reply, 0 addr mask
  0 addr mask reply, 0 irdp advertisement, 0 irdp solicitation
Sent:
  0 total, 0 errors, 0 unreachable, 0 time exceed
  0 parameter, 0 source quench, 0 redirect, 0 echo,
  0 echo reply, 0 timestamp, 0 timestamp reply, 0 addr mask
  0 addr mask reply, 0 irdp advertisement, 0 irdp solicitation

UDP Statistics
  1 received, 0 sent, 1 no port, 0 input errors

TCP Statistics
  0 active opens, 0 passive opens, 0 failed attempts
  0 active resets, 0 passive resets, 0 input errors
  138 in segments, 141 out segments, 4 retransmission

RIP Statistics
  0 requests sent, 0 requests received
  0 responses sent, 0 responses received
  0 unrecognized, 0 bad version, 0 bad addr family, 0 bad req format
  0 bad metrics, 0 bad resp format, 0 resp not from rip port
  0 resp from loopback, 0 packets rejected
```

The **show ip traffic** command displays the following information.

Table 16.14: CLI Display of IP Traffic Statistics – Layer 3 Switch

This Field...	Displays...
IP statistics	
received	The total number of IP packets received by the device.
sent	The total number of IP packets originated and sent by the device.
forwarded	The total number of IP packets received by the device and forwarded to other devices.
filtered	The total number of IP packets filtered by the device.
fragmented	The total number of IP packets fragmented by this device to accommodate the MTU of this device or of another device.

Table 16.14: CLI Display of IP Traffic Statistics – Layer 3 Switch (Continued)

This Field...	Displays...
reassembled	The total number of fragmented IP packets that this device re-assembled.
bad header	The number of IP packets dropped by the device due to a bad packet header.
no route	The number of packets dropped by the device because there was no route.
unknown proto	The number of packets dropped by the device because the value in the Protocol field of the packet header is unrecognized by this device.
no buffer	This information is used by Foundry customer support.
other errors	The number of packets that this device dropped due to error types other than the types listed above.
ICMP statistics The ICMP statistics are derived from RFC 792, “Internet Control Message Protocol”, RFC 950, “Internet Standard Subnetting Procedure”, and RFC 1256, “ICMP Router Discovery Messages”. Statistics are organized into Sent and Received. The field descriptions below apply to each.	
total	The total number of ICMP messages sent or received by the device.
errors	This information is used by Foundry customer support.
unreachable	The number of Destination Unreachable messages sent or received by the device.
time exceed	The number of Time Exceeded messages sent or received by the device.
parameter	The number of Parameter Problem messages sent or received by the device.
source quench	The number of Source Quench messages sent or received by the device.
redirect	The number of Redirect messages sent or received by the device.
echo	The number of Echo messages sent or received by the device.
echo reply	The number of Echo Reply messages sent or received by the device.
timestamp	The number of Timestamp messages sent or received by the device.
timestamp reply	The number of Timestamp Reply messages sent or received by the device.
addr mask	The number of Address Mask Request messages sent or received by the device.
addr mask reply	The number of Address Mask Replies messages sent or received by the device.
irdp advertisement	The number of ICMP Router Discovery Protocol (IRDP) Advertisement messages sent or received by the device.

Table 16.14: CLI Display of IP Traffic Statistics – Layer 3 Switch (Continued)

This Field...	Displays...
irdp solicitation	The number of IRDP Solicitation messages sent or received by the device.
UDP statistics	
received	The number of UDP packets received by the device.
sent	The number of UDP packets sent by the device.
no port	The number of UDP packets dropped because the packet did not contain a valid UDP port number.
input errors	This information is used by Foundry customer support.
TCP statistics The TCP statistics are derived from RFC 793, “Transmission Control Protocol”.	
active opens	The number of TCP connections opened by this device by sending a TCP SYN to another device.
passive opens	The number of TCP connections opened by this device in response to connection requests (TCP SYNs) received from other devices.
failed attempts	This information is used by Foundry customer support.
active resets	The number of TCP connections this device reset by sending a TCP RESET message to the device at the other end of the connection.
passive resets	The number of TCP connections this device reset because the device at the other end of the connection sent a TCP RESET message.
input errors	This information is used by Foundry customer support.
in segments	The number of TCP segments received by the device.
out segments	The number of TCP segments sent by the device.
retransmission	The number of segments that this device retransmitted because the retransmission timer for the segment had expired before the device at the other end of the connection had acknowledged receipt of the segment.
RIP statistics The RIP statistics are derived from RFC 1058, “Routing Information Protocol”.	
requests sent	The number of requests this device has sent to another RIP router for all or part of its RIP routing table.
requests received	The number of requests this device has received from another RIP router for all or part of this device’s RIP routing table.
responses sent	The number of responses this device has sent to another RIP router’s request for all or part of this device’s RIP routing table.
responses received	The number of responses this device has received to requests for all or part of another RIP router’s routing table.
unrecognized	This information is used by Foundry customer support.

Table 16.14: CLI Display of IP Traffic Statistics – Layer 3 Switch (Continued)

This Field...	Displays...
bad version	The number of RIP packets dropped by the device because the RIP version was either invalid or is not supported by this device.
bad addr family	The number of RIP packets dropped because the value in the Address Family Identifier field of the packet's header was invalid.
bad req format	The number of RIP request packets this router dropped because the format was bad.
bad metrics	This information is used by Foundry customer support.
bad resp format	The number of responses to RIP request packets this router dropped because the format was bad.
resp not from rip port	This information is used by Foundry customer support.
resp from loopback	The number of RIP responses received from loopback interfaces.
packets rejected	This information is used by Foundry customer support.

Displaying IP Information – Layer 2 Switches

You can display the following IP configuration information statistics on Layer 2 Switches:

- Global IP settings – see "Displaying Global IP Configuration Information" on page 16-73.
- ARP entries – see "Displaying ARP Entries" on page 16-74.
- IP traffic statistics – see "Displaying IP Traffic Statistics" on page 16-75.

Displaying Global IP Configuration Information

To display the Layer 2 Switch's IP address and default gateway, enter the following command from any level of the CLI:

```
FESX424 Switch(config)# show ip

     Switch IP address: 192.168.1.2

           Subnet mask: 255.255.255.0

Default router address: 192.168.1.1
   TFTP server address: None
Configuration filename: None
        Image filename: None
```

Syntax: show ip

This display shows the following information.

Table 16.15: CLI Display of Global IP Configuration Information – Layer 2 Switch

This Field...	Displays...
IP configuration	
Switch IP address	The management IP address you configured on the Layer 2 Switch. Specify this address for Telnet or Web management access.
Subnet mask	The sub-net mask for the management IP address.
Default router address	The address of the default gateway, if you specified one.
Most recent TFTP access	
TFTP server address	The IP address of the most-recently contacted TFTP server, if the Layer 2 Switch has contacted a TFTP server since the last time the software was reloaded or the Layer 2 Switch was rebooted.
Configuration filename	The name under which the Layer 2 Switch's startup-config file was uploaded or downloaded during the most recent TFTP access.
Image filename	The name of the Layer 2 Switch flash image (system software file) that was uploaded or downloaded during the most recent TFTP access.

Displaying ARP Entries

To display the entries the Layer 2 Switch has placed in its ARP cache, enter the following command from any level of the CLI:

```
FESX424 Switch(config)# show arp

         IP              Mac         Port Age VlanId
192.168.1.170       0010.5a11.d042     7   0      1
Total Arp Entries : 1
```

Syntax: show arp

This display shows the following information.

Table 16.16: CLI Display of ARP Cache

This Field...	Displays...
IP	The IP address of the device.
Mac	The MAC address of the device. **Note**: If the MAC address is all zeros, the entry is for the default gateway, but the Layer 2 Switch does not have a link to the gateway.
Port	The port on which the entry was learned.
Age	The number of minutes the entry has remained unused. If this value reaches the ARP aging period, the entry is removed from the cache.

Table 16.16: CLI Display of ARP Cache (Continued)

This Field...	Displays...
VlanId	The VLAN the port that learned the entry is in. **Note**: If the MAC address is all zeros, this field shows a random VLAN ID, since the Layer 2 Switch does not yet know which port the device for this entry is attached to.
Total ARP Entries	The number of entries in the ARP cache.

Displaying IP Traffic Statistics

To display IP traffic statistics on a Layer 2 Switch, enter the following command at any CLI level:

```
FESX424 Switch# show ip traffic

IP Statistics
  27 received, 24 sent
  0 fragmented, 0 reassembled, 0 bad header
  0 no route, 0 unknown proto, 0 no buffer, 0 other errors

ICMP Statistics
Received:
  0 total, 0 errors, 0 unreachable, 0 time exceed
  0 parameter, 0 source quench, 0 redirect, 0 echo,
  0 echo reply, 0 timestamp, 0 timestamp rely, 0 addr mask
  0 addr mask reply, 0 irdp advertisement, 0 irdp solicitation
Sent:
  0 total, 0 errors, 0 unreachable, 0 time exceed
  0 parameter, 0 source quench, 0 redirect, 0 echo,
  0 echo reply, 0 timestamp, 0 timestamp rely, 0 addr mask
  0 addr mask reply, 0 irdp advertisement, 0 irdp solicitation

UDP Statistics
  0 received, 0 sent, 0 no port, 0 input errors

TCP Statistics
  1 current active tcbs, 4 tcbs allocated, 0 tcbs freed 0 tcbs protected
  0 active opens, 0 passive opens, 0 failed attempts
  0 active resets, 0 passive resets, 0 input errors
 27 in segments, 24 out segments, 0 retransmission
```

Syntax: show ip traffic

The **show ip traffic** command displays the following information.

Table 16.17: CLI Display of IP Traffic Statistics – Layer 2 Switch

This Field...	Displays...
IP statistics	
received	The total number of IP packets received by the device.

Table 16.17: CLI Display of IP Traffic Statistics – Layer 2 Switch (Continued)

This Field...	Displays...
sent	The total number of IP packets originated and sent by the device.
fragmented	The total number of IP packets fragmented by this device to accommodate the MTU of this device or of another device.
reassembled	The total number of fragmented IP packets that this device re-assembled.
bad header	The number of IP packets dropped by the device due to a bad packet header.
no route	The number of packets dropped by the device because there was no route.
unknown proto	The number of packets dropped by the device because the value in the Protocol field of the packet header is unrecognized by this device.
no buffer	This information is used by Foundry customer support.
other errors	The number of packets that this device dropped due to error types other than the types listed above.
ICMP statistics The ICMP statistics are derived from RFC 792, “Internet Control Message Protocol”, RFC 950, “Internet Standard Subnetting Procedure”, and RFC 1256, “ICMP Router Discovery Messages”. Statistics are organized into Sent and Received. The field descriptions below apply to each.	
total	The total number of ICMP messages sent or received by the device.
errors	This information is used by Foundry customer support.
unreachable	The number of Destination Unreachable messages sent or received by the device.
time exceed	The number of Time Exceeded messages sent or received by the device.
parameter	The number of Parameter Problem messages sent or received by the device.
source quench	The number of Source Quench messages sent or received by the device.
redirect	The number of Redirect messages sent or received by the device.
echo	The number of Echo messages sent or received by the device.
echo reply	The number of Echo Reply messages sent or received by the device.
timestamp	The number of Timestamp messages sent or received by the device.
timestamp reply	The number of Timestamp Reply messages sent or received by the device.
addr mask	The number of Address Mask Request messages sent or received by the device.
addr mask reply	The number of Address Mask Replies messages sent or received by the device.

Table 16.17: CLI Display of IP Traffic Statistics – Layer 2 Switch (Continued)

This Field...	Displays...
irdp advertisement	The number of ICMP Router Discovery Protocol (IRDP) Advertisement messages sent or received by the device.
irdp solicitation	The number of IRDP Solicitation messages sent or received by the device.
UDP statistics	
received	The number of UDP packets received by the device.
sent	The number of UDP packets sent by the device.
no port	The number of UDP packets dropped because the packet did not contain a valid UDP port number.
input errors	This information is used by Foundry customer support.
TCP statistics The TCP statistics are derived from RFC 793, “Transmission Control Protocol”.	
current active tcbs	The number of TCP Control Blocks (TCBs) that are currently active.
tcbs allocated	The number of TCBs that have been allocated.
tcbs freed	The number of TCBs that have been freed.
tcbs protected	This information is used by Foundry customer support.
active opens	The number of TCP connections opened by this device by sending a TCP SYN to another device.
passive opens	The number of TCP connections opened by this device in response to connection requests (TCP SYNs) received from other devices.
failed attempts	This information is used by Foundry customer support.
active resets	The number of TCP connections this device reset by sending a TCP RESET message to the device at the other end of the connection.
passive resets	The number of TCP connections this device reset because the device at the other end of the connection sent a TCP RESET message.
input errors	This information is used by Foundry customer support.
in segments	The number of TCP segments received by the device.
out segments	The number of TCP segments sent by the device.
retransmission	The number of segments that this device retransmitted because the retransmission timer for the segment had expired before the device at the other end of the connection had acknowledged receipt of the segment.

Chapter 17
Configuring RIP

This chapter describes how to configure RIP on a Foundry Layer 3 Switch.

This chapter contains the topics listed in Table 17.1.

Table 17.1: Chapter Contents

RIP Overview

Routing Information Protocol (RIP) is an IP route exchange protocol that uses a ***distance vector*** (a number representing distance) to measure the cost of a given route. The ***cost*** is a distance vector because the cost often is equivalent to the number of router hops between the Foundry Layer 3 Switch and the destination network.

A Foundry Layer 3 Switch can receive multiple paths to a destination. The software evaluates the paths, selects the best path, and saves the path in the IP route table as the route to the destination. Typically, the best path is the path with the fewest hops. A hop is another router through which packets must travel to reach the destination. If the Foundry Layer 3 Switch receives a RIP update from another router that contains a path with fewer hops than the path stored in the Foundry Layer 3 Switch's route table, the Layer 3 Switch replaces the older route with the newer one. The Layer 3 Switch then includes the new path in the updates it sends to other RIP routers, including Foundry Layer 3 Switches.

RIP routers, including the Foundry Layer 3 Switch, also can modify a route's cost, generally by adding to it, to bias the selection of a route for a given destination. In this case, the actual number of router hops may be the same, but the route has an administratively higher cost and is thus less likely to be used than other, lower-cost routes.

A RIP route can have a maximum cost of 15. Any destination with a higher cost is considered unreachable. Although limiting to larger networks, the low maximum hop count prevents endless loops in the network.

Foundry Layer 3 Switches support the following RIP versions:

- Version 1
- V1 compatible with V2
- Version 2 (the default)

ICMP Host Unreachable Message for Undeliverable ARPs

If the router receives an ARP request packet that it is unable to deliver to the final destination because of the ARP timeout and no ARP response is received (router knows of no route to the destination address), the router sends an ICMP Host Unreachable message to the source.

RIP Parameters and Defaults

The following tables list the RIP parameters, their default values, and where to find configuration information.

RIP Global Parameters

Table 17.2 lists the global RIP parameters and their default values, and indicates where you can find configuration information.

Table 17.2: RIP Global Parameters

Parameter	Description	Default	See page...
RIP state	The global state of the protocol **Note**: You also must enable the protocol on individual interfaces. Globally enabling the protocol does not allow interfaces to send and receive RIP information. See Table 17.3 on page 17-3.	Disabled	17-4
Administrative distance	The administrative distance is a numeric value assigned to each type of route on the router. When the router is selecting from among multiple routes (sometimes of different origins) to the same destination, the router compares the administrative distances of the routes and selects the route with the lowest administrative distance. This parameter applies to routes originated by RIP. The administrative distance stays with a route when it is redistributed into other routing protocols.	120	17-5
Redistribution	RIP can redistribute routes from other routing protocols such as OSPF and BGP4 into RIP. A redistributed route is one that a router learns through another protocol, then distributes into RIP.	Disabled	17-6
Redistribution metric	RIP assigns a RIP metric (cost) to each external route redistributed from another routing protocol into RIP. An external route is a route with at least one hop (packets must travel through at least one other router to reach the destination). This parameter applies to routes that are redistributed from other protocols into RIP.	1 (one)	17-7

Table 17.2: RIP Global Parameters (Continued)

Parameter	Description	Default	See page...
Update interval	How often the router sends route updates to its RIP neighbors	30 seconds	17-7
Learning default routes	The router can learn default routes from its RIP neighbors. **Note**: You also can enable or disable this parameter on an individual interface basis. See Table 17.3 on page 17-3.	Disabled	17-8
Advertising and learning with specific neighbors	The Layer 3 Switch learns and advertises RIP routes with all its neighbors by default. You can prevent the Layer 3 Switch from advertising routes to specific neighbors or learning routes from specific neighbors.	Learning and advertising permitted for all neighbors	17-8

RIP Interface Parameters

Table 17.3 lists the interface-level RIP parameters and their default values, and indicates where you can find configuration information.

Table 17.3: RIP Interface Parameters

Parameter	Description	Default	See page...
RIP state and version	The state of the protocol and the version that is supported on the interface. The version can be one of the following: • Version 1 only • Version 2 only • Version 1, but also compatible with version 2 **Note**: You also must enable RIP globally.	Disabled	17-4
Metric	A numeric cost the router adds to RIP routes learned on the interface. This parameter applies only to RIP routes.	1 (one)	17-4
Learning default routes	Locally overrides the global setting. See Table 17.2 on page 17-2.	Disabled	17-8
Loop prevention	The method a router uses to prevent routing loops caused by advertising a route on the same interface as the one on which the router learned the route. • Split horizon – The router does not advertise a route on the same interface as the one on which the router learned the route. • Poison reverse – The router assigns a cost of 16 (“infinite” or “unreachable”) to a route before advertising it on the same interface as the one on which the router learned the route.	Poison reverse **Note**: Enabling split horizon disables poison reverse on the interface.	17-8

Table 17.3: RIP Interface Parameters (Continued)

Parameter	Description	Default	See page...
Advertising and learning specific routes	You can control the routes that a Layer 3 Switch learns or advertises.	The Layer 3 Switch learns and advertises all RIP routes on all interfaces.	17-9

Configuring RIP Parameters

Use the following procedures to configure RIP parameters on a system-wide and individual interface basis.

Enabling RIP

RIP is disabled by default. To enable it, use the following method.

NOTE: You must enable the protocol globally and also on individual interfaces on which you want to advertise RIP. Globally enabling the protocol does not enable it on individual interfaces.

To enable RIP globally, enter the following command:

```
FastIron SuperX Router(config)# router rip
```

Syntax: [no] router rip

After globally enabling the protocol, you must enable it on individual interfaces. You can enable the protocol on physical interfaces as well as virtual routing interfaces. To enable RIP on an interface, enter commands such as the following:

```
FastIron SuperX Router(config)# interface ethernet 1/1
FastIron SuperX Router(config-if-1/1)# ip rip v1-only
```

Syntax: [no] ip rip v1-only | v1-compatible-v2 | v2-only

NOTE: You must specify the RIP version.

Configuring Metric Parameters

By default, a Foundry Layer 3 Switch port increases the cost of a RIP route that is learned on the port by one. You can configure individual ports to add more than one to a learned route's cost. In addition, you can configure a RIP offset list to increase the metric for learned or advertised routes based on network address.

Changing the Cost of Routes Learned on a Port

By default, a Foundry Layer 3 Switch port increases the cost of a RIP route that is learned on the port. The Layer 3 Switch increases the cost by adding one to the route's metric before storing the route.

You can change the amount that an individual port adds to the metric of RIP routes learned on the port. To do so, use the following method.

NOTE: RIP considers a route with a metric of 16 to be unreachable. Use this metric only if you do not want the route to be used. In fact, you can prevent the Layer 3 Switch from using a specific port for routes learned though that port by setting its metric to 16.

To increase the cost a port adds to RIP routes learned in that port, enter commands such as the following:


```
FastIron SuperX Router(config)# interface ethernet 6/1
FastIron SuperX Router(config-if-6/1)# ip metric 5
```

These commands configure port 6/1 to add 5 to the cost of each route learned on the port.

Syntax: ip metric <1-16>

Configuring a RIP Offset List

A RIP offset list allows you to add to the metric of specific inbound or outbound routes learned or advertised by RIP. RIP offset lists provide a simple method for adding to the cost of specific routes and therefore biasing the Layer 3 Switch's route selection away from those routes.

An offset list consists of the following parameters:

- An ACL that specifies the routes to which to add the metric.
- The direction:
  - In applies to routes the Layer 3 Switch learns from RIP neighbors.
  - Out applies to routes the Layer 3 Switch is advertising to its RIP neighbors.
- The type and number of a specific port to which the offset list applies (optional).

The software adds the offset value to the routing metric (cost) of the routes that match the ACL. If a route matches both a global offset list and an interface-based offset list, the interface-based offset list takes precedence. The interface-based offset list's metric is added to the route in this case.

You can configure up to 24 global RIP offset lists and up to 24 RIP offset lists on each interface.

To configure a global RIP offset list, enter commands such as the following:

```
FastIron SuperX Router(config)# access-list 21 deny 160.1.0.0 0.0.255.255
FastIron SuperX Router(config)# access-list 21 permit any
FastIron SuperX Router(config)# router rip
FastIron SuperX Router(config-rip-router)# offset-list 21 out 10
```

The commands in this example configure a standard ACL. The ACL matches on all IP networks except 160.1.x.x. When the Layer 3 Switch advertises a route that matches ACL 21, the offset list adds 10 to the route's metric.

Syntax: [no] <acl-number-or-name> in | out offset [ethernet [<slotnum>/]<portnum>]

In the following example, the Layer 3 Switch uses ACL 21 to add 10 to the metric of routes received on Ethernet port 2/1.

```
FastIron SuperX Router(config-rip-router)# offset-list 21 in ethernet 2/1
```

Changing the Administrative Distance

By default, the Layer 3 Switch assigns the default RIP administrative distance (120) to RIP routes. When comparing routes based on administrative distance, the Layer 3 Switch selects the route with the lower distance. You can change the administrative distance for RIP routes.

NOTE: See "Changing Administrative Distances" on page 21-29 for a list of the default distances for all route sources.

To change the administrative distance for RIP routes, enter a command such as the following:

```
FastIron SuperX Router(config-rip-router)# distance 140
```

This command changes the administrative distance to 140 for all RIP routes.

Syntax: [no] distance <num>

Configuring Redistribution

You can configure the Layer 3 Switch to redistribute routes learned through Open Shortest Path First (OSPF) or Border Gateway Protocol version 4 (BGP4) into RIP. When you redistribute a route from one of these other protocols into RIP, the Layer 3 Switch can use RIP to advertise the route to its RIP neighbors.

To configure redistribution, perform the following tasks:

- Configure redistribution filters (optional). You can configure filters to permit or deny redistribution for a route based on its origin (OSPF, BGP4, and so on), the destination network address, and the route's metric. You also can configure a filter to set the metric based on these criteria.
- Change the default redistribution metric (optional). The Layer 3 Switch assigns a RIP metric of one to each redistributed route by default. You can change the default metric to a value up to 16.
- Enable redistribution.

NOTE: Do not enable redistribution until you configure the other redistribution parameters.

Configuring Redistribution Filters

RIP redistribution filters apply to all interfaces. The software uses the filters in ascending numerical order and immediately takes the action specified by the filter. Thus, if filter 1 denies redistribution of a given route, the software does not redistribute the route, regardless of whether a filter with a higher ID permits redistribution of that route.

NOTE: The default redistribution action is permit, even after you configure and apply redistribution filters to the virtual routing interface. If you want to tightly control redistribution, apply a filter to deny all routes as the last filter (the filter with the highest ID), then apply filters with lower filter IDs to allow specific routes.

To configure a redistribution filter, enter a command such as the following:

```
FESX424 Router(config-rip-router)# deny redistribute 2 all address 207.92.0.0
255.255.0.0
```

This command denies redistribution for all types of routes to the 207.92.x.x network.

Syntax: [no] permit | deny redistribute <filter-num> all | bgp | ospf | static address <ip-addr> <ip-mask> [match-metric <value> | set-metric <value>]

The <filter-num> specifies the redistribution filter ID. The software uses the filters in ascending numerical order. Thus, if filter 1 denies a route from being redistributed, the software does not redistribute that route even if a filter with a higher ID permits redistribution of the route.

The **all** parameter applies redistribution to all route types.

The **bgp** parameter applies redistribution to BGP4 routes only.

The **ospf** parameter applies redistribution to OSPF routes only.

The **static** parameter applies redistribution to IP static routes only.

The **address** <ip-addr> <ip-mask> parameters apply redistribution to the specified network and sub-net address. Use 0 to specify "any". For example, "207.92.0.0 255.255.0.0" means "any 207.92.x.x sub-net". However, to specify any sub-net (all sub-nets match the filter), enter "address 255.255.255.255 255.255.255.255".

The **match-metric** <value> parameter applies the redistribution filter only to those routes with the specified metric value; possible values are from 1 – 15.

The **set-metric** <value> parameter sets the RIP metric value that will be applied to those routes imported into RIP.

The following command denies redistribution into RIP for all OSPF routes:


```
FESX424 Router(config-rip-router)# deny redistribute 3 ospf address 207.92.0.0
255.255.0.0
```

The following command denies redistribution for all OSPF routes that have a metric of 10:

```
FESX424 Router(config-rip-router)# deny redistribute 3 ospf address 207.92.0.0
255.255.0.0 match-metric 10
```

The following commands deny redistribution of all routes except routes for 10.10.10.x and 20.20.20.x:

```
FESX424 Router(config-rip-router)# deny redistribute 64 static address
255.255.255.255 255.255.255.255
FESX424 Router(config-rip-router)# permit redistribute 1 static address 10.10.10.0
255.255.255.0
FESX424 Router(config-rip-router)# permit redistribute 2 static address 20.20.20.0
255.255.255.0
```

NOTE: This example assumes that the highest RIP redistribution filter ID configured on the device is 64.

Changing the Redistribution Metric

When the Layer 3 Switch redistributes a route into RIP, the software assigns a RIP metric (cost) to the route. By default, the software assigns a metric of one to each route that is redistributed into RIP. You can increase the metric that the Layer 3 Switch assigns, up to 15.

To change the RIP metric the Layer 3 Switch assigns to redistributed routes, enter a command such as the following:

```
FESX424 Router(config-rip-router)# default-metric 10
```

This command assigns a RIP metric of 10 to each route that is redistributed into RIP.

Syntax: [no] default-metric <1-15>

Enabling Redistribution

After you configure redistribution parameters, you need to enable redistribution.

To enable RIP redistribution, enter the following command:

```
FESX424 Router(config-rip-router)# redistribution
```

Syntax: [no] redistribution

Configuring Route Learning and Advertising Parameters

By default, a Foundry Layer 3 Switch learns routes from all its RIP neighbors and advertises RIP routes to those neighbors.

You can configure the following learning and advertising parameters:

- Update interval – The update interval specifies how often the Layer 3 Switch sends RIP route advertisements to its neighbors. The default is 30 seconds. You can change the interval to a value from 1 – 1000 seconds.
- Learning and advertising of RIP default routes – The Layer 3 Switch learns and advertises RIP default routes by default. You can disable learning and advertising of default routes on a global or individual interface basis.
- Learning of standard RIP routes – By default, the Layer 3 Switch can learn RIP routes from all its RIP neighbors. You can configure RIP neighbor filters to explicitly permit or deny learning from specific neighbors.

Changing the Update Interval for Route Advertisements

The update interval specifies how often the Layer 3 Switch sends route advertisements to its RIP neighbors. You can specify an interval from 1 – 1000 seconds. The default is 30 seconds.

To change the RIP update interval, enter a command such as the following:

```
FESX424 Router(config-rip-router)# update 120
```

This command configures the Layer 3 Switch to send RIP updates every 120 seconds.

Syntax: update-time <1-1000>

Enabling Learning of RIP Default Routes

By default, the Layer 3 Switch does not learn RIP default routes. You can enable learning of RIP default routes on a global or interface basis.

To enable learning of default RIP routes on a global basis, enter the following command:

```
FESX424 Router(config-rip-router)# learn-default
```

Syntax: [no] learn-default

To enable learning of default RIP routes on an interface basis, enter commands such as the following:

```
FESX424 Router(config)# interface ethernet 1
FESX424 Router(config-if-1)# ip rip learn-default
```

Syntax: [no] ip rip learn-default

Configuring a RIP Neighbor Filter

By default, a Foundry Layer 3 Switch learns RIP routes from all its RIP neighbors. Neighbor filters allow you to specify the neighbor routers from which the Foundry device can receive RIP routes. Neighbor filters apply globally to all ports.

To configure a RIP neighbor filters, enter a command such as the following:

```
FESX424 Router(config-rip-router)# neighbor 1 deny any
```

Syntax: [no] neighbor <filter-num> permit | deny <source-ip-address> | any

This command configures the Layer 3 Switch so that the device does not learn any RIP routes from any RIP neighbors.

The following commands configure the Layer 3 Switch to learn routes from all neighbors except 192.168.1.170. Once you define a RIP neighbor filter, the default action changes from learning all routes from all neighbors to denying all routes from all neighbors except the ones you explicitly permit. Thus, to deny learning from a specific neighbor but allow all other neighbors, you must add a filter that allows learning from all neighbors. Make sure you add the filter to permit all neighbors as the last filter (the one with the highest filter number). Otherwise, the software can match on the permit all filter before a filter that denies a specific neighbor, and learn routes from that neighbor.

```
FESX424 Router(config-rip-router)# neighbor 2 deny 192.16.1.170
FESX424 Router(config-rip-router)# neighbor 1024 permit any
```

Changing the Route Loop Prevention Method

RIP can use the following methods to prevent routing loops:

- Split horizon – The Layer 3 Switch does not advertise a route on the same interface as the one on which the router learned the route.
- Poison reverse – The Layer 3 Switch assigns a cost of 16 (“infinite” or “unreachable”) to a route before advertising it on the same interface as the one on which the router learned the route. This is the default.

These loop prevention methods are configurable on an individual interface basis. One of the methods is always in effect on an interface enabled for RIP. Thus, if you disable one method, the other method is enabled.

NOTE: These methods are in addition to RIP’s maximum valid route cost of 15.

To disable poison reverse and enable split horizon on an interface, enter commands such as the following:

```
FastIron SuperX Router(config)# interface ethernet 1/1
```



```
FastIron SuperX Router(config-if-1/1)# no ip rip poison-reverse
```

Syntax: [no] ip rip poison-reverse

To disable split horizon and enable poison reverse on an interface, enter commands such as the following:

```
FastIron SuperX Router(config)# interface ethernet 1/1
FastIron SuperX Router(config-if-1/1)# ip rip poison-reverse
```

Suppressing RIP Route Advertisement on a VRRP or VRRPE Backup Interface

NOTE: This section applies only if you configure the Layer 3 Switch for Virtual Router Redundancy Protocol (VRRP) or VRRP Extended (VRRPE). See "Configuring VRRP and VRRPE" on page 22-1.

Normally, a VRRP or VRRPE Backup includes route information for the virtual IP address (the backed up interface) in RIP advertisements. As a result, other routers receive multiple paths for the backed up interface and might sometimes unsuccessfully use the path to the Backup rather than the path to the Master.

You can prevent the Backups from advertising route information for the backed up interface by enabling suppression of the advertisements.

To suppress RIP advertisements for the backed up interface, enter the following commands:

```
FastIron SuperX Router(config)# router rip
FastIron SuperX Router(config-rip-router)# use-vrrp-path
```

Syntax: [no] use-vrrp-path

The syntax is the same for VRRP and VRRPE.

Configuring RIP Route Filters

You can configure RIP route filters to permit or deny learning or advertising of specific routes. Configure the filters globally, then apply them to individual interfaces. When you apply a RIP route filter to an interface, you specify whether the filter applies to learned routes (in) or advertised routes (out).

NOTE: A route is defined by the destination's IP address and network mask.

NOTE: By default, routes that do not match a route filter are learned or advertised. To prevent a route from being learned or advertised, you must configure a filter to deny the route.

To configure RIP filters, enter commands such as the following:

```
FastIron SuperX Router(config-rip-router)# filter 1 permit 192.53.4.1 255.255.255.0
FastIron SuperX Router(config-rip-router)# filter 2 permit 192.53.5.1 255.255.255.0
FastIron SuperX Router(config-rip-router)# filter 3 permit 192.53.6.1 255.255.255.0
FastIron SuperX Router(config-rip-router)# filter 4 deny 192.53.7.1 255.255.255.0
```

These commands explicitly permit RIP routes to three networks, and deny the route to one network.

Since the default action is permit, all other routes (routes not explicitly permitted or denied by the filters) can be learned or advertised.

Syntax: filter <filter-num> permit | deny <source-ip-address> | any <source-mask> | any [log]

Applying a RIP Route Filter to an Interface

Once you define RIP route filters, you must assign them to individual interfaces. The filters do not take effect until you apply them to interfaces. When you apply a RIP route filter, you also specify whether the filter applies to learned routes or advertised routes:

- Out filters apply to routes the Layer 3 Switch advertises to its neighbor on the interface.

- In filters apply to routes the Layer 3 Switch learns from its neighbor on the interface.

To apply RIP route filters to an interface, enter commands such as the following:

```
FastIron SuperX Router(config)# interface ethernet 1/2
FastIron SuperX Router(config-if-1/2)# ip rip filter-group in 2 3 4
```

Syntax: [no] ip rip filter-group in | out <filter-list>

These commands apply RIP route filters 2, 3, and 4 to all routes learned from the RIP neighbor on port 1/2.

Displaying RIP Filters

To display the RIP filters configured on the router, enter the following command at any CLI level:

```
FESX424 Router> show ip rip

               RIP Route Filter Table
  Index   Action   Route IP Address    Subnet Mask
  1       deny     any                 any

               RIP Neighbor Filter Table
  Index   Action   Neighbor IP Address
  1       permit   any
```

Syntax: show ip rip

This display shows the following information.

Table 17.4: CLI Display of RIP Filter Information

This Field...	Displays...
Route filters The rows underneath “RIP Route Filter Table” list the RIP route filters. If no RIP route filters are configured on the device, the following message is displayed instead: “No Filters are configured in RIP Route Filter Table”.	
Index	The filter number. You assign this number when you configure the filter.
Action	The action the router takes if a RIP route packet matches the IP address and sub-net mask of the filter. The action can be one of the following: • deny – RIP route packets that match the address and network mask information in the filter are dropped. If applied to an interface’s outbound filter group, the filter prevents the router from advertising the route on that interface. If applied to an interface’s inbound filter group, the filter prevents the router from adding the route to its IP route table. • permit – RIP route packets that match the address and network mask information are accepted. If applied to an interface’s outbound filter group, the filter allows the router to advertise the route on that interface. If applied to an interface’s inbound filter group, the filter allows the router to add the route to its IP route table.
Route IP Address	The IP address of the route’s destination network or host.

Table 17.4: CLI Display of RIP Filter Information (Continued)

This Field...	Displays...
Subnet Mask	The network mask for the IP address.
Neighbor filters The rows underneath "RIP Neighbor Filter Table" list the RIP neighbor filters. If no RIP neighbor filters are configured on the device, the following message is displayed instead: "No Filters are configured in RIP Neighbor Filter Table".	
Index	The filter number. You assign this number when you configure the filter.
Action	The action the router takes for RIP route packets to or from the specified neighbor: • deny – If the filter is applied to an interface's outbound filter group, the filter prevents the router from advertising RIP routes to the specified neighbor on that interface. If the filter is applied to an interface's inbound filter group, the filter prevents the router from receiving RIP updates from the specified neighbor. • permit – If the filter is applied to an interface's outbound filter group, the filter allows the router to advertise RIP routes to the specified neighbor on that interface. If the filter is applied to an interface's inbound filter group, the filter allows the router to receive RIP updates from the specified neighbor.
Neighbor IP Address	The IP address of the RIP neighbor.

Displaying CPU Utilization Statistics

You can display CPU utilization statistics for RIP and other IP protocols.

To display CPU utilization statistics for RIP for the previous one-second, one-minute, five-minute, and fifteen-minute intervals, enter the following command at any level of the CLI:

```
FESX424 Router# show process cpu
Process Name    5Sec(%)    1Min(%)    5Min(%)    15Min(%)    Runtime(ms)
ARP               0.01       0.03       0.09       0.22              9
BGP               0.04       0.06       0.08       0.14             13
GVRP              0.00       0.00       0.00       0.00              0
ICMP              0.00       0.00       0.00       0.00              0
IP                0.00       0.00       0.00       0.00              0
OSPF              0.00       0.00       0.00       0.00              0
RIP               0.04       0.07       0.08       0.09              7
STP               0.00       0.00       0.00       0.00              0
VRRP              0.00       0.00       0.00       0.00              0
```

If the software has been running less than 15 minutes (the maximum interval for utilization statistics), the command indicates how long the software has been running. Here is an example:

```
FESX424 Router# show process cpu
The system has only been up for 6 seconds.
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.01      0.00      0.00       0.00             0
BGP              0.00      0.00      0.00       0.00             0
GVRP             0.00      0.00      0.00       0.00             0
ICMP             0.01      0.00      0.00       0.00             1
IP               0.00      0.00      0.00       0.00             0
OSPF             0.00      0.00      0.00       0.00             0
RIP              0.00      0.00      0.00       0.00             0
STP              0.00      0.00      0.00       0.00             0
VRRP             0.00      0.00      0.00       0.00             0
```

To display utilization statistics for a specific number of seconds, enter a command such as the following:

```
FESX424 Router# show process cpu 2
Statistics for last 1 sec and 80 ms
Process Name   Sec(%)   Time(ms)
ARP              0.00          0
BGP              0.00          0
GVRP             0.00          0
ICMP             0.01          1
IP               0.00          0
OSPF             0.00          0
RIP              0.00          0
STP              0.01          0
VRRP             0.00          0
```

When you specify how many seconds' worth of statistics you want to display, the software selects the sample that most closely matches the number of seconds you specified. In this example, statistics are requested for the previous two seconds. The closest sample available is actually for the previous 1 second plus 80 milliseconds.

Syntax: show process cpu [<num>]

The <num> parameter specifies the number of seconds and can be from 1 – 900. If you use this parameter, the command lists the usage statistics only for the specified number of seconds. If you do not use this parameter, the command lists the usage statistics for the previous one-second, one-minute, five-minute, and fifteen-minute intervals.

Chapter 18
Configuring IP Multicast Traffic Reduction

This chapter describes how to configure IP multicast traffic reduction and PIM SM Traffic Snooping parameters on a Foundry device.

This chapter contains the following information:

Table 18.1: Chapter Contents

Overview

Foundry Layer 2 Switches and Layer 3 Switches forward all IP multicast traffic by default based on the Layer 2 information in the packets. Optionally, you can enable these Foundry devices to make forwarding decisions in hardware, based on multicast group by enabling the IP Multicast Traffic Reduction feature.

When this feature is enabled, these Foundry devices examine the MAC address in an IP multicast packet and forward the packet only on the ports from which the device has received Group Membership reports for that group, instead of forwarding all multicast traffic to all ports. The device sends traffic for other groups out all ports.

When you enable IP Multicast Traffic Reduction, you also can configure the following features:

- IGMP mode – When you enable IP Multicast Traffic Reduction, the device passively listens for IGMP Group Membership reports by default. If the multicast domain does not have a router to send IGMP queries to elicit these Group Membership reports, you can enable the device to actively send the IGMP queries.
- Query interval – The query interval specifies how often the device sends Group Membership queries. This query interval applies only to the active IGMP mode. The default is 60 seconds. You can change the interval to a value from 10 – 600 seconds.
- Age interval – The age interval specifies how long an IGMP group can remain in the IGMP group table without the device receiving a Group Membership report for the group. If the age interval expires before the device

receives another Group Membership report for the group, the device removes the entry from the table. The default is 140 seconds. You can change the interval to a value from 10 – 1220 seconds.

- Forwarding policy – The device forwards all IP multicast traffic by default but you can enable the device to forward IP multicast traffic only for groups for which the device has received a Group Membership report, and drop traffic for all other groups.

Support for IGMP V2 Snooping

IGMP V2 snooping has been supported on FESX devices running *router* software since release 02.0.00. Release 02.2.00 supports IGMP V2 snooping on both *router* and *switch* software images. On both images, IGMP V2 snooping is MAC-based. This differs from IGMP V2 snooping on the BigIron/FastIron router images, which match on both IP source and group (S,G) entries programmed in the Layer 4 CAM. In contrast, the FESX router images match on Layer 2 destination MAC address entries.

When Layer 2 CAM is used, traffic is switched solely based on the destination MAC address. Consequently, traffic of the same group coming to the same port, regardless of its source, is switched in the same way. In addition, the lowest 23 bits of the group address are mapped to a MAC address. In this way, multiple groups (for example, 224.1.1.1 and 225.1.1.1) have the same MAC address. Groups having the same MAC address are switched to the same destination ports, which are the superset of individual group output ports. Thus, the use of Layer 2 CAM might cause unwanted packets to be sent to some ports. However, the switch generally needs far less Layer 2 CAM than it does Layer 4 CAM, which is required for each stream with a different source and group.

NOTE: Layer 2 IGMP snooping is automatically enabled with Layer 3 multicast routing. If Layer 3 multicast routing is enabled on your system, do not attempt to enable Layer 2 IGMP snooping.

Configuring IP Multicast Traffic Reduction

By default, Foundry devices forward all IP multicast traffic out all ports except the port on which the traffic was received. To reduce multicast traffic through the device, you can enable IP Multicast Traffic Reduction. This feature configures the device to forward multicast traffic only on the ports attached to multicast group members, instead of forwarding all multicast traffic to all ports. The device determines the ports that are attached to multicast group members based on entries in the IGMP table. Each entry in the table consists of MAC addresses and the Foundry device ports from which the device has received Group Membership reports for that group.

By default, the device broadcasts traffic addressed to an IP multicast group that doesn't have any entries in the IGMP table. When you enable IP Multicast Traffic Reduction, the device determines the ports that are attached to multicast group members based on entries in the IGMP table. The IGMP table entries are created when the VLAN receives a group membership report for a group. Each entry in the table consists of an IP multicast group address and the Foundry device ports from which the device has received Group Membership reports.

When the device receives traffic for an IP multicast group, the device looks in the IGMP table for an entry corresponding to that group. If the device finds an entry, the device forwards the group traffic out the ports listed in the corresponding entries, as long as the ports are members of the same VLAN. If the table does not contain an entry corresponding to the group or if the port is a member of the default VLAN, the device broadcasts the traffic.

NOTE: When one or more Foundry devices are running Layer 2 IP Multicast Traffic reduction, configure one of the devices for active IGMP and leave the other devices configured for passive IGMP. However, if the IP multicast domain contains a multicast-capable router, configure all the Foundry devices for passive IGMP and allow the router to actively send the IGMP queries.

Enabling IP Multicast Traffic Reduction

To enable IP Multicast Traffic Reduction, enter the following command:

```
FESX424 Router(config)# ip multicast
```


Syntax: [no] ip multicast

NOTE: If the "route-only" feature is enabled on the Layer 3 Switch, then IP Multicast Traffic Reduction will not be supported.

NOTE: This feature is not supported on the default VLAN of Layer 3 Switches.

To verify that IP Multicast Traffic Reduction is enabled, enter the following command at any level of the CLI:

```
FESX424 Router(config)# show ip multicast
IP multicast is enabled - Active
```

Syntax: show ip multicast

Changing the IGMP Mode

When you enable IP Multicast Traffic Reduction on the device, IGMP also is enabled. The device uses IGMP to maintain a table of the Group Membership reports received by the device. You can use active or passive IGMP mode. The default mode is passive.

- Active – When active IGMP mode is enabled, a Foundry device actively sends out IGMP queries to identify IP multicast groups on the network and makes entries in the IGMP table based on the Group Membership reports received from the network.

  NOTE: Routers in the network generally handle this operation. Use the active IGMP mode only when the device is in a stand-alone Layer 2 Switched network with no external IP multicast router attachments. In this case, enable the active IGMP mode on only one of the devices and leave the other devices configured for passive IGMP mode.

- Passive – When passive IGMP mode is enabled, the device listens for IGMP Group Membership reports but does not send IGMP queries. The passive mode is sometimes called “IGMP snooping”. Use this mode when another device in the network is actively sending queries.

To enable active IGMP, enter the following command:

```
FESX424 Router(config)# ip multicast active
FESX424 Router(config)# write memory
FESX424 Router(config)# end
FESX424 Router# reload
```

Syntax: [no] ip multicast active | passive

To enable passive IGMP, enter the following command:

```
FESX424 Router(config)# ip multicast passive
FESX424 Router(config)# write memory
FESX424 Router(config)# end
FESX424 Router# reload
```

Disabling IGMP on Individual Ports

NOTE: IP Multicast Traffic Reduction cannot be disabled on individual ports of a Layer 3 Switch. You cannot use the **ip-multicast-disable** command that is available on Layer 2 Switches. IP multicast must be enabled and disabled globally on Layer 3 Switches.

By default, when you enable IP multicast on a Foundry device, all ports on the device are configured for IGMP. If you are using active IGMP, all ports can send IGMP queries and receive IGMP reports. If you are using passive IGMP, all ports can receive IGMP queries.

You can disable IGMP on individual ports of a Layer 2 Switch if you want to block all IP multicast traffic on those ports. When you disable IGMP on an individual port, the device does not forward any multicast traffic out the port, but other ports can still send and receive multicast traffic.

To disable IGMP on a port, use the following CLI method.

```
FastIron SuperX Switch(config)# int e 1/5
FastIron SuperX Switch(config-if-1/5)# ip-multicast-disable
```

Syntax: [no] ip-multicast-disable

The command in this example disables IGMP on port 1/5 but does not affect the state of IGMP on other ports.

Modifying the Query Interval

If IP Multicast Traffic Reduction is set to active mode, you can modify the query interval, which specifies how often a Foundry device enabled for active IP Multicast Traffic Reduction sends Group Membership queries.

NOTE: The query interval applies only to the active mode of IP Multicast Traffic reduction.

To modify the query interval, enter a command such as the following:

```
FESX424 Router(config)# ip multicast query-interval 120
```

Syntax: [no] ip multicast query-interval <interval>

The <interval> parameter specifies the interval between queries. You can specify a value from 10 – 600 seconds. The default is 60 seconds.

Modifying the Age Interval

When the device receives a Group Membership report, the device makes an entry in the IGMP group table for the group in the report. The age interval specifies how long the entry can remain in the table without the device receiving another Group Membership report.

To modify the age interval, enter a command such as the following:

```
FESX424 Router(config)# ip multicast age-interval 280
```

Syntax: [no] ip multicast age-interval <interval>

The <interval> parameter specifies the interval between queries. You can specify a value from 10 – 1220 seconds. The default is 140 seconds.

Filtering Multicast Groups

By default, Foundry devices forward multicast traffic for all valid multicast groups. You can configure a Foundry device to filter out all multicast traffic for groups other than the ones for which the device has received Group Membership reports.

When the device starts up, it forwards all multicast groups even though multicast traffic filters are configured. This process continues until the device receives a group membership report. Once the group membership report is received, the device drops all multicast packets for groups other than the ones for which the device has received the group membership report.

To enable IP multicast filtering, enter the following command:

```
FESX424 Router(config)# ip multicast filter
```

Syntax: [no] ip multicast filter

PIM SM Traffic Snooping

By default, when a Foundry device receives an IP multicast packet, the device does not examine the multicast information in the packet. Instead, the device simply forwards the packet out all ports except the port that received the packet. In some networks, this method can cause unnecessary traffic overhead in the network. For example, if the Foundry device is attached to only one group source and two group receivers, but has devices attached to every port, the device forwards group traffic out all ports in the same broadcast domain except the port attached to the source, even though there are only two receivers for the group.

PIM SM traffic snooping eliminates the superfluous traffic by configuring the device to forward IP multicast group traffic only on the ports that are attached to receivers for the group.

PIM SM traffic snooping requires IP multicast traffic reduction to be enabled on the device. IP multicast traffic reduction configures the device to listen for IGMP messages. PIM SM traffic snooping provides a finer level of multicast traffic control by configuring the device to listen specifically for PIM SM join and prune messages sent from one PIM SM router to another through the device.

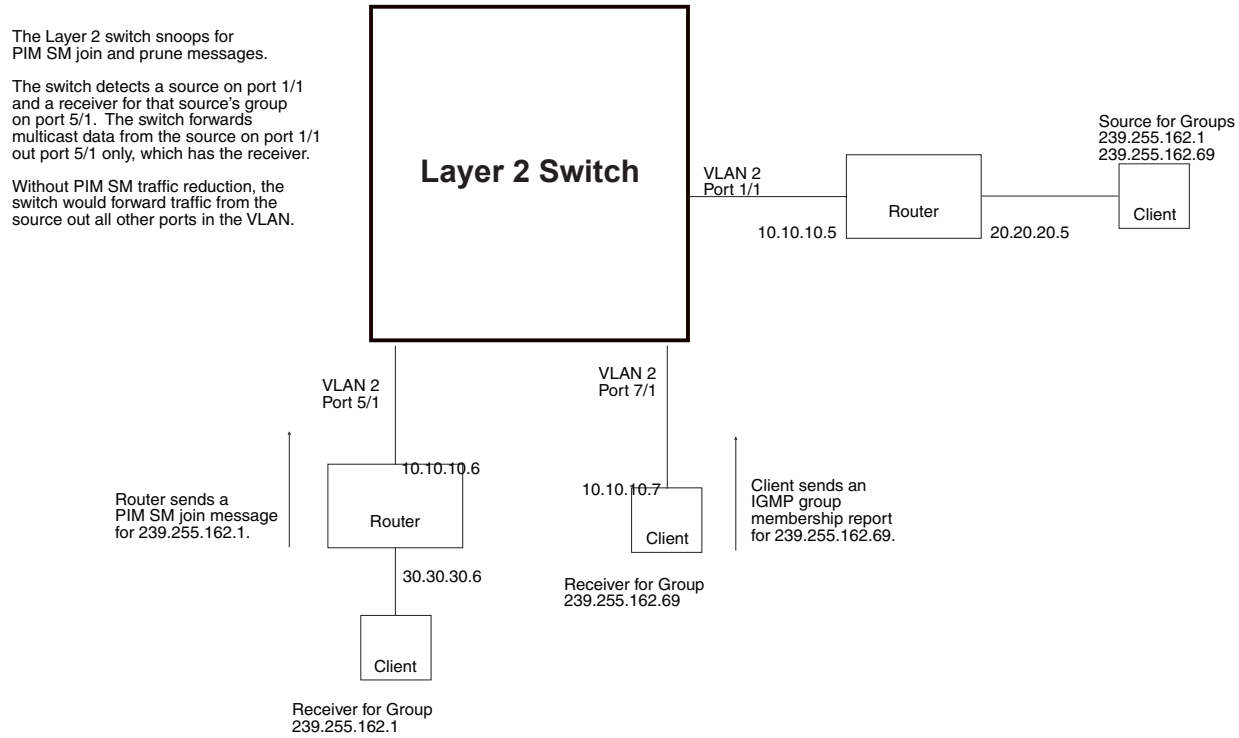
Configuration Notes

- This feature applies only to PIM SM version 2 (PIM V2).
- This feature is supported in the Layer 3 code and in the Layer 2 switch code starting in release 02.2.00.
- This feature is supported in software release 02.2.00 and later for the FESX and FWSX.
- This feature is supported in software release 02.3.01 and later for the FSX.

Application Examples

Figure 18.1 shows an example application of the PIM SM traffic snooping feature. In this example, a device is connected through an IP router to a PIM SM group source that is sending traffic for two PIM SM groups. The device also is connected to a receiver for each of the groups.

Figure 18.1 PIM SM traffic reduction in enterprise network



When PIM SM traffic snooping is enabled, the device starts listening for PIM SM join and prune messages and IGMP group membership reports. Until the device receives a PIM SM join message or an IGMP group membership report, the device forwards IP multicast traffic out all ports. Once the device receives a join message or group membership report for a group, the device forwards subsequent traffic for that group only on the ports from which the join messages or IGMP reports were received.

In this example, the router connected to the receiver for group 239.255.162.1 sends a join message toward the group's source. Since PIM SM traffic snooping is enabled on the device, the device examines the join message to learn the group ID, then makes a forwarding entry for the group ID and the port connected to the receiver's router. The next time the device receives traffic for 239.255.162.1 from the group's source, the device forwards the traffic only on port 5/1, since that is the only port connected to a receiver for the group.

Notice that the receiver for group 239.255.162.69 is directly connected to the device. As result, the device does not see a join message on behalf of the client. However, since IP multicast traffic reduction also is enabled, the device uses the IGMP group membership report from the client to select the port for forwarding traffic to group 239.255.162.69 receivers.

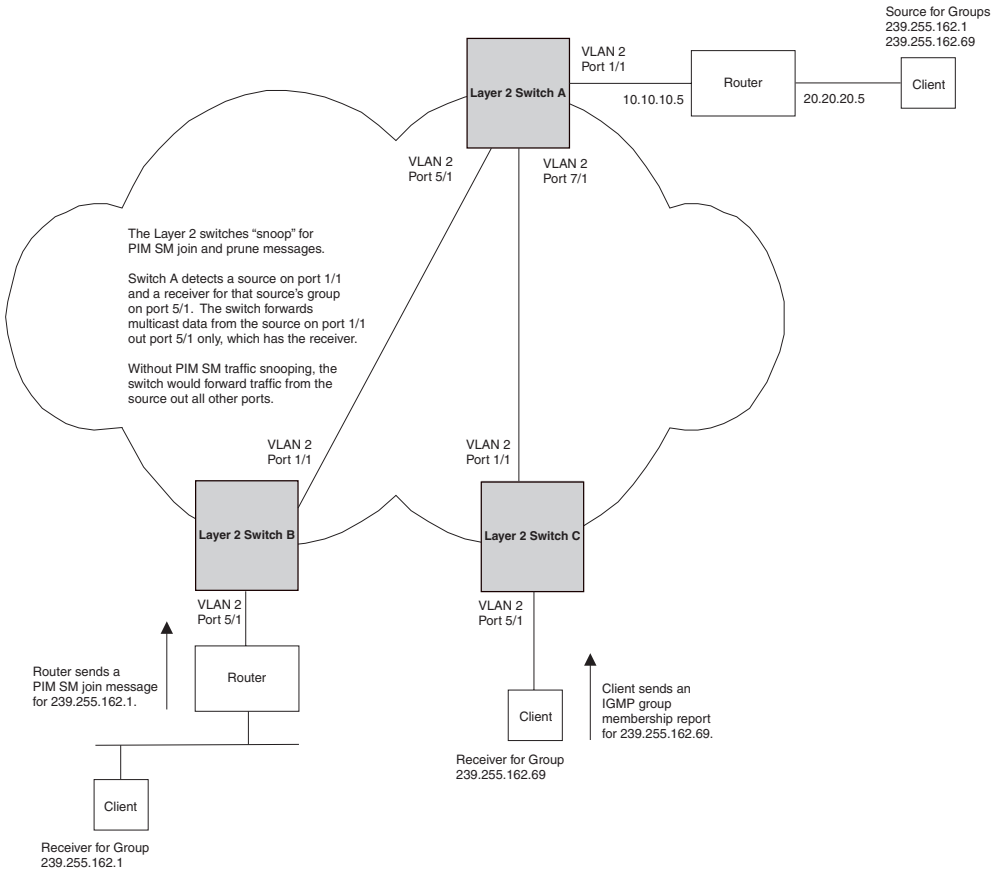
The IP multicast traffic reduction feature and the PIM SM traffic snooping feature together build a list of groups and forwarding ports for the VLAN. The list includes PIM SM groups learned through join messages as well as MAC addresses learned through IGMP group membership reports. In this case, even though the device never sees a join message for the receiver for group 239.255.162.69, the device nonetheless learns about the receiver and forwards group traffic to the receiver.

The device stops forwarding IP multicast traffic on a port for a group if the port receives a prune message for the group.

Notice that the ports connected to the source and the receivers are all in the same port-based VLAN on the device. This is required for the PIM SM snooping feature. The feature also requires the source and the downstream router to be on different IP sub-nets, as shown in Figure 18.1.

Figure NOTE: shows another example application for PIM SM traffic snooping. This example shows devices on the edge of a Global Ethernet cloud (a Layer 2 Packet over SONET cloud). Assume that each device is attached to numerous other devices such as other Layer 2 Switches and Layer 3 Switches (routers).

NOTE: This example assumes that the devices are actually FastIron devices running Layer 2 Switch software.
PIM SM traffic reduction in Global Ethernet environment



The devices on the edge of the Global Ethernet cloud are configured for IP multicast traffic reduction and PIM SM traffic snooping. Although this application uses multiple devices, the feature has the same requirements and works the same way as it does on a single device.

Configuration Requirements

- IP multicast traffic reduction must be enabled on the device that will be running PIM SM snooping. The PIM SM traffic snooping feature requires IP multicast traffic reduction.

  NOTE: Use the passive mode of IP multicast traffic reduction instead of the active mode. The passive mode assumes that a router is sending group membership queries as well as join and prune messages on behalf of receivers. The active mode configures the device to send group membership queries.

- All the device ports connected to the source and receivers or routers must be in the same port-based VLAN.

- The PIM SM snooping feature assumes that the group source and the device are in different sub-nets and communicate through a router. The source must be in a different IP sub-net than the receivers. A PIM SM router sends PIM join and prune messages on behalf of a multicast group receiver only when the router and the source are in different sub-nets. When the receiver and source are in the same sub-net, they do not need the router in order to find one another. They find one another directly within the sub-net.

  The device forwards all IP multicast traffic by default. Once you enable IP multicast traffic reduction and PIM SM traffic snooping, the device initially blocks all PIM SM traffic instead of forwarding it. The device forwards PIM SM traffic to a receiver only when the device receives a join message from the receiver. Consequently, if the source and the downstream router are in the same sub-net, and PIM SM traffic snooping is enabled, the device blocks the PIM SM traffic and never starts forwarding the traffic. This is because the device never receives a join message from the downstream router for the group. The downstream router and group find each other without a join message because they are in the same sub-net.

NOTE: If the “route-only” feature is enabled on a Layer 3 Switch, PINM SM traffic snooping will not be supported.

Enabling PIM SM Traffic Snooping

To enable PIM SM traffic snooping, you must enable IP multicast traffic reduction, then enable snooping. To enable PIM SM traffic snooping, enter the following commands at the global CONFIG level of the CLI:

```
FESX424 Switch(config)# ip multicast
FESX424 Switch(config)# ip pimsm-snooping
```

The first command enables IP multicast traffic reduction. This feature is similar to PIM SM traffic snooping but listens only for IGMP information, not PIM SM information. You must enable both IP multicast traffic reduction and PIM SM traffic snooping to enable the device to listen for PIM SM join and prune messages.

Syntax: [no] ip multicast [active | passive]

This command enables IP multicast traffic reduction. The **active | passive** parameter specifies the mode. The PIM SM traffic snooping feature assumes that the network has routers that are running PIM SM.

Syntax: [no] ip pimsm-snooping

This command enables PIM SM traffic snooping.

To disable the feature, enter the following command:

```
FESX424 Switch(config)# no ip pimsm-snooping
```

If you also want to disable IP multicast traffic reduction, enter the following command:

```
FESX424 Switch(config)# no ip multicast
```

Displaying IP Multicast Information

The following sections show how to display and clear IP multicast reduction information.

Displaying Multicast Information on Layer 2 Switches

To display IP multicast information on Layer 2 Switches, including the state of the traffic reduction and traffic snooping features, use the following CLI methods.

To display IP multicast information, enter the following command at any level of the CLI:

```
FastIron SuperX Switch# show ip multicast
IP multicast is enabled - Passive
VLAN ID 100
   Querier: 1.100.100.7, (port: 3/1)
   Router Ports:  3/1 3/2 3/3
   Total number of Multicast Group in vlan: 3
1     Group: 224.0.1.22, fid 08ac, NO cam
   Forwarding Port: 3/3
2     Group: 239.255.162.2, fid 08aa, cam 8
   Forwarding Port: 3/1 3/2
3     Group: 239.255.163.2, fid 08a9, cam 10
   Forwarding Port: 3/1 3/2

VLAN ID 4008
   Querier: 1.1.5.1, (port: 3/48)
   Router Ports:  3/48
   Total number of Multicast Group in vlan: 0
```

Syntax: show ip multicast

This display shows the following information.

This Field...	Displays...
The IP multicast traffic snooping state	The first line of the display indicates whether IP multicast traffic snooping is enabled or disabled. If enabled, it indicates if the feature is configured as passive or active.
VLAN ID	The port-based VLAN to which the information listed applies.
Querier	The IP address of the device that actively sends IGMP queries.
(port)	The port on which the queries are being sent out.
Router Ports	The ports that are connected to routers that support IP multicast.
Total Number of Multicast Group in VLAN	The total number of groups for which the VLAN's ports have received IGMP group membership reports, join messages, or prune messages.
Multicast Group	Address of the IP multicast group. **Note**: The fid and camindex values are used by Foundry Technical Support for troubleshooting.
Forwarding Port	The forwarding ports for the IP multicast group.

You also can display PIM SM information on Layer 2 Switches by entering the following command, at any level of the CLI:

```
FastIron SuperX Router(config)# show ip pim
PIMSM snooping is enabled
VLAN ID 22
PIMSM Neighbor list:
        5.5.5.2 : 3/4   expire 95 s
        5.5.5.3 : 3/10   expire 180 s
        5.5.5.1 : 5/3   expire 160 s
Multicast Group: 239.255.162.1, fid 000026bc camindex 2058
   Forwarding Port: 3/4 3/10 5/3
   PIMv2 Group Port: 3/4 3/10 5/3
   (Source, Port) list:
       55.55.55.2, port: 3/10 5/3
       42.42.42.42, port: 3/4 3/10
       5.5.5.1, port: 3/10
       162.162.162.162, port: 3/4 5/3
```

Syntax: show ip pim

This display shows the following information.

This Field...	Displays...
The PIM SM traffic snooping state	The first line of the display indicates whether the feature is enabled or disabled.
VLAN ID	The port-based VLAN to which the neighbors and groups listed below the VLAN ID apply. Each port-based VLAN is a separate Layer 2 broadcast domain. **Note**: PIM SM traffic snooping requires the source and the receivers to be in the same port-based VLAN on the device. If the source and receivers are in different port-based VLANs, the device blocks the multicast traffic.
PIM SM Neighbor list	The PIM SM routers that are attached to the device's ports in the VLAN. The value following "expire" indicates how many seconds the device will wait for a hello message from the neighbor before determining that the neighbor is no longer present and removing the neighbor from the list.
Multicast Group	The IP multicast group ID. **Note**: The fid and camindex values are used by Foundry Technical Support for troubleshooting.
Forwarding Port	The port(s) attached to the group's receivers. A port is listed here when it receives a join message for the group, an IGMP membership report for the group, or both.
PIMv2 Group Port	The port(s) on which the device has received PIM SM join messages for the group.
Source, Port list	The IP address of each PIM SM source and the device ports connected to the receivers of the source.

Displaying Multicast Information for a Specific Group

You can display multicast information for a specific group, by entering a command such as the following at any level of the CLI:

```
FastIron SuperX Switch# show ip multicast 239.255.162.2
VLAN ID 100
Active 1.100.100.7 Router Ports  3/1 3/2 3/3
 Group: 239.255.162.2, fid 08aa, cam 8
   Forwarding Port: 3/1 3/2
group 239.255.162.2 in 1 vlans
```

Syntax: show ip multicast <group-address>

This display shows the following information.

This Field...	Displays...
VLAN ID	The port-based VLAN to which the information listed applies.
Active	The IP address of the device that actively sends IGMP queries.
Router Ports	The ports that are connected to routers that support IP multicast.
Group	Address of the IP multicast group. **Note**: The fid and camindex values are used by Foundry Technical Support for troubleshooting.
Forwarding Port	The forwarding ports for the IP multicast group.

Displaying Usage of Hardware Resource by Multicast Groups

You can display how much hardware resource (CAM and FID) is currently being used by multicast groups by entering a command such as the following at any level of the CLI:

```
FastIron SuperX Switch# show ip multicast hardware
Hw resource is shared by groups with the same lower 23 bits
VLAN ID 2
Total number of HW resource in vlan: 1
1     Group: 0.1.1.1, HW-ref-cnt=1, vidx 8191
   Forwarding Port: 2
VLAN ID 1
Total number of HW resource in vlan: 0
```

If you want to display the amount of hardware resource that is currently being used by a specific group, enter a command such as the following at any level of the CLI:

```
FastIron SuperX Switch# show ip multicast hardware 239.255.163.2
VLAN ID 100
 Group: 239.255.163.2, HW-ref-cnt=1, fid 08a9, cam 10, dma=8,
   Forwarding Port: 1 2
group 239.255.163.2 in 1 vlans
```

Syntax: show ip multicast hardware [<group-address> | vlan <vlan-id>]

Enter the address of a group for <group-address> if you want to display the hardware resource usage of a particular group.

Likewise, enter the ID of a VLAN for <vlan-id> if you want display the hardware resource usage of groups in a VLAN.

The display shows the following information:

This Field...	Displays...
VLAN ID	The port-based VLAN to which the information listed below applies.
Total number of HW resource in VLAN	The number of resources in the VLAN.
Group	Address of the IP multicast group that is using the entry. In the display above, group “0.1.1.1” is using this entry. The field HW-ref-cnt shows the number of groups that are sharing this entry. Multiple groups could share one entry because only low 23 bits are significant. **Note**: The vidx, fid, cam, and dma values are used by Foundry Technical Support for troubleshooting.
Forwarding Port	The forwarding ports for the IP multicast group.

Displaying Software Resource Usage

You can display the amount of software resources used by each IGMP and PIM process that is enabled on a Layer 2 Switch by entering the following command at any level of the CLI:

```
FastIron SuperX Router# show ip multicast resource
                      alloc in-use  avail allo-fail up-limit   size
pim neighbor             32      3     29         0      512     19
pim source-hash         256      3    253         0    10000    484
pim source             1024      7   1017         0   400000      6
pim source port        1024      7   1017         0   200000     13
igmp vlan struct         16      2     14         0      255    479
igmp mdb                256      3    253         0    10000    385
igmp hw resource        256      3    253         0    10000   5786
igmp port-age          2048      8   2040         0   100000      8
igmp leave              512      0    512         0 no-limit      8
In use: hw-res: 3, cam: 2, fid: 6
cpu forwarded packets: 411
```

Syntax: show ip multicast resource

Displaying Multicast Traffic Statistics

The **show ip multicast statistics** command shows the following message if the Layer 2 Switch receives an IGMP V3 report:

```
*** Warning! IGMPv3 reports: 10
```

The warning shows the count of IGMP V3 reports that were received by a Layer 2 Switch. Refer to the *Foundry Switch and Router Command Line Interface Reference* for information on this command.

Displaying Multicast Information by VLAN

You can display multicast information for a specific VLAN by entering a command such as the following at any level of the CLI:

```
FESX424 Router# show ip multicast vlan 100
Only display vlan 100
VLAN ID 100
   Querier: 1.100.100.7, (port: 3/1)
   Router Ports:  3/1 3/2
   Total number of Multicast Group in vlan: 3
1     Group: 224.0.1.22, fid 08ac, NO cam
   Forwarding Port: 3/3
2     Group: 239.255.162.2, fid 08aa, cam 8
   Forwarding Port: 3/1 3/2
3     Group: 239.255.163.2, fid 08a9, cam 10
   Forwarding Port: 3/1 3/2
```

Syntax: show ip multicast vlan <vlan-id>

Enter the ID of the VLAN for <vlan-id>.

This Field...	Displays...
VLAN ID	The port-based VLAN to which the information listed below the VLAN ID applies. Each port-based VLAN is a separate Layer 2 broadcast domain.
Querier	The IP address of the device that actively sends IGMP queries.
(port)	The port on which the queries are being sent out.
Router Ports	The ports that are connected to a switch that support IP multicast.
Total Number of Multicast Group in VLAN	The total number of groups for which the VLAN's ports have received IGMP group membership reports, join messages, or prune messages.
Multicast Group	Address of the IP multicast group. **Note**: The fid and camindex values are used by Foundry Technical Support for troubleshooting.
Forwarding Port	The forwarding ports for the IP multicast group.

Displaying PIM SM Snooping Information

You can display PIM SM snooping information for all groups by entering the following command at any level of the CLI on a Layer 2 Switch:

```
FastIron SuperX Switch# show ip pimsm-snooping vlan 100
VLAN ID 100, total 3 entries
PIMSM Neighbor list:
        1.100.100.12     : 3/3 expire 120 s
        1.100.100.10     : 3/2 expire 170 s
        1.100.100.7      : 3/1 expire 160 s
1     Group: 224.0.1.22, fid 08ac, NO cam
   Forwarding Port: 3/3
   PIMv2 Group Port: 3/3
   (Source, Port) list: 1 entries
2     Group: 239.255.162.2, fid 08aa, cam 8
   Forwarding Port: 3/1 3/2
   PIMv2 Group Port: 3/1 3/2
   (Source, Port) list: 3 entries
3     Group: 239.255.163.2, fid 08a9, cam 10
   Forwarding Port: 3/1 3/2
   PIMv2 Group Port: 3/1 3/2
   (Source, Port) list: 3 entries
VLAN ID 4008, total 0 entries
PIMSM Neighbor list:
```

Syntax: show ip pimsm-snooping vlan <vlan-id>

Enter the ID of the VLAN for the **vlan** <vlan-id> parameter.

If you want to display PIM SM snooping information for one source or one group, enter a command as in the following example. The command also displays the (source, port) list of the group.

```
FastIron SuperX Switch# show ip pimsm-snooping 239.255.163.2
Show pimsm snooping group 239.255.163.2 in all vlan
VLAN ID 100
 Group: 239.255.163.2, fid 08a9, cam 10
   Forwarding Port: 3/1 3/2
   PIMv2 Group Port: 3/1 3/2
   (Source, Port) list: 3 entries
       1    192.168.176.44, age=0, port: 3/2
       2    158.158.158.158, age=0, port: 3/1
       3    1.1.7.1, age=0, port: 3/2
```

Syntax: show ip pimsm-snooping <group-address> | <source-address>

If the address you entered is within the range of source addresses, then the router treats it as the source address. Likewise, if the address falls in the range of group addresses, then the router assumes that you are requesting a report for that group.

This display shows the following information.

This Field...	Displays...
VLAN ID	The port-based VLAN to which the information listed below apply and the number of members in the VLAN.
PIM SM Neighbor list	The PIM SM routers that are attached to the Layer 2 Switch's ports in the VLAN. The value following "expires" indicates how many seconds the Layer 2 Switch will wait for a hello message from the neighbor before determining that the neighbor is no longer present and removing the neighbor from the list.
Multicast Group	The IP address of the multicast group. **Note**: The fid and camindex values are used by Foundry Technical Support for troubleshooting.
Forwarding Port	The port(s) attached to the group's receivers. A port is listed here when it receives a join message for the group, an IGMP membership report for the group, or both.
PIMv2 Group Port	The port(s) on which the Layer 2 Switch has received PIM SM join messages for the group.
Source, Port list	The IP address of each PIM SM source and the Layer 2 Switch ports connected to the receivers of the source.

Displaying PIM SM Snooping Information for a Specific Source in a Group

You can display PIM SM snooping information for a specific (source, group) pair by entering commands such as the following at any level of the CLI:

```
FastIron SuperX Switch# show ip pimsm-snooping 239.255.163.2 192.168.176.44
Show pimsm snooping source 192.168.176.44, group 239.255.163.2 in all vlan
VLAN ID 100, G=239.255.163.2, S=192.168.176.44, age=0, port: 3/2
```

Syntax: show ip pimsm-snooping <group-address> <source-ip-address>

The Foundry device determines which address is the group address and which one is the source address based on the ranges that the address fall into. If the address is within the range of source addresses, then the router treats it as the source address. Likewise, if the address falls in the range of group addresses, then the router assumes it is a group address.

The output shows the following information.

This Field...	Displays...
VLAN ID	VLAN membership of the source
Group	Address of the group
Source	IP address of the source
Age	Age of the source.

This Field...	Displays...
Port	Port on which the source is sending traffic

Displaying IP Multicast Statistics

To display IP multicast statistics on a device, enter the following commands at any level of the CLI:

The command in this example shows statistics for two port-based VLANs.

Syntax: show ip multicast statistics

Clearing IP Multicast Statistics

To clear IP multicast statistics on a device, enter the following command at the Privileged EXEC level of the CLI:

```
FastIron SuperX Switch# clear ip multicast statistics
```

This command resets statistics counters for all the statistics displayed by the **show ip multicast statistics** command to zero.

Syntax: clear ip multicast statistics

Clearing IGMP Group Flows

To clear all the IGMP flows learned by the device, enter the following command at the Privileged EXEC level of the CLI:

```
FastIron SuperX Switch# clear ip multicast all
```


The following example shows IGMP flows information listed by the **show ip multicast** command, followed by removal of the information by the **clear ip multicast all** command.

```
FastIron SuperX Switch# show ip multicast
IP multicast is enabled - Active
VLAN ID 1
Active 192.168.2.30 Router Ports  4/13
Multicast Group: 239.255.162.5, Port: 4/4 4/13
Multicast Group: 239.255.162.4, Port: 4/10 4/13

FESX424 Router# clear ip multicast all

FESX424 Router# show ip multicast
IP multicast is enabled - Active
VLAN ID 1
Active 192.168.2.30 Router Ports  4/13
```

To clear the learned IGMP flows for a specific IP multicast group, enter a command such as the following:

```
FastIron SuperX Switch# clear ip multicast group 239.255.162.5
```

The following example shows how to clear the IGMP flows for a specific group and retain reports for other groups.

```
FastIron SuperX Switch# show ip multicast
IP multicast is enabled - Active
VLAN ID 1
Active 192.168.2.30 Router Ports  4/13
Multicast Group: 239.255.162.5, Port: 4/4 4/13
Multicast Group: 239.255.162.4, Port: 4/10 4/13

FESX424 Router# clear ip multicast group 239.255.162.5

FESX424 Router# show ip multicast
IP multicast is enabled - Active
VLAN ID 1
Active 192.168.2.30 Router Ports  4/13
Multicast Group: 239.255.162.4, Port: 4/10 4/13
```

Syntax: clear ip multicast all | group <group-id>

The **all** parameter clears the learned flows for all groups.

The **group** <group-id> parameter clears the flows for the specified group but does not clear the flows for other groups.


```
FastIron SuperX Switch# show ip multicast statistics
IP multicast is enabled - Passive

VLAN ID 1
Reports Received:                34
Leaves Received:                 21
General Queries Received:        60
Group Specific Queries Received: 2
Others Received:                 0
General Queries Sent:            0
Group Specific Queries Sent:     0

VLAN ID 2
Reports Received:                0
Leaves Received:                 0
General Queries Received:        60
Group Specific Queries Received: 2
Others Received:                 0
General Queries Sent:            0
Group Specific Queries Sent:     0
```


Chapter 19
Configuring IP Multicast Protocols

This chapter describes how to configure Foundry Layer 3 Switches for Protocol Independent Multicast (PIM) and Distance Vector Multicast Routing Protocol (DVMRP). Foundry Layer 3 Switches support the following IP multicast versions:

- Internet Group Management Protocol (IGMP) V1 and V2
- Internet Group Management Protocol (IGMP) V3
- PIM Dense mode (PIM DM) V1 (draft-ietf-pim-dm-05) and V2 (draft-ietf-pim-v2-dm-03)
- PIM Sparse mode (PIM SM) V2 (RFC 2362)
- DVMRP V2 (RFC 1075)

NOTE: Each of the multicast protocols uses IGMP. IGMP is automatically enabled on an interface when you configure PIM or DVMRP on an interface and is disabled on the interface if you disable PIM or DVMRP on the interface.

NOTE: This chapter applies only to IP multicast routing. To configure Layer 2 IP multicast features, see "Configuring IP Multicast Traffic Reduction" on page 18-1.

This chapter contains the following information:

Table 19.1: Chapter Contents

Overview of IP Multicasting

Multicast protocols allow a group or channel to be accessed over different networks by multiple stations (clients) for the receipt and transmit of multicast data.

Distribution of stock quotes, video transmissions such as news services and remote classrooms, and video conferencing are all examples of applications that use multicast routing.

Foundry Layer 3 Switches support two different multicast routing protocols—Distance Vector Multicast Routing Protocol (DVMRP) and Protocol-Independent Multicast (PIM) protocol along with the Internet Group Membership Protocol (IGMP).

PIM and DVMRP are broadcast and pruning multicast protocols that deliver IP multicast datagrams. The protocols employ reverse path lookup check and pruning to allow source-specific multicast delivery trees to reach all group members. DVMRP and PIM build a different multicast tree for each source and destination host group.

NOTE: Both DVMRP and PIM can concurrently operate on different ports of a Foundry Layer 3 Switch.

Multicast Terms

The following are commonly used terms in discussing multicast-capable routers. These terms are used throughout this chapter:

Node: Refers to a router or Layer 3 Switch.

Root Node: The node that initiates the tree building process. It is also the router that sends the multicast packets down the multicast delivery tree.

Upstream: Represents the direction from which a router receives multicast data packets. An ***upstream router*** is a node that sends multicast packets.

Downstream: Represents the direction to which a router forwards multicast data packets. A ***downstream router*** is a node that receives multicast packets from upstream transmissions.

Group Presence: Means that a multicast group has been learned from one of the directly connected interfaces. Members of the multicast group are present on the router.

Intermediate nodes: Routers that are in the path between source routers and leaf routers.

Leaf nodes: Routers that do not have any downstream routers.

Multicast Tree: A unique tree is built for each source group (S,G) pair. A multicast tree is comprised of a root node and one or more nodes that are leaf or intermediate nodes.

Changing Global IP Multicast Parameters

The following configurable parameters apply to PIM-DM, PIM-SM, and DVMRP.

- Maximum number of PIM or DVMRP groups – You can change the maximum number of groups of each type for which the software will allocate memory. By default, Layer 3 Switches support up to 1024 PIM groups and 1024 DVMRP groups.
- Internet Group Membership Protocol (IGMP) V1 and V2 parameters – You can change the query interval, group membership time, and maximum response time.
- Hardware forwarding of fragmented IP multicast packets – You can enable the Layer 3 Switch to forward all fragments of fragmented IP multicast packets in hardware.

Changing Dynamic Memory Allocation for IP Multicast Groups

Layer 3 Switches support up to 1024 PIM groups and 1024 DVMRP groups by default. Memory for the groups is allocated dynamically as needed. For each protocol, previous releases support a maximum of 255 groups and 255 IGMP memberships.

NOTE: The number of interface groups you can configure for DVMRP and PIM is unlimited; therefore, the **system-max dvmrp-max-int-group** and the **system-max pim-max-int-group** commands that define their maximum table sizes have been removed.

The software allocates memory globally for each group, and also allocates memory separately for each interface's IGMP membership in a multicast group. An interface becomes a member of a multicast group when the interface receives an IGMP group membership report. For example, if the Layer 3 Switch learns about one multicast group, global memory for one group is used. In addition, if three interfaces on the device receive IGMP group membership reports for the group, interface memory for three IGMP memberships also is used.

Since the same group can use multiple allocations of memory (one for the group itself and one for each interface's membership in the group), you can increase the maximum number of IGMP memberships, up to 8192.

NOTE: The total for IGMP memberships applies to the device, not to individual interfaces. You can have up to 8192 IGMP memberships on all the individual interfaces, not up to 8192 IGMP memberships on each interface.

Increasing the Number of IGMP Membership

To increase the number of IGMP membership interfaces you can have for PIM, enter commands such as the following:

```
FastIron SuperX Router(config)# system-max pim-max-int-group 4000
FastIron SuperX Router(config)# write memory
```

This command enables the device to have up to 4000 IGMP memberships for PIM.

NOTE: The **system-max pim-max-int-group** command is no longer available since you can configure an unlimited number of PIM interface groups for DVMRP.

Syntax: [no] system-max pim-max-int-group <num>

The <num> parameter specifies the maximum number of IGMP memberships for PIM, and can be from 256 – 8192.

To increase the number of IGMP memberships interfaces you can have for DVMRP, enter commands such as the following:

```
FastIron SuperX Router(config)# system-max dvmrp-max-int-group 3000
FastIron SuperX Router(config)# write memory
```

NOTE: The **system-max dvmrp-max-int-group** command is no longer available since you can configure an unlimited number of DVMRP interface groups.

Syntax: [no] system-max dvmrp-max-int-group <num>

The <num> parameter specifies the maximum number of IGMP memberships for DVMRP, and can be from 256 – 8192.

NOTE: You do not need to reload the software to place these changes into effect.

Defining the Maximum Number of Multicast Flows

The Multicast Flow table is shared by PIM and DVMRP. It defines the maximum number of flows for a PIM or DVMRP multicast switching that can be written in hardware (CAM). To define the maximum number of entries for the Multicast Flow table, enter a command such as the following:

```
FastIron SuperX Router(config)# system-max multicast-flow 2048
```

Syntax: system-max multicast-flow <num>

The <num> parameter specifies the maximum number of PIM and DVMRP multicast cache flows that can be stored in the CAM. Enter a number from 512 – 2048. The default is 1024.

NOTE: Do not set this maximum too high since you may run out of resources in the CAM.

Defining the Maximum Number of DVMRP Cache Entries

The DVMRP cache system parameter defines the maximum number of repeated DVMRP traffic being sent from the same source address and being received by the same destination address. To define this maximum, enter a command such as the following:

```
FastIron SuperX Router(config)# system-max dvmrp-mcache 500
```

Syntax: system-max dvmrp-mcache <num>

The <num> parameter specifies the maximum number of multicast cache entries for DVMRP. Enter a number from 128 – 2048. The default is 512.

Defining the Maximum Number of PIM Cache Entries

The PIM cache system parameter defines the maximum number of repeated PIM traffic being sent from the same source address and being received by the same destination address. To define this maximum, enter a command such as the following:

```
FastIron SuperX Router(config)# system-max pim-mcache 999
```

Syntax: system-max pim-mcache <num>

The <num> parameter specifies the maximum number of multicast cache entries for PIM. Enter a number from 256 – 4096. The default is 1024.

Changing IGMP V1 and V2 Parameters

IGMP allows Foundry routers to limit the multicast of IGMP packets to only those ports on the router that are identified as IP Multicast members. This section applies to Foundry devices that support IGMP versions 1 and 2.

The router actively sends out host queries to identify IP Multicast groups on the network, inserts the group information in an IGMP packet, and forwards the packet to IP Multicast neighbors.

The following IGMP V1 and V2 parameters apply to PIM and DVMRP:

- IGMP query interval – Specifies how often the Layer 3 Switch queries an interface for group membership. Possible values are 1 – 3600. The default is 60.
- IGMP group membership time – Specifies how many seconds an IP Multicast group can remain on a Layer 3 Switch interface in the absence of a group report. Possible values are 1 – 7200. The default is 60.
- IGMP maximum response time – Specifies how many seconds the Layer 3 Switch will wait for an IGMP response from an interface before concluding that the group member on that interface is down and removing the interface from the group. Possible values are 1 – 10. The default is 5.

To change these parameters, you must first enable IP multicast routing by entering the following CLI command at the global CLI level:

```
FastIron SuperX Router(config)# ip multicast-routing
```

Syntax: [no] ip multicast-routing

NOTE: You must enter the **ip multicast-routing** command before changing the global IP Multicast parameters. Otherwise, the changes do not take effect and the software uses the default values.

Modifying IGMP (V1 and V2) Query Interval Period

The IGMP query interval period defines how often a router will query an interface for group membership. Possible values are 1 – 3,600 seconds and the default value is 60 seconds.

To modify the default value for the IGMP (V1 and V2) query interval, enter the following:

```
FastIron SuperX Router(config)# ip igmp query 120
```

Syntax: ip igmp query-interval <1-3600>

Modifying IGMP (V1 and V2) Membership Time

Group membership time defines how long a group will remain active on an interface in the absence of a group report. Possible values are from 1 – 7200 seconds and the default value is 140 seconds.

To define an IGMP (V1 and V2) membership time of 240 seconds, enter the following:

```
FastIron SuperX Router(config)# ip igmp group-membership-time 240
```

Syntax: ip igmp group-membership-time <1-7200>

Modifying IGMP (V1 and V2) Maximum Response Time

Maximum response time defines how long the Layer 3 Switch will wait for an IGMP (V1 and V2) response from an interface before concluding that the group member on that interface is down and removing the interface from the group. Possible values are 1 – 10. The default is 5.

To change the IGMP (V1 and V2) maximum response time, enter a command such as the following at the global CONFIG level of the CLI:

```
FastIron SuperX Router(config)# ip igmp max-response-time 8
```

Syntax: [no] ip igmp max-response-time <num>

The <num> parameter specifies the number of seconds and can be a value from 1 – 10. The default is 5.

Adding an Interface to a Multicast Group

You can manually add an interface to a multicast group. This is useful in the following cases:

- Hosts attached to the interface are unable to add themselves as members of the group using IGMP.
- There are no members for the group attached to the interface.

When you manually add an interface to a multicast group, the Foundry device forwards multicast packets for the group but does not itself accept packets for the group.

You can manually add a multicast group to individual ports only. If the port is a member of a virtual routing interface, you must add the ports to the group individually.

To manually add a port to a multicast group, enter a command such as the following at the configuration level for the port:

```
FastIron SuperX Router(config-if-1/1)# ip igmp static-group 224.2.2.2
```

This command adds port 1/1 to multicast group 224.2.2.2.

To add a port that is a member of a virtual routing interface to a multicast group, enter a command such as the following at the configuration level for the virtual routing interface:

```
FastIron SuperX Router(config-vif-1)# ip igmp static-group 224.2.2.2 ethernet 5/2
```

This command adds port 5/2 in virtual routing interface 1 to multicast group 224.2.2.2.

Syntax: [no] ip igmp static-group <ip-addr> [ethernet <portnum>]

The <ip-addr> parameter specifies the group number.

The **ethernet** <portnum> parameter specifies the port number. Use this parameter if the port is a member of a virtual routing interface, and you are entering this command at the configuration level for the virtual routing interface.

Manually added groups are included in the group information displayed by the following commands:

- **show ip igmp group**
- **show ip pim group**

PIM Dense

NOTE: This section describes the “dense” mode of PIM, described in RFC 1075. See “PIM Sparse” on page 19-13 for information about PIM Sparse.

PIM was introduced to simplify some of the complexity of the routing protocol at the cost of additional overhead tied with a greater replication of forwarded multicast packets. PIM is similar to DVMRP in that PIM builds source-routed multicast delivery trees and employs reverse path check when forwarding multicast packets.

There are two modes in which PIM operates: Dense and Sparse. The Dense Mode is suitable for densely populated multicast groups, primarily in the LAN environment. The Sparse Mode is suitable for sparsely populated multicast groups with the focus on WAN.

PIM primarily differs from DVMRP by using the IP routing table instead of maintaining its own, thereby being routing protocol independent.

Initiating PIM Multicasts on a Network

Once PIM is enabled on each router, a network user can begin a video conference multicast from the server on R1 as shown in Figure 19.1. When a multicast packet is received on a PIM-capable router interface, the interface checks its IP routing table to determine whether the interface that received the message provides the shortest path back to the source. If the interface does provide the shortest path back to the source, the multicast packet is then forwarded to all neighboring PIM routers. Otherwise, the multicast packet is discarded and a prune message is sent back upstream.

In Figure 19.1, the root node (R1) is forwarding multicast packets for group 229.225.0.1, which it receives from the server, to its downstream nodes, R2, R3, and R4. Router R4 is an intermediate router with R5 and R6 as its downstream routers. Because R5 and R6 have no downstream interfaces, they are leaf nodes. The receivers in this example are those workstations that are resident on routers R2, R3, and R6.

Pruning a Multicast Tree

As multicast packets reach these leaf routers, the routers check their IGMP databases for the group. If the group is not in a router's IGMP database, the router discards the packet and sends a prune message to the upstream router. The router that discarded the packet also maintains the prune state for the source, group (S,G) pair. The branch is then pruned (removed) from the multicast tree. No further multicast packets for that specific (S,G) pair will be received from that upstream router until the prune state expires. You can configure the PIM Prune Timer (the length of time that a prune state is considered valid).

For example, in Figure 19.1 the sender with address 207.95.5.1 is sending multicast packets to the group 229.225.0.1. If a PIM router receives any groups other than that group, the router discards the group and sends a prune message to the upstream PIM router.

In Figure 19.2, Router R5 is a leaf node with no group members in its IGMP database. Therefore, the router must be pruned from the multicast tree. R5 sends a prune message upstream to its neighbor router R4 to remove itself from the multicast delivery tree and install a prune state, as seen in Figure 19.2. Router 5 will not receive any further multicast traffic until the prune age interval expires.

When a node on the multicast delivery tree has all of its downstream branches (downstream interfaces) in the prune state, a prune message is sent upstream. In the case of R4, if both R5 and R6 are in a prune state at the same time, R4 becomes a leaf node with no downstream interfaces and sends a prune message to R1. With R4 in a prune state, the resulting multicast delivery tree would consist only of leaf nodes R2 and R3.

Figure 19.1 Transmission of multicast packets from the source to host group members

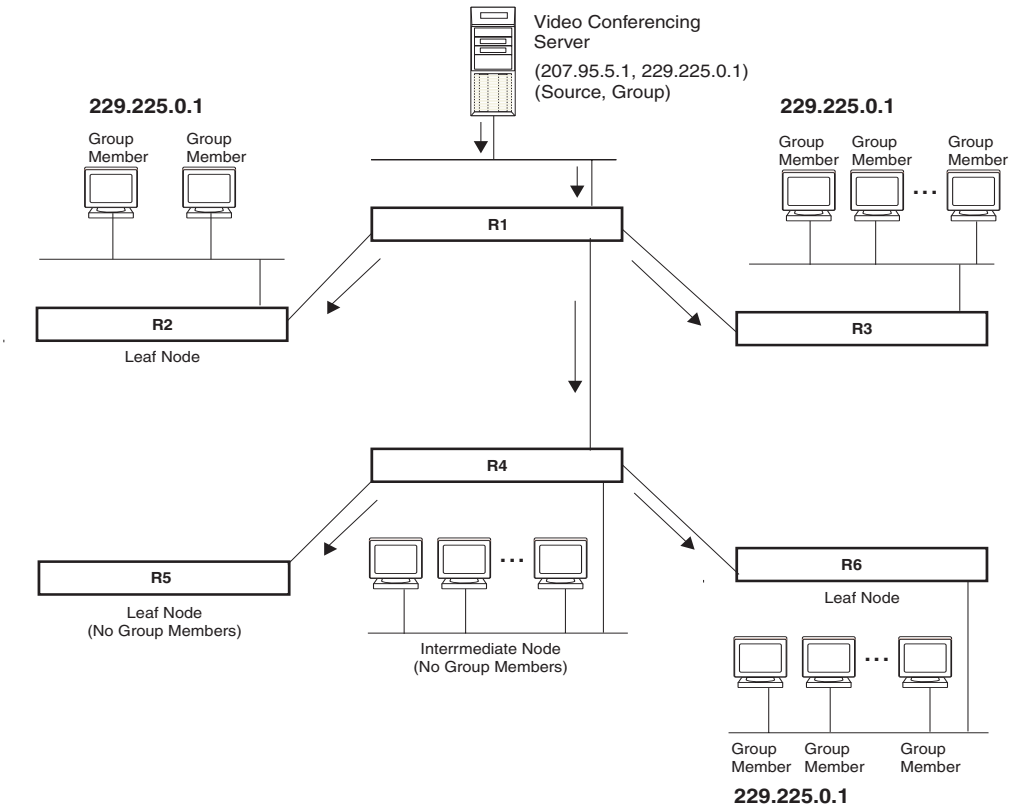
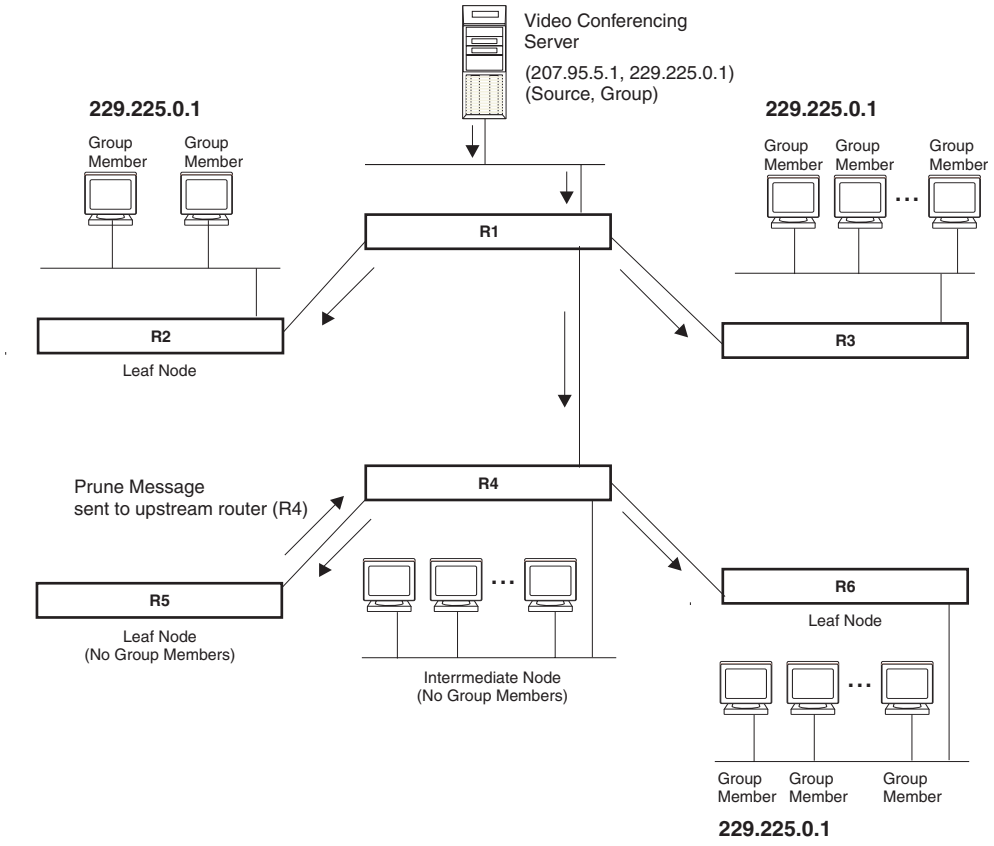


Figure 19.2 Pruning leaf nodes from a multicast tree



Grafts to a Multicast Tree

A PIM router restores pruned branches to a multicast tree by sending graft messages towards the upstream router. Graft messages start at the leaf node and travel up the tree, first sending the message to its neighbor upstream router.

In the example above, if a new 229.255.0.1 group member joins on router R6, which was previously pruned, a graft is sent upstream to R4. Since the forwarding state for this entry is in a prune state, R4 sends a graft to R1. Once R4 has joined the tree, R4 along with R6 once again receive multicast packets.

Prune and graft messages are continuously used to maintain the multicast delivery tree. No configuration is required on your part.

PIM DM Versions

Foundry devices support PIM DM V1 and V2. The default is V2. You can specify the version on an individual interface basis.

The primary difference between PIM DM V1 and V2 is the methods the protocols use for messaging:

- PIM DM V1 – uses the Internet Group Management Protocol (IGMP) to send messages
- PIM DM V2 – sends messages to the multicast address 224.0.0.13 (ALL-PIM-ROUTERS) with protocol number 103

The CLI commands for configuring and managing PIM DM are the same for V1 and V2. The only difference is the command you use to enable the protocol on an interface.

NOTE: Version 2 is the default PIM DM version. The only difference between version 1 and version 2 is the way the protocol sends messages. The change is not apparent in most configurations. You can use version 2 instead of version 1 with no impact to your network. However, if you want to continue to use PIM DM V1 on an interface, you must change the version, then save the configuration.

NOTE: The note above doesn't mean you can run different PIM versions on devices that are connected to each other. The devices must run the same version of PIM. If you want to connect a Layer 3 Switch running PIM to a device that is running PIM V1, you must change the version on the Layer 3 Switch to V1 (or change the version on the device to V2, if supported).

Configuring PIM DM

NOTE: This section describes how to configure the "dense" mode of PIM, described in RFC 1075. See "Configuring PIM Sparse" on page 19-15 for information about configuring PIM Sparse.

Enabling PIM on the Router and an Interface

By default, PIM is disabled. To enable PIM:

- Enable the feature globally.
- Configure the IP interfaces that will use PIM.
- Enable PIM locally on the ports that have the IP interfaces you configured for PIM.
- Reload the software to place PIM into effect.

Suppose you want to initiate the use of desktop video for fellow users on a sprawling campus network. All destination workstations have the appropriate hardware and software but the Foundry routers that connect the various buildings need to be configured to support PIM multicasts from the designated video conference server as shown in Figure 19.1 on page 19-7.

PIM is enabled on each of the Foundry routers shown in Figure 19.1, on which multicasts are expected. You can enable PIM on each router independently or remotely from one of the routers with a Telnet connection. Follow the same steps for each router. A reset of the router is required when PIM is first enabled. Thereafter, all changes are dynamic.

Globally Enabling and Disabling PIM

To globally enable PIM, enter the following command:

```
FastIron SuperX Router(config)# router pim
```

Syntax: [no] router pim

The behavior of the **[no] router pim** command is as follows:

- Entering **router pim** command to enable PIM does not require a software reload.
- Entering a **no router pim** command removes all configuration for PIM multicast on a Layer 3 Switch (**router pim** level) only.

Globally Enabling and Disabling PIM without Deleting Multicast Configuration

As stated above entering a **no router pim** command deletes the PIM configuration. If you want to disable PIM without deleting any PIM configuration, enter the following command:

```
FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# disable-pim
```

Syntax: [no] disable-pim

Use the [no] version of the command to re-enable PIM.

Enabling a PIM version

USING THE CLI

To enable PIM on an interface, globally enable PIM, then enable PIM on interface 3, enter the following commands:

```
FESX424 Router(config)# router pim
FESX424 Router(config)# int e 3
FESX424 Router(config-if-e1000-3)# ip address 207.95.5.1/24
FESX424 Router(config-if-e1000-3)# ip pim
FESX424 Router(config-if-e1000-3)# write memory
FESX424 Router(config-if-e1000-3)# end
FESX424 Router# reload
```

Syntax: [no] ip pim [version 1 | 2]

The **version 1 | 2** parameter specifies the PIM DM version. The default version is 2.

If you have enabled PIM version 1 but need to enable version 2 instead, enter either of the following commands at the configuration level for the interface:

```
FastIron SuperX Router(config-if-1/1)# ip pim version 2
```

```
FastIron SuperX Router(config-if-1/1)# no ip pim version 1
```

To disable PIM DM on the interface, enter the following command:

```
FastIron SuperX Router(config-if-1/1)# no ip pim
```

Modifying PIM Global Parameters

PIM global parameters come with preset values. The defaults work well in most networks, but you can modify the following parameters if you need to:

- Neighbor timeout
- Hello timer
- Prune timer
- Prune wait timer
- Graft retransmit timer
- Inactivity timer

Modifying Neighbor Timeout

Neighbor timeout is the interval after which a PIM router will consider a neighbor to be absent. Absence of PIM hello messages from a neighboring router indicates that a neighbor is not present.

The default value is 180 seconds.

To apply a PIM neighbor timeout value of 360 seconds to all ports on the router operating with PIM, enter the following:

```
FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# nbr-timeout 360
```

Syntax: nbr-timeout <60-8000>

The default is 180 seconds.

Modifying Hello Timer

This parameter defines the interval at which periodic hellos are sent out PIM interfaces. Routers use hello messages to inform neighboring routers of their presence. The default rate is 60 seconds.

To apply a PIM hello timer of 120 seconds to all ports on the router operating with PIM, enter the following:

```
FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# hello-timer 120
```


Syntax: hello-timer <10-3600>

The default is 60 seconds.

Modifying Prune Timer

This parameter defines how long a Foundry PIM router will maintain a prune state for a forwarding entry.

The first received multicast interface is forwarded to all other PIM interfaces on the router. If there is no presence of groups on that interface, the leaf node sends a prune message upstream and stores a prune state. This prune state travels up the tree and installs a prune state.

A prune state is maintained until the prune timer expires or a graft message is received for the forwarding entry. The default value is 180 seconds.

To set the PIM prune timer to 90, enter the following:

```
FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# prune-timer 90
```

Syntax: prune-timer <10-3600>

The default is 180 seconds.

Modifying the Prune Wait Timer

The CLI command **prune-wait** allows you to configure the amount of time a PIM router will wait before stopping traffic to neighbor routers that do not want the traffic. The value can be from zero to three seconds. The default is three seconds. A smaller prune wait value reduces flooding of unwanted traffic.

A prune wait value of zero causes the PIM router to stop traffic immediately upon receiving a prune message. If there are two or more neighbors on the physical port, then the **prune-wait** command should not be used because one neighbor may send a prune message while the other sends a join message at the during time or in less than three seconds.

To set the prune wait time to zero, enter the following commands:

```
FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# prune-wait 0
```

Syntax: prune-wait <time>

where <time> can be 0 - 3 seconds. A value of 0 causes the PIM router to stop traffic immediately upon receiving a prune message. The default is 3 seconds.

Viewing the Prune Wait Time

To view the prune wait time, enter the following command at any level of the CLI:

```
FastIron SuperX Router(config)#show ip pim dense

Global PIM Dense Mode Settings
Hello interval: 60, Neighbor timeout: 180
Graft Retransmit interval: 180, Inactivity interval: 180
Route Expire interval: 200, Route Discard interval: 340
Prune age: 180, Prune wait: 3
```

Modifying Graft Retransmit Timer

The Graft Retransmit Timer defines the interval between the transmission of graft messages.

A graft message is sent by a router to cancel a prune state. When a router receives a graft message, the router responds with a Graft Ack (acknowledge) message. If this Graft Ack message is lost, the router that sent the graft message will resend it.

To change the graft retransmit timer from the default of 180 to 90 seconds, enter the following:

```
FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# graft-retransmit-timer 90
```

Syntax: graft-retransmit-timer <10-3600>

The default is 180 seconds.

Modifying Inactivity Timer

The router deletes a forwarding entry if the entry is not used to send multicast packets. The PIM inactivity timer defines how long a forwarding entry can remain unused before the router deletes it.

To apply a PIM inactivity timer of 90 seconds to all PIM interfaces, enter the following:

```
FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# inactivity-timer 90
```

Syntax: inactivity-timer <10-3600>

The default is 180 seconds.

Selection of Shortest Path Back to Source

By default, when a multicast packet is received on a PIM-capable router interface in a multi-path topology, the interface checks its IP routing table to determine the shortest path back to the source. If the alternate paths have the same cost, the first alternate path in the table is picked as the path back to the source. For example, in the table below, the first four routes have the same cost back to the source. However, 137.80.127.3 will be chosen as the path to the source since it is the first one on the list. The router rejects traffic from any port other than Port V11 on which 137.80.127.3 resides.

```
    Total number of IP routes: 19
    B:BGP D:Connected  R:RIP  S:Static  O:OSPF *:Candidate default
            Destination     NetMask         Gateway         Port       Cost
Type
       ..
    9       172.17.41.4     255.255.255.252*137.80.127.3    v11        2
O
            172.17.41.4     255.255.255.252 137.80.126.3    v10        2
O
            172.17.41.4     255.255.255.252 137.80.129.1    v13        2
O
            172.17.41.4     255.255.255.252 137.80.128.3    v12        2
O
    10      172.17.41.8     255.255.255.252 0.0.0.0         1/2        1
D
```

When the Highest IP RPF feature is enabled, the selection of the shortest path back to the source is based on which Reverse Path Forwarding (RPF) neighbor in the IP routing table has the highest IP address, if the cost of the routes are the same. For example, in the table above, Gateway 137.80.129.1 will be chosen as the shortest path to the source because it is the RPF neighbor with the highest IP address.

When choosing the RPF, the router first checks the Multicast Routing Table. If the table is not available, it chooses an RPF from the IP Routing Table. Multicast route is configured using the **ip mroute** command.

To enable the Highest IP RPF feature, enter commands such as the following:

```
FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# highest-ip-rpf
```

The command immediately enables the Highest IP RPF feature; there is no need to reboot the device.

Syntax: [no] highest-ip-rpf

Entering the **no** version of the command disables the feature; the shortest path back to the source will be based on the first entry in the IP routing table. If some PIM traffic paths were selected based on the highest IP RPF, these paths are changed immediately to use the first RPF in the routing table.

Failover Time in a Multi-Path Topology

When a port in a multi-path topology fails, and the failed port is the input port of the downstream router, a new path is re-established within a few seconds, depending on the routing protocol being used.

No configuration is required for this feature.

Modifying the TTL

The TTL defines the minimum value required in a packet for it to be forwarded out of the interface.

For example, if the TTL for an interface is set at 10, it means that only those packets with a TTL value of 10 or more will be forwarded. Likewise, if an interface is configured with a TTL Threshold value of 1, all packets received on that interface will be forwarded. Possible TTL values are 1 to 31. The default TTL value is 1.

Configuration Notes

- If the TTL for an interface is greater than 1, PIM packets received on the interface are always forwarded in software because each packet's TTL must be examined. Therefore, Foundry does not recommend modifying the TTL under normal operating conditions.
- Multicast packets with a TTL value of 1 are switched within the same VLAN. These packets cannot be routed between different VLANs.

Configuration Syntax

To configure a TTL of 24, enter the following:

```
FastIron SuperX Router(config-if-3/24)# ip pim ttl 24
```

Syntax: ip pim ttl <1-31>

Dropping PIM Traffic in Hardware

Unwanted PIM Dense or PIM Sparse multicast traffic can be dropped in hardware on Layer 3 Switches. This feature does not apply to DVMRP traffic. Refer to "Passive Multicast Route Insertion" on page 19-31.

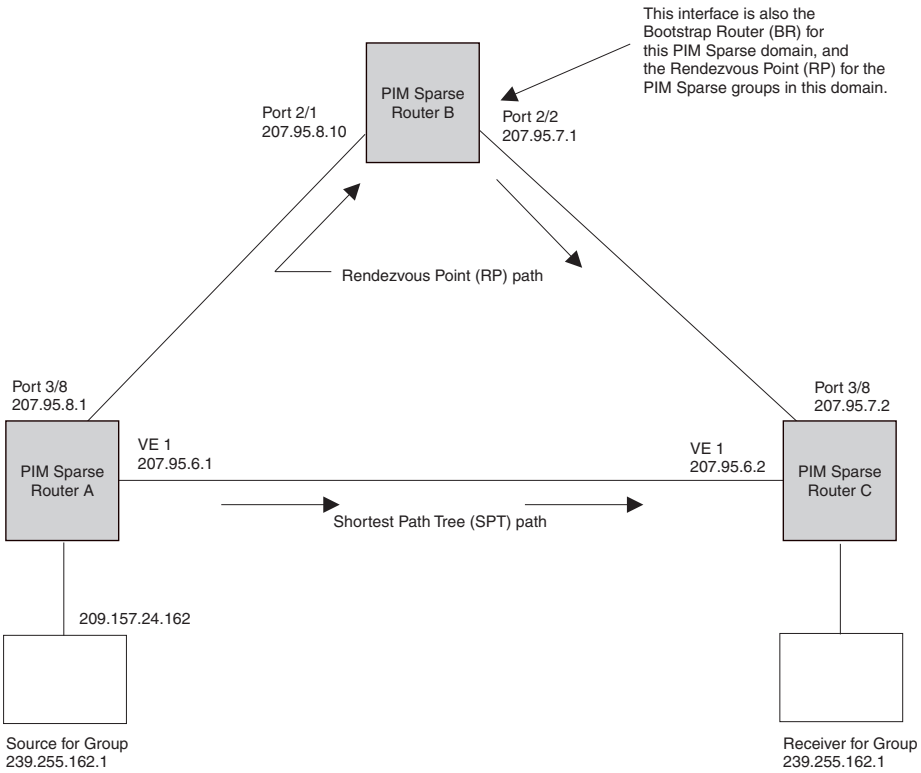
PIM Sparse

Foundry devices support Protocol Independent Multicast (PIM) Sparse version 2. PIM Sparse provides multicasting that is especially suitable for widely distributed multicast environments. The Foundry implementation is based on RFC 2362.

In a PIM Sparse network, a PIM Sparse router that is connected to a host that wants to receive information for a multicast group must explicitly send a join request on behalf of the receiver (host).

PIM Sparse routers are organized into domains. A PIM Sparse domain is a contiguous set of routers that all implement PIM and are configured to operate within a common boundary. Figure 19.3 shows a simple example of a PIM Sparse domain. This example shows three Layer 3 Switches configured as PIM Sparse routers. The configuration is described in detail following the figure.

Figure 19.3 Example PIM Sparse domain



PIM Sparse Router Types

Routers that are configured with PIM Sparse interfaces also can be configured to fill one or more of the following roles:

- PMBR – A PIM router that has some interfaces within the PIM domain and other interface outside the PIM domain. PBMRs connect the PIM domain to the Internet.

  NOTE: You cannot configure a Foundry routing interface as a PMBR interface for PIM Sparse in the current software release.

- BSR – The Bootstrap Router (BSR) distributes RP information to the other PIM Sparse routers within the domain. Each PIM Sparse domain has one active BSR. For redundancy, you can configure ports on multiple routers as candidate BSRs. The PIM Sparse protocol uses an election process to select one of the candidate BSRs as the BSR for the domain. The BSR with the highest BSR priority (a user-configurable parameter) is elected. If the priorities result in a tie, then the candidate BSR interface with the highest IP address is elected**.** In the example in Figure 19.3, PIM Sparse router B is the BSR. Port 2/2 is configured as a candidate BSR.

- RP – The RP is the meeting point for PIM Sparse sources and receivers. A PIM Sparse domain can have multiple RPs, but each PIM Sparse multicast group address can have only one active RP. PIM Sparse routers learn the addresses of RPs and the groups for which they are responsible from messages that the BSR sends to each of the PIM Sparse routers. In the example in Figure 19.3, PIM Sparse router B is the RP. Port 2/2 is configured as a candidate Rendezvous Point (RP).

  To enhance overall network performance, Foundry Layer 3 Switches use the RP to forward only the first packet from a group source to the group's receivers. After the first packet, the Layer 3 Switch calculates the shortest path between the receiver and source (the Shortest Path Tree, or SPT) and uses the SPT for subsequent packets from the source to the receiver. The Layer 3 Switch calculates a separate SPT for each source-receiver pair.

NOTE: Foundry Networks recommends that you configure the same ports as candidate BSRs and RPs.

RP Paths and SPT Paths

Figure 19.3 shows two paths for packets from the source for group 239.255.162.1 and a receiver for the group. The source is attached to PIM Sparse router A and the recipient is attached to PIM Sparse router C. PIM Sparse router B in is the RP for this multicast group. As a result, the default path for packets from the source to the receiver is through the RP. However, the path through the RP sometimes is not the shortest path. In this case, the shortest path between the source and the receiver is over the direct link between router A and router C, which bypasses the RP (router B).

To optimize PIM traffic, the protocol contains a mechanism for calculating the Shortest Path Tree (SPT) between a given source and receiver. PIM Sparse routers can use the SPT as an alternative to using the RP for forwarding traffic from a source to a receiver. By default, Foundry Layer 3 Switches forward the first packet they receive from a given source to a given receiver using the RP path, but forward subsequent packets from that source to that receiver through the SPT. In Figure 19.3, Layer 3 Switch A forwards the first packet from group 239.255.162.1's source to the destination by sending the packet to router B, which is the RP. Router B then sends the packet to router C. For the second and all future packets that router A receives from the source for the receiver, router A forwards them directly to router C using the SPT path.

Configuring PIM Sparse

To configure a Foundry Layer 3 Switch for PIM Sparse, perform the following tasks:

- Configure the following global parameter:
  - Enable the PIM Sparse mode of multicast routing.
- Configure the following interface parameters:
  - Configure an IP address on the interface
  - Enable PIM Sparse.
  - Identify the interface as a PIM Sparse border, if applicable.

NOTE: You cannot configure a Foundry routing interface as a PMBR interface for PIM Sparse in the current software release.

- Configure the following PIM Sparse global parameters:
  - Identify the Layer 3 Switch as a candidate PIM Sparse Bootstrap Router (BSR), if applicable.
  - Identify the Layer 3 Switch as a candidate PIM Sparse Rendezvous Point (RP), if applicable.
  - Specify the IP address of the RP (if you want to statically select the RP).

NOTE: Foundry Networks recommends that you configure the same Layer 3 Switch as both the BSR and the RP.

Limitations in this Release

The implementation of PIM Sparse in the current software release has the following limitations:

- PIM Border Routers (PMBRs) are not supported. Thus, you cannot configure a Foundry routing interface as a PMBR interface for PIM Sparse.
- PIM Sparse and regular PIM (dense mode) cannot be used on the same interface.
- You cannot configure or display PIM Sparse information using the Web management interface. (You can display some general PIM information, but not specific PIM Sparse information.)

Configuring Global PIM Sparse Parameters

To configure the PIM Sparse global parameters, use either of the following methods.

NOTE: When PIM routing is enabled on a FastIron device, the line rate for receive traffic is reduced by about 5%. The reduction occurs due to overhead from the VLAN multicasting feature, which PIM routing uses. This behavior is normal and does not indicate a problem with the device.

To configure basic global PIM Sparse parameters, enter commands such as the following on each Layer 3 Switch within the PIM Sparse domain:

```
FastIron SuperX Router(config)# router pim
```

Syntax: [no] router pim

NOTE: You do not need to globally enable IP multicast routing when configuring PIM Sparse.

The command in this example enables IP multicast routing, and enables the PIM Sparse mode of IP multicast routing. The command does not configure the Layer 3 Switch as a candidate PIM Sparse Bootstrap Router (BSR) and candidate Rendezvous Point (RP). You can configure a Foundry Layer 3 Switch as a PIM Sparse router without configuring the Layer 3 Switch as a candidate BSR and RP. However, if you do configure the Layer 3 Switch as one of these, Foundry Networks recommends that you configure the Layer 3 Switch as both of these. See “Configuring BSRs” on page 19-17.

The behavior of the **[no] router pim** command is as follows:

- Entering **no router pim** command to disable PIM or DVMRP does not require a software reload.
- Entering a **no router pim** command removes all configuration for PIM multicast on a Layer 3 Switch (**router pim** level) only.

Globally Enabling and Disabling PIM without Deleting the Multicast Configuration

As stated above entering a **no router pim** command deletes the PIM configuration. If you want to disable PIM without deleting any PIM configuration, enter the following command:

```
FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# disable-pim
```

Syntax: [no] disable-pim

Use the [no] version of the command to re-enable PIM.

Configuring PIM Interface Parameters

After you enable IP multicast routing and PIM Sparse at the global level, you must enable it on the individual interfaces connected to the PIM Sparse network. To do so, use the following CLI method.

To enable PIM Sparse mode on an interface, enter commands such as the following:

```
FastIron SuperX Router(config)# interface ethernet 2/2
FastIron SuperX Router(config-if-2/2)# ip address 207.95.7.1 255.255.255.0
FastIron SuperX Router(config-if-2/2)# ip pim-sparse
```

Syntax: [no] ip pim-sparse

The commands in this example add an IP interface to port 2/2, then enable PIM Sparse on the interface.

If the interface is on the border of the PIM Sparse domain, you also must enter the following command:

```
FastIron SuperX Router(config-if-2/2)# ip pim border
```

Syntax: [no] ip pim border

NOTE: You cannot configure a Foundry routing interface as a PMBR interface for PIM Sparse in the current software release.

Configuring BSRs

In addition to the global and interface parameters in the sections above, you need to identify an interface on at least one Layer 3 Switch as a candidate PIM Sparse Bootstrap router (BSR) and candidate PIM Sparse Rendezvous Point (RP).

NOTE: It is possible to configure the Layer 3 Switch as only a candidate BSR or RP, but Foundry Networks recommends that you configure the same interface on the same Layer 3 Switch as both a BSR and an RP.

This section presents how to configure BSRs. Refer to “Configuring RPs” on page 19-17 for instructions on how to configure RPs.

To configure the Layer 3 Switch as a candidate BSR and RP, enter commands such as the following:

```
FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# bsr-candidate ethernet 2/2 30 255
BSR address: 207.95.7.1, hash mask length: 30, priority: 255
```

This command configures the PIM Sparse interface on port 2/2 as a BSR candidate, with a hash mask length of 30 and a priority of 255. The information shown in italics above is displayed by the CLI after you enter the candidate BSR configuration command.

Syntax: [no] bsr-candidate ethernet [<slotnum>/]<portnum> | loopback <num> | ve <num> <hash-mask-length> [<priority>]

The <slotnum> parameter is required on chassis devices.

The <portnum> | **loopback** <num> | **ve** <num> parameter specifies the interface. The Layer 3 Switch will advertise the specified interface’s IP address as a candidate BSR.

- Enter **ethernet** [<slotnum>/] <portnum> for a physical interface (port).
- Enter **ve** <num> for a virtual interface.
- Enter **loopback** <num> for a loopback interface.

The <hash-mask-length> parameter specifies the number of bits in a group address that are significant when calculating the group-to-RP mapping. You can specify a value from 1 – 32.

NOTE: Foundry Networks recommends you specify 30 for IP version 4 (IPv4) networks.

The <priority> specifies the BSR priority. You can specify a value from 0 – 255. When the election process for BSR takes place, the candidate BSR with the highest priority becomes the BSR. The default is 0.

Configuring RPs

Enter a command such as the following to configure the Layer 3 Switch as a candidate RP:

```
FastIron SuperX Router(config-pim-router)# rp-candidate ethernet 2/2
```

Syntax: [no] rp-candidate ethernet [<slotnum>/]<portnum> | loopback <num> | ve <num>

The <slotnum> parameter is required on chassis devices.

The <portnum> | **loopback** <num> | **ve** <num> parameter specifies the interface. The Layer 3 Switch will advertise the specified interface’s IP address as a candidate RP.

- Enter **ethernet** [<slotnum>/]<portnum> for a physical interface (port).
- Enter **ve** <num> for a virtual interface.
- Enter **loopback** <num> for a loopback interface.

By default, this command configures the Layer 3 Switch as a candidate RP for all group numbers beginning with 224. As a result, the Layer 3 Switch is a candidate RP for all valid PIM Sparse group numbers. You can change this by adding or deleting specific address ranges. The following example narrows the group number range for which the Layer 3 Switch is a candidate RP by explicitly adding a range.

```
FastIron SuperX Router(config-pim-router)# rp-candidate add 224.126.0.0 16
```

Syntax: [no] rp-candidate add <group-addr> <mask-bits>

The <group-addr> <mask-bits> specifies the group address and the number of significant bits in the sub-net mask. In this example, the Layer 3 Switch is a candidate RP for all groups that begin with 224.126. When you add a range, you override the default. The Layer 3 Switch then becomes a candidate RP only for the group address range(s) you add.

You also can change the group numbers for which the Layer 3 Switch is a candidate RP by deleting address ranges. For example, to delete all addresses from 224.126.22.0 – 224.126.22.255, enter the following command:

```
FastIron SuperX Router(config-pim-router)# rp-candidate delete 224.126.22.0 24
```

Syntax: [no] rp-candidate delete <group-addr> <mask-bits>

The usage of the <group-addr> <mask-bits> parameter is the same as for the **rp-candidate add** command.

If you enter both commands shown in the example above, the net effect is that the Layer 3 Switch becomes a candidate RP for groups 224.126.0.0 – 224.126.21.255 and groups 224.126.23.0 – 224.126.255.255.

Updating PIM-Sparse Forwarding Entries with New RP Configuration

If you make changes to your static RP configuration, the entries in the PIM-Sparse multicast forwarding table continue to use the old RP configuration until they are aged out.

The **clear pim rp-map** command allows you to update the entries in the static multicast forwarding table immediately after making RP configuration changes. This command is meant to be used with **rp-address** command.

To update the entries in a PIM sparse static multicast forwarding table with new RP configuration, enter the following command at the privileged EXEC level of the CLI:

```
FastIron SuperX Router(config)# clear pim rp-map
```

Syntax: clear pim rp-map

Statically Specifying the RP

Foundry Networks recommends that you use the PIM Sparse protocol's RP election process so that a backup RP can automatically take over if the active RP router becomes unavailable. However, if you do not want the RP to be selected by the RP election process but instead you want to explicitly identify the RP by its IP address, you can do using the following CLI method.

If you explicitly specify the RP, the Layer 3 Switch uses the specified RP for all group-to-RP mappings and overrides the set of candidate RPs supplied by the BSR.

NOTE: Specify the same IP address as the RP on all PIM Sparse routers within the PIM Sparse domain. Make sure the router is on the backbone or is otherwise well connected to the rest of the network.

To specify the IP address of the RP, enter commands such as the following:

```
FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# rp-address 207.95.7.1
```

Syntax: [no] rp-address <ip-addr>

The <ip-addr> parameter specifies the IP address of the RP.

The command in the example above identifies the router interface at IP address 207.95.7.1 as the RP for the PIM Sparse domain. The Layer 3 Switch will use the specified RP and ignore group-to-RP mappings received from the BSR.

Changing the Shortest Path Tree (SPT) Threshold

In a typical PIM Sparse domain, there may be two or more paths from a DR (designated router) for a multicast source to a PIM group receiver.

- Path through the RP – This is the path the Layer 3 Switch uses the first time it receives traffic for a PIM group. However, the path through the RP may not be the shortest path from the Layer 3 Switch to the receiver.
- Shortest Path – Each PIM Sparse router that is a DR for a multicast source calculates a shortest path tree (SPT) to all the PIM Sparse group receivers within the domain, with the Layer 3 Switch itself as the root of the tree. The first time a Foundry Layer 3 Switch configured as a PIM router receives a packet for a PIM receiver, the Layer 3 Switch sends the packet to the RP for the group. The Layer 3 Switch also calculates the SPT from itself to the receiver. The next time the Layer 3 Switch receives a PIM Sparse packet for the receiver, the Layer 3 Switch sends the packet toward the receiver using the shortest route, which may not pass through the RP.

By default, the device switches from the RP to the SPT after receiving the first packet for a given PIM Sparse group. The Layer 3 Switch maintains a separate counter for each PIM Sparse source-group pair.

After the Layer 3 Switch receives a packet for a given source-group pair, the Layer 3 Switch starts a PIM data timer for that source-group pair. If the Layer 3 Switch does not receive another packet for the source-group pair before the timer expires, it reverts to using the RP for the next packet received for the source-group pair. In accordance with the PIM Sparse RFC's recommendation, the timer is 210 seconds and is not configurable. The counter is reset to zero each time the Layer 3 Switch receives a packet for the source-group pair.

You can change the number of packets that the Layer 3 Switch sends using the RP before switching to using the SPT. To do so, use the following CLI method.

```
FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# spt-threshold 1000
```

Syntax: [no] spt-threshold infinity | <num>

The **infinity** | <num> parameter specifies the number of packets. If you specify **infinity**, the Layer 3 Switch sends packets using the RP indefinitely and does not switch over to the SPT. If you enter a specific number of packets, the Layer 3 Switch does not switch over to using the SPT until it has sent the number of packets you specify using the RP.

Changing the PIM Join and Prune Message Interval

By default, the Layer 3 Switch sends PIM Sparse Join/Prune messages every 60 seconds. These messages inform other PIM Sparse routers about clients who want to become receivers (Join) or stop being receivers (Prune) for PIM Sparse groups.

You can change the Join/Prune message interval using the following CLI method.

NOTE: Use the same Join/Prune message interval on all the PIM Sparse routers in the PIM Sparse domain. If the routers do not all use the same timer interval, the performance of PIM Sparse can be adversely affected.

To change the Join/Prune interval, enter commands such as the following:

```
FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# message-interval 30
```

Syntax: [no] message-interval <num>

The <num> parameter specifies the number of seconds and can from 1 – 65535. The default is 60.

Dropping PIM Traffic in Hardware

Unwanted PIM Dense or PIM Sparse multicast traffic can be dropped in hardware on Layer 3 Switches. This feature does not apply to DVMRP traffic. Refer to "Passive Multicast Route Insertion" on page 19-31.

Displaying PIM Sparse Configuration Information and Statistics

You can display the following PIM Sparse information:

- Basic PIM Sparse configuration information
- Group information
- BSR information
- Candidate RP information
- RP-to-group mappings
- RP information for a PIM Sparse group
- RP set list
- PIM Neighbor information
- The PIM flow cache
- The PIM multicast cache
- PIM traffic statistics

Displaying Basic PIM Sparse Configuration Information

To display basic configuration information for PIM Sparse, enter the following command at any CLI level:

```
FastIron SuperX Router(config-pim-router)# show ip pim sparse

Global PIM Sparse Mode Settings
  Hello interval: 60, Neighbor timeout: 180
  Bootstrap Msg interval: 130, Candidate-RP Advertisement interval: 60
  Join/Prune interval: 60, SPT Threshold: 1

Interface Ethernet e3/8
TTL Threshold: 1, Enabled
Local Address: 207.95.8.1

Interface Ve 1
TTL Threshold: 1, Enabled
Local Address: 207.95.6.1
```

Syntax: show ip pim sparse

This example shows the PIM Sparse configuration information on PIM Sparse router A in Figure 19.3.

This display shows the following information.

This Field...	Displays...
Global PIM Sparse mode settings	
Hello interval	How frequently the Layer 3 Switch sends PIM Sparse hello messages to its PIM Sparse neighbors. This field show the number of seconds between hello messages. PIM Sparse routers use hello messages to discover one another.

This Field...	Displays...
Neighbor timeout	How many seconds the Layer 3 Switch will wait for a hello message from a neighbor before determining that the neighbor is no longer present and removing cached PIM Sparse forwarding entries for the neighbor.
Bootstrap Msg interval	How frequently the BSR configured on the Layer 3 Switch sends the RP set to the RPs within the PIM Sparse domain. The RP set is a list of candidate RPs and their group prefixes. A candidate RP's group prefix indicates the range of PIM Sparse group numbers for which it can be an RP. **Note**: This field contains a value only if an interface on the Layer 3 Switch is elected to be the BSR. Otherwise, the field is blank.
Candidate-RP Advertisement interval	How frequently the candidate PR configured on the Layer 3 Switch sends candidate RP advertisement messages to the BSR. **Note**: This field contains a value only if an interface on the Layer 3 Switch is configured as a candidate RP. Otherwise, the field is blank.
Join/Prune interval	How frequently the Layer 3 Switch sends PIM Sparse Join/Prune messages for the multicast groups it is forwarding. This field show the number of seconds between Join/Prune messages. The Layer 3 Switch sends Join/Prune messages on behalf of multicast receivers who want to join or leave a PIM Sparse group. When forwarding packets from PIM Sparse sources, the Layer 3 Switch sends the packets only on the interfaces on which it has received join requests in Join/Prune messages for the source's group. You can change the Join/Prune interval if needed. See "Changing the PIM Join and Prune Message Interval" on page 19-19.
SPT Threshold	The number of packets the Layer 3 Switch sends using the path through the RP before switching to using the SPT path.
PIM Sparse interface information **Note**: You also can display IP multicast interface information using the **show ip pim interface** command. However, this command lists all IP multicast interfaces, including regular PIM (dense mode) and DVMRP interfaces. The **show ip pim sparse** command lists only the PIM Sparse interfaces.	
Interface	The type of interface and the interface number. The interface type can be one of the following: • Ethernet • VE The number is either a port number (and slot number if applicable) or the virtual interface (VE) number.
TTL Threshold	Following the TTL threshold value, the interface state is listed. The interface state can be one of the following: • Disabled • Enabled
Local Address	Indicates the IP address configured on the port or virtual interface.

Displaying a List of Multicast Groups

To display a list of the IP multicast groups the Layer 3 Switch is forwarding, enter the following command at any CLI level:

```
FastIron SuperX Router(config-pim-router)# show ip pim group

Total number of Groups: 2
Index 1         Group 239.255.162.1       Ports e3/11
```

Syntax: show ip pim group

This display shows the following information.

This Field...	Displays...
Total number of Groups	Lists the total number of IP multicast groups the Layer 3 Switch is forwarding. **Note**: This list can include groups that are not PIM Sparse groups. If interfaces on the Layer 3 Switch are configured for regular PIM (dense mode) or DVMRP, these groups are listed too.
Index	The index number of the table entry in the display.
Group	The multicast group address
Ports	The Layer 3 Switch ports connected to the receivers of the groups.

Displaying BSR Information

To display BSR information, enter the following command at any CLI level:

```
FastIron SuperX Router(config-pim-router)# show ip pim bsr

PIMv2 Bootstrap information

This system is the elected Bootstrap Router (BSR)
  BSR address: 207.95.7.1
  Uptime: 00:33:52, BSR priority: 5, Hash mask length: 32
  Next bootstrap message in 00:00:20

Next Candidate-RP-advertisement in 00:00:10
  RP: 207.95.7.1
    group prefixes:
    224.0.0.0 / 4
Candidate-RP-advertisement period: 60
```

This example show information displayed on a Layer 3 Switch that has been elected as the BSR. The following example shows information displayed on a Layer 3 Switch that is not the BSR. Notice that some fields shown in the example above do not appear in the example below.

```
FastIron SuperX Router(config-pim-router)# show ip pim bsr

PIMv2 Bootstrap information
```



```
local BSR address  = 207.95.7.1
local BSR priority = 5
```

Syntax: show ip pim bsr

This display shows the following information.

This Field...	Displays...
BSR address or local BSR address	The IP address of the interface configured as the PIM Sparse Bootstrap Router (BSR). **Note**: If the word "local" does not appear in the field, this Layer 3 Switch is the BSR. If the word "local" does appear, this Layer 3 Switch is not the BSR.
Uptime	The amount of time the BSR has been running. **Note**: This field appears only if this Layer 3 Switch is the BSR.
BSR priority or local BSR priority	The priority assigned to the interface for use during the BSR election process. During BSR election, the priorities of the candidate BSRs are compared and the interface with the highest BSR priority becomes the BSR. **Note**: If the word "local" does not appear in the field, this Layer 3 Switch is the BSR. If the word "local" does appear, this Layer 3 Switch is not the BSR.
Hash mask length	The number of significant bits in the IP multicast group comparison mask. This mask determines the IP multicast group numbers for which the Layer 3 Switch can be a BSR. The default is 32 bits, which allows the Layer 3 Switch to be a BSR for any valid IP multicast group number. **Note**: This field appears only if this Layer 3 Switch is the BSR.
Next bootstrap message in	Indicates how many seconds will pass before the BSR sends its next Bootstrap message. **Note**: This field appears only if this Layer 3 Switch is the BSR.
Next Candidate-PR-advertisement message in	Indicates how many seconds will pass before the BSR sends its next candidate PR advertisement message. **Note**: This field appears only if this Layer 3 Switch is the BSR.
RP	Indicates the IP address of the Rendezvous Point (RP). **Note**: This field appears only if this Layer 3 Switch is the BSR.
group prefixes	Indicates the multicast groups for which the RP listed by the previous field is a candidate RP. **Note**: This field appears only if this Layer 3 Switch is the BSR.
Candidate-RP-advertisement period	Indicates how frequently the BSR sends candidate RP advertisement messages. **Note**: This field appears only if this Layer 3 Switch is the BSR.

Displaying Candidate RP Information

To display candidate RP information, enter the following command at any CLI level:

```
FastIron SuperX Router(config-pim-router)# show ip pim rp-candidate

Next Candidate-RP-advertisement in 00:00:10
  RP: 207.95.7.1
    group prefixes:
    224.0.0.0 / 4

  Candidate-RP-advertisement period: 60
```

This example show information displayed on a Layer 3 Switch that is a candidate RP. The following example shows the message displayed on a Layer 3 Switch that is not a candidate RP.

```
FastIron SuperX Router(config-pim-router)# show ip pim rp-candidate
```

This system is not a Candidate-RP.

Syntax: show ip pim rp-candidate

This display shows the following information.

This Field...	Displays...
Candidate-RP-advertisement in	Indicates how many seconds will pass before the BSR sends its next RP message. **Note**: This field appears only if this Layer 3 Switch is a candidate RP.
RP	Indicates the IP address of the Rendezvous Point (RP). **Note**: This field appears only if this Layer 3 Switch is a candidate RP.
group prefixes	Indicates the multicast groups for which the RP listed by the previous field is a candidate RP. **Note**: This field appears only if this Layer 3 Switch is a candidate RP.
Candidate-RP-advertisement period	Indicates how frequently the BSR sends candidate RP advertisement messages. **Note**: This field appears only if this Layer 3 Switch is a candidate RP.

Displaying RP-to-Group Mappings

To display RP-to-group-mappings, enter the following command at any CLI level:

```
FastIron SuperX Router(config-pim-router)# show ip pim rp-map
Number of group-to-RP mappings: 6

Group address        RP address
------------------------------
1 239.255.163.1  99.99.99.5
2 239.255.163.2  99.99.99.5
3 239.255.163.3  99.99.99.5
4 239.255.162.1  99.99.99.5
5 239.255.162.2  43.43.43.1
```



```
6 239.255.162.3   99.99.99.5
```

Syntax: show ip pim rp-map

This display shows the following information.

This Field...	Displays...
Group address	Indicates the PIM Sparse multicast group address using the listed RP.
RP address	Indicates the IP address of the Rendezvous Point (RP) for the listed PIM Sparse group.

Displaying RP Information for a PIM Sparse Group

To display RP information for a PIM Sparse group, enter the following command at any CLI level:

```
FastIron SuperX Router(config-pim-router)# show ip pim rp-hash 239.255.162.1

  RP: 207.95.7.1, v2
    Info source: 207.95.7.1, via bootstrap
```

Syntax: show ip pim rp-hash <group-addr>

The <group-addr> parameter is the address of a PIM Sparse IP multicast group.

This display shows the following information.

This Field...	Displays...
RP	Indicates the IP address of the Rendezvous Point (RP) for the specified PIM Sparse group. Following the IP address is the port or virtual interface through which this Layer 3 Switch learned the identity of the RP.
Info source	Indicates the IP address on which the RP information was received. Following the IP address is the method through which this Layer 3 Switch learned the identity of the RP.

Displaying the RP Set List

To display the RP set list, enter the following command at any CLI level:

```
FastIron SuperX Router(config)#show ip pim rp-set
Group address Static-RP-address Override
--------------------------------------------------
Access-List 44 99.99.99.5 On
Number of group prefixes Learnt from BSR: 1
Group prefix = 239.255.162.0/24 # RPs expected: 1
# RPs received: 1
RP 1: 43.43.43.1 priority=0 age=0
```

Syntax: show ip pim rp-set

This display shows the following information.

This Field...	Displays...
Number of group prefixes	The number f PIM Sparse group prefixes for which the RP is responsible.
Group prefix	Indicates the multicast groups for which the RP listed by the previous field is a candidate RP.
RPs expected/received	Indicates how many RPs were expected and received in the latest Bootstrap message.
RP <num>	Indicates the RP number. If there are multiple RPs in the PIM Sparse domain, a line of information for each of them is listed, and they are numbered in ascending numerical order.
priority	The RP priority of the candidate RP. During the election process, the candidate RP with the highest priority is elected as the RP.
age	The age (in seconds) of this RP-set. **Note**: If this Layer 3 Switch is not a BSR, this field contains zero. Only the BSR ages the RP-set.

Displaying Multicast Neighbor Information

To display information about the Layer 3 Switch's PIM neighbors, enter the following command at any CLI level:

```
FastIron SuperX Router(config-pim-router)# show ip pim nbr

Port Neighbor        Holdtime Age   UpTime
                     sec      sec   sec
e3/8   207.95.8.10      180      60    900
Port Neighbor        Holdtime Age   UpTime
                     sec      sec   sec
v1     207.95.6.2       180      60    900
```

Syntax: show ip pim nbr

This display shows the following information.

This Field...	Displays...
Port	The interface through which the Layer 3 Switch is connected to the neighbor.
Neighbor	The IP interface of the PIM neighbor interface.

This Field...	Displays...
Holdtime sec	Indicates how many seconds the neighbor wants this Layer 3 Switch to hold the entry for this neighbor in memory. The neighbor sends the Hold Time in its Hello packets. • If the Layer 3 Switch receives a new Hello packet before the Hold Time received in the previous packet expires, the Layer 3 Switch updates its table entry for the neighbor. • If the Layer 3 Switch does not receive a new Hello packet from the neighbor before the Hold time expires, the Layer 3 Switch assumes the neighbor is no longer available and removes the entry for the neighbor.
Age sec	The number of seconds since the Layer 3 Switch received the last hello message from the neighbor.
UpTime sec	The number of seconds the PIM neighbor has been up. This timer starts when the Layer 3 Switch receives the first Hello messages from the neighbor.

Displaying Information About an Upstream Neighbor Device

You can view information about the upstream neighbor device for a given source IP address for IP Protocol Independent Multicast (PIM) and Distance Vector Multicast Routing Protocol (DVMRP) packets. For PIM, the software uses the IP route table or multicast route table to lookup the upstream neighbor device. For DVMRP, the software uses the DVMRP route table to locate the upstream neighbor device.

Enter the following command at the Privileged EXEC level of the CLI:

```
FastIron SuperX Router# show ip pim rpf 1.1.20.2

directly connected or via an L2 neighbor
```

Syntax: show ip pim | dvmrp rpf <IP address>

where <IP address> is a valid source IP address

NOTE: If there are multiple equal cost paths to the source, the **show ip pim rpf** command output may not be accurate. If your system has multiple equal cost paths, use the command **sh ip pim mcache** to view information about the upstream neighbor. For more information about this command, see the *Foundry Switch and Router Command Line Interface Reference*.

Displaying the PIM Flow Cache

To display the PIM flow cache, enter the following command at any CLI level:

```
FastIron SuperX Router(config-pim-router)# show ip pim flowcache

     Source          Group           Parent CamFlags CamIndex  Fid     Flags
1    209.157.24.162  239.255.162.1   v2     00000700 2023      00004411 F
2    209.157.24.162  239.255.162.1   v2     00000700 201b      00004411 F
3    209.157.24.162  239.255.162.1   v2     00000700 201d      00004411 F
4    209.157.24.162  239.255.162.1   v2     00000700 201e      00004411 F
```

Syntax: show ip pim flowcache

This display shows the following information.

This Field...	Displays...
Source	Indicates the source of the PIM Sparse group.
Group	Indicates the PIM Sparse group.
Parent	Indicates the port or virtual interface from which the Layer 3 Switch receives packets from the group's source.
CamFlags	This field is used by Foundry technical support for troubleshooting.
CamIndex	This field is used by Foundry technical support for troubleshooting.
Fid	This field is used by Foundry technical support for troubleshooting.
Flags	This field is used by Foundry technical support for troubleshooting.

Displaying the PIM Multicast Cache

To display the PIM multicast cache, enter the following command at any CLI level:

```
FastIron SuperX Router(config-pim-router)# show ip pim mcache

1     (*,239.255.162.1) RP207.95.7.1 forward port v1, Count 2
      member ports ethe 3/3
      virtual ports v2
      prune ports
      virtual prune ports

2     (209.157.24.162,239.255.162.4) forward port v2, flags 00004900 Count 130
      member ports
      virtual ports
      prune ports
      virtual prune ports

3     (209.157.24.162,239.255.162.1) forward port v2, flags 00005a01 Count 12
      member ports ethe 3/8
      virtual ports
      prune ports
      virtual prune ports
```

Syntax: show ip pim mcache

This display shows the following information.

This Field...	Displays...
(*<source>*, *<group>*)	The comma-separated values in parentheses is a source-group pair. The *<source>* is the PIM source for the multicast *<group>*. For example, the following entry means source 209.157.24.162 for group 239.255.162.1: (209.157.24.162,239.255.162.1) If the *<source>* value is * (asterisk), this cache entry uses the RP path. The * value means “all sources”. If the *<source>* is a specific source address, this cache entry uses the SPT path.
RP<ip-addr>	Indicates the RP for the group for this cache entry. **Note**: The RP address appears only if the RPT flag is set to 1 and the SPT flag is set to 0 (see below).
forward port	The port through which the Layer 3 Switch reaches the source.
Count	The number of packets forwarded using this cache entry.
Sparse Mode	Indicates whether the cache entry is for regular PIM (dense mode) or PIM Sparse. This flag can have one of the following values: • 0 – The entry is not for PIM Sparse (and is therefore for the dense mode of PIM). • 1– The entry is for PIM Sparse.
RPT	Indicates whether the cache entry uses the RP path or the SPT path. The RPT flag can have one of the following values: • 0 – The SPT path is used instead of the RP path. • 1– The RP path is used instead of the SPT path. **Note**: The values of the RP and SPT flags are always opposite (one is set to 0 and the other is set to 1).
SPT	Indicates whether the cache entry uses the RP path or the SPT path. The SP flag can have one of the following values: • 0 – The RP path is used instead of the SPT path. • 1– The SPT path is used instead of the RP path. **Note**: The values of the RP and SPT flags are always opposite (one is set to 0 and the other is set to 1).
Register Suppress	Indicates whether the Register Suppress timer is running. This field can have one of the following values: • 0 – The timer is not running. • 1 – The timer is running.
member ports	Indicates the Layer 3 Switch physical ports to which the receivers for the source and group are attached. The receivers can be directly attached or indirectly attached through other PIM Sparse routers.

This Field...	Displays...
virtual ports	Indicates the virtual interfaces to which the receivers for the source and group are attached. The receivers can be directly attached or indirectly attached through other PIM Sparse routers.
prune ports	Indicates the physical ports on which the Layer 3 Switch has received a prune notification (in a Join/Prune message) to remove the receiver from the list of recipients for the group.
virtual prune ports	Indicates the virtual interfaces ports on which the Layer 3 Switch has received a prune notification (in a Join/Prune message) to remove the receiver from the list of recipients for the group.

Displaying PIM Traffic Statistics

To display PIM traffic statistics, use the following CLI method.

```
FastIron SuperX Router(config-pim-router)# show ip pim traffic

Port    Hello           J/P          Register       RegStop          Assert
     [Rx     Tx]    [Rx     Tx]    [Rx     Tx]    [Rx     Tx]    [Rx     Tx]
e3/8   19     19      32     0       0       0       37     0       0       0

Port    Hello           J/P          Register       RegStop          Assert
     [Rx     Tx]    [Rx     Tx]    [Rx     Tx]    [Rx     Tx]    [Rx     Tx]
v1     18     19      0      20      0       0       0      0       0       0

Port    Hello           J/P          Register       RegStop          Assert
     [Rx     Tx]    [Rx     Tx]    [Rx     Tx]    [Rx     Tx]    [Rx     Tx]
v2     0      19      0      0       0       16      0      0       0       0

Total 37      57     32      0       0       0       0      0       0       0
IGMP Statistics:
  Total Recv/Xmit 85/110
  Total Discard/chksum  0/0
```

Syntax: show ip pim traffic

NOTE: If you have configured interfaces for standard PIM (dense mode) on the Layer 3 Switch, statistics for these interfaces are listed first by the display.

This display shows the following information.

This Field...	Displays...
Port	The port or virtual interface on which the PIM interface is configured.
Hello	The number of PIM Hello messages sent or received on the interface.
J/P	The number of Join/Prune messages sent or received on the interface. **Note**: Unlike PIM dense, PIM Sparse uses the same messages for Joins and Prunes.

This Field...	Displays...
Register	The number of Register messages sent or received on the interface.
RegStop	The number of Register Stop messages sent or received on the interface.
Assert	The number of Assert messages sent or received on the interface.
Total Recv/Xmit	The total number of IGMP messages sent and received by the Layer 3 Switch.
Total Discard/chksum	The total number of IGMP messages discarded, including a separate counter for those that failed the checksum comparison.

Displaying and Clearing PIM Errors

If you want to determine how many PIM errors there are on the device, enter the following command:

```
FastIron SuperX Router# show ip pim error
**** Warning counter pim route change = 1
HW tagged replication enabled, SW processed pkts 0
```

Syntax: show ip pim error

This command displays the number of warnings and non-zero PIM errors on the device. This count can increase during transition periods such as reboots and topology changes; however, if the device is stable, the number of errors should not increase. If warnings keep increasing in a stable topology, then there may be a configuration error or problems on the device.

To clear the counter for PIM errors, enter the following command:

```
FastIron SuperX Router# clear pim counters
```

Syntax: clear pim counters

Passive Multicast Route Insertion

NOTE: This feature was introduced in software release 02.4.00.

Passive Multicast Route Insertion (PMRI) enables a Layer 3 switch running PIM Sparse to create an entry for a multicast route (e.g., (S,G)), with no directly attached clients or when connected to another PIM router (transit network).

PMRI is critical for Service Providers wanting to deliver IP-TV services or multicast-based video services. Service Providers, who have transit networks, distribute multicast-based video services to other Service Providers, regardless of whether a client subscribes to a video service.

To configure PMRI, enter the following command at the **router pim** level of the CLI:

```
FESX424 Router(config)# router pim
FESX424 Router#(config-pim-router)# hardware-drop
```

Syntax: [no] hardware-drop

When you enable PMRI, the **show ip pim mcache** command output displays the multicast cache entry along with a **drop** flag, indicating that the device is dropping packets in hardware. If the **HW** flag is set to 1 (**HW=1**), it implies

that the packets are being dropped in hardware. If the **HW** flag is set to 0, (**HW=0)**, it indicates that the packets are being processed in software. The following shows an example display output.

```
FESX424 router# show ip pim mcache
1 (10.10.10.18 226.0.1.56) in v10 (e1), cnt=2
Source is directly connected
Sparse Mode, RPT=0 SPT=1 REG=1 MSDP Adv=0 MSDP Create=0
fast=0 slow=0 pru=1 graft age drop
age=0s up-time=2m HW=1 L2-vidx=8191
```

DVMRP Overview

Foundry routers provide multicast routing with the ***Distance Vector Multicast Routing Protocol (DVMRP)*** routing protocol. DVMRP uses ***Internet Group Membership Protocol (IGMP)*** to manage the IP multicast groups.

DVMRP is a broadcast and pruning multicast protocol that delivers IP multicast datagrams to its intended receivers. The receiver registers the interested groups using IGMP. DVMRP builds a multicast delivery tree with the sender forming the root. Initially, multicast datagrams are delivered to all nodes on the tree. Those leaves that do not have any group members send ***prune messages*** to the upstream router, noting the absence of a group. The upstream router maintains a prune state for this group for the given sender. A prune state is aged out after a given configurable interval, allowing multicasts to resume.

DVMRP employs ***reverse path forwarding*** and ***pruning*** to keep source specific multicast delivery trees with the minimum number of branches required to reach all group members. DVMRP builds a multicast tree for each source and destination host group.

Initiating DVMRP Multicasts on a Network

Once DVMRP is enabled on each router, a network user can begin a video conference multicast from the server on R1. ***Multicast Delivery Trees*** are initially formed by source-originated multicast packets that are propagated to downstream interfaces as seen in Figure 19.4. When a multicast packet is received on a DVMRP-capable router interface, the interface checks its DVMRP routing table to determine whether the interface that received the message provides the shortest path back to the source. If the interface does provide the shortest path, the interface forwards the multicast packet to adjacent peer DVMRP routers, except for the router interface that originated the packet. Otherwise, the interface discards the multicast packet and sends a prune message back upstream. This process is known as ***reverse path forwarding***.

In Figure 19.4, the root node (R1) is forwarding multicast packets for group 229.225.0.2 that it receives from the server to its downstream nodes, R2, R3, and R4. Router R4 is an intermediate router with R5 and R6 as its downstream routers. Because R5 and R6 have no downstream interfaces, they are leaf nodes.

The receivers in this example are those workstations that are resident on routers R2, R3, and R6.

Pruning a Multicast Tree

After the multicast tree is constructed, ***pruning*** of the tree will occur after IP multicast packets begin to traverse the tree.

As multicast packets reach leaf networks (sub-nets with no downstream interfaces), the local IGMP database checks for the recently arrived IP multicast packet address. If the local database does not contain the address (the address has not been learned), the router prunes (removes) the address from the multicast tree and no longer receives multicasts until the prune age expires.

In Figure 19.5, Router 5 is a leaf node with no group members in its local database. Consequently, Router 5 sends a prune message to its upstream router. This router will not receive any further multicast traffic until the prune age interval expires.

Figure 19.4 Downstream broadcast of IP multicast packets from source host

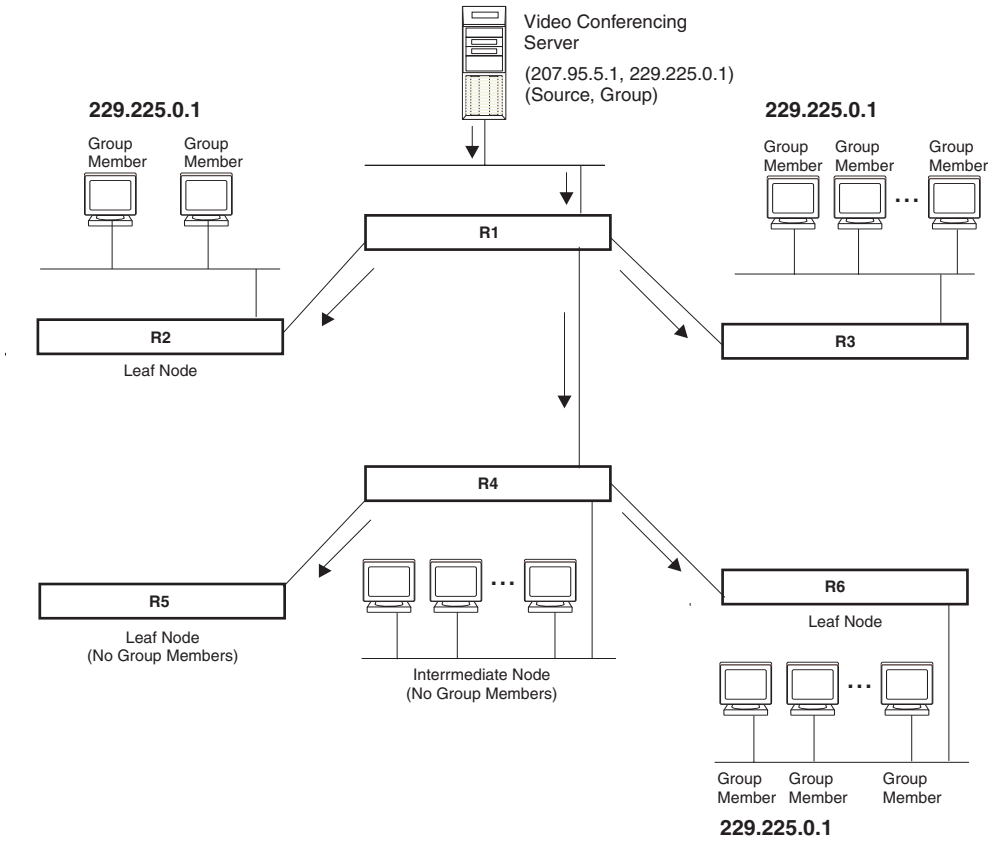
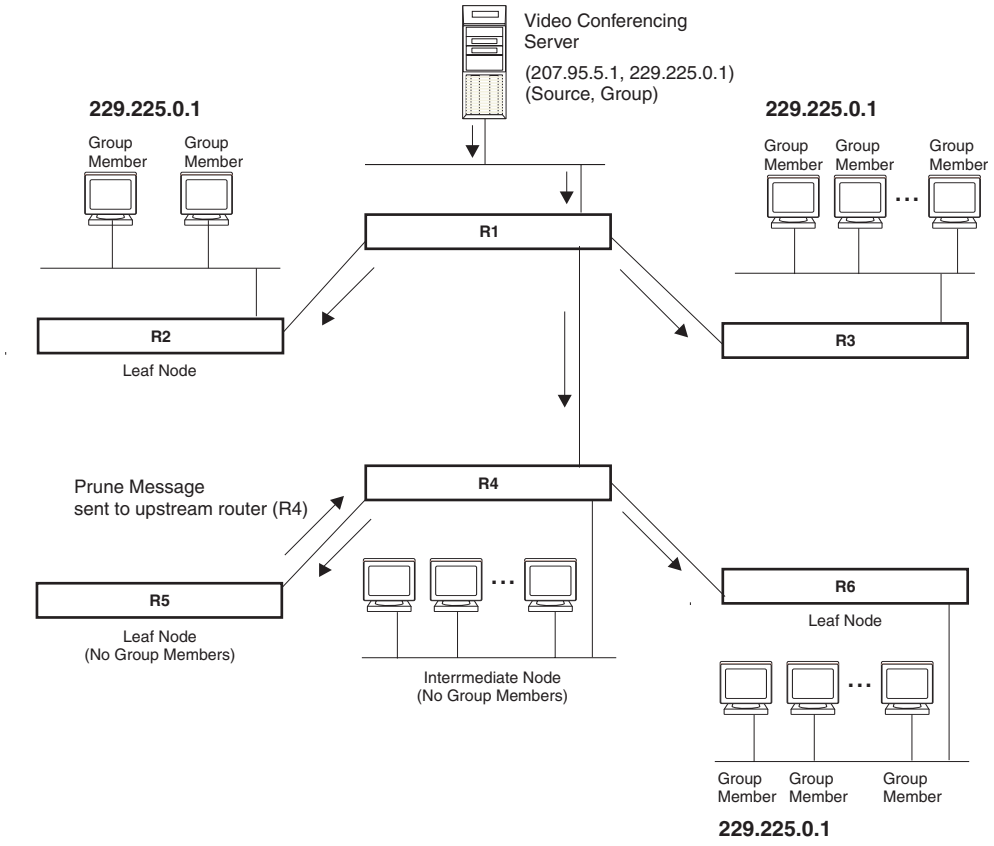


Figure 19.5 Pruning leaf nodes from a multicast tree



Grafts to a Multicast Tree

A DVMRP router restores pruned branches to a multicast tree by sending graft messages towards the upstream router. Graft messages start at the leaf node and travel up the tree, first sending the message to its neighbor upstream router.

In the example above, if a new 229.255.0.1 group member joins on router R6, which had been pruned previously, a graft will be sent upstream to R4. Since the forwarding state for this entry is in a prune state, R4 sends a graft to R1. Once R4 has joined the tree, it along with R6 will once again receive multicast packets.

You do not need to perform any configuration to maintain the multicast delivery tree. The prune and graft messages automatically maintain the tree.

Configuring DVMRP

Enabling DVMRP on the Layer 3 Switch and Interface

Suppose you want to initiate the use of desktop video for fellow users on a sprawling campus network. All destination workstations have the appropriate hardware and software but the Layer 3 Switches that connect the various buildings need to be configured to support DVMRP multicasts from the designated video conference server as seen in Figure 19.4.

DVMRP is enabled on each of the Foundry Layer 3 Switches shown in Figure 19.4, on which multicasts are expected. You can enable DVMRP on each Layer 3 Switch independently or remotely from one Layer 3 Switch by a Telnet connection. Follow the same steps for each router.

Globally Enabling and Disabling DVMRP

To globally enable DVMRP, enter the following command:

```
Router1(config)# router dvmrp
```

Syntax: [no] router dvmrp

The behavior of the **[no] router dvmrp** command is as follows:

- Entering a **router dvmrp** command to enable DVMRP does not require a software reload.
- Entering a **no router dvmrp** command removes all configuration for PIM multicast on a Layer 3 Switch (**router pim** level) only.

Globally Enabling or Disabling DVMRP without Deleting Multicast Configuration

As stated above enter **no router dvmrp** removed PIM configuration. If you want to disable or enable DVMRP without removing PIM configuration, enter the following command:

```
FastIron SuperX Router(config)# router dvmrp
FastIron SuperX Router(config-pim-router)# disable-dvmrp
```

Syntax: [no] disable-dvmrp

Use the [no] version of the command to re-enable DVMRP.

Enabling DVMRP on an Interface

After globally enabling DVMRP on a Layer 3 Switch, enable it on each interface that will support the protocol.

To enable DVMRP on Router 1 and interface 3, enter the following:

```
Router1(config)# router dvmrp
Router1(config-dvmrp-router)# int e 3
Router1(config-if-3)# ip dvmrp
```

Modifying DVMRP Global Parameters

DVMRP global parameters come with preset values. The defaults work well in most networks, but you can modify the following global parameters if you need to:

- Neighbor timeout
- Route expire time
- Route discard time
- Prune age
- Graft retransmit time
- Probe interval
- Report interval
- Trigger interval
- Default route

Modifying Neighbor Timeout

The neighbor timeout specifies the period of time that a router will wait before it defines an attached DVMRP neighbor router as down. Possible values are 40 – 8000 seconds. The default value is 180 seconds.

To modify the neighbor timeout value to 100, enter the following:

```
FastIron SuperX Router(config-dvmrp-router)# nbr 100
```

Syntax: nbr-timeout <40-8000>

The default is 180 seconds.

Modifying Route Expires Time

The Route Expire Time defines how long a route is considered valid in the absence of the next route update. Possible values are from 20 – 4000 seconds. The default value is 200 seconds.

To modify the route expire setting to 50, enter the following:

```
FastIron SuperX Router(config-dvmrp-router)# route-expire-timeout 50
```

Syntax: route-expire-timeout <20-4000>

Modifying Route Discard Time

The Route Discard Time defines the period of time before a route is deleted. Possible values are from 40 – 8000 seconds. The default value is 340 seconds.

To modify the route discard setting to 150, enter the following:

```
FastIron SuperX Router(config-dvmrp-router)# route-discard-timeout 150
```

Syntax: route-discard-timeout <40-8000>

Modifying Prune Age

The Prune Age defines how long a prune state will remain in effect for a source-routed multicast tree. After the prune age period expires, flooding will resume. Possible values are from 20 – 3600 seconds. The default value is 180 seconds.

To modify the prune age setting to 150, enter the following:

```
FastIron SuperX Router(config-dvmrp-router)# prune 25
```

Syntax: prune-age <20-3600>

Modifying Graft Retransmit Time

The Graft Retransmit Time defines the initial period of time that a router sending a graft message will wait for a graft acknowledgement from an upstream router before re-transmitting that message.

Subsequent retransmissions are sent at an interval twice that of the preceding interval. Possible values are from 5 – 3600 seconds. The default value is 10 seconds.

To modify the setting for graft retransmit time to 120, enter the following:

```
FastIron SuperX Router(config-dvmrp-router)# graft 120
```

Syntax: graft-retransmit-time <5-3600>

Modifying Probe Interval

The Probe Interval defines how often neighbor probe messages are sent to the ALL-DVMRP-ROUTERS IP multicast group address. A router's probe message lists those neighbor DVMRP routers from which it has received probes. Possible values are from 5 – 30 seconds. The default value is 10 seconds.

To modify the probe interval setting to 10, enter the following:

```
FastIron SuperX Router(config-dvmrp-router)# probe 10
```

Syntax: probe-interval <5-30>

Modifying Report Interval

The Report Interval defines how often routers propagate their complete routing tables to other neighbor DVMRP routers. Possible values are from 10 – 2000 seconds. The default value is 60 seconds.

To support propagation of DVMRP routing information to the network every 90 seconds, enter the following:

```
FastIron SuperX Router(config-dvmrp-router)# report 90
```

Syntax: report-interval <10-2000>

Modifying Trigger Interval

The Trigger Interval defines how often trigger updates, which reflect changes in the network topology, are sent. Example changes in a network topology include router up or down or changes in the metric. Possible values are from 5 – 30 seconds. The default value is 5 seconds.

To support the sending of trigger updates every 20 seconds, enter the following:

```
FastIron SuperX Router(config-dvmrp-router)# trigger-interval 20
```

Syntax: trigger-interval <5-30>

Modifying Default Route

To define the default gateway for DVMRP, enter the following:

```
FastIron SuperX Router(config-dvmrp-router)# default-gateway 192.35.4.1
```

Syntax: default-gateway <ip-addr>

Modifying DVMRP Interface Parameters

DVMRP global parameters come with preset values. The defaults work well in most networks, but you can modify the following interface parameters if you need to:

- TTL
- Metric
- Advertising

Modifying the TTL

The TTL defines the minimum value required in a packet in order for the packet to be forwarded out the interface. For example, if the TTL for an interface is set at 10 it means that only those packets with a TTL value of 10 or more are forwarded. Likewise, if an interface is configured with a TTL Threshold value of 1, all packets received on that interface are forwarded. Possible values are from 1 – 64. The default value is 1.

To set a TTL of 64, enter the following:

```
FastIron SuperX Router(config)# int e 1/4
FastIron SuperX Router(config-if-1/4)# ip dvmrp ttl 60
```

Syntax: ttl-threshold <1-64>

Modifying the Metric

The router uses the metric when establishing reverse paths to some networks on directly attached interfaces. Possible values are from 1 – 31 hops. The default is 1.

NOTE: This command is not supported on Foundry Layer 2 Switches.

To set a metric of 15 for a DVMRP interface, enter the following:

```
FastIron SuperX Router(config)# interface 3/5
FastIron SuperX Router(config-if-3/5)# ip dvmrp metric 15
```

Syntax: ip dvmrp metric <1-31>

Enabling Advertising

You can turn the advertisement of a local route on (enable) or off (disable) on the interface. By default, advertising is enabled.

To enable advertising on an interface, enter the following:

```
FastIron SuperX Router(config-if-1/4)# ip dvmrp advertise-local on
```

Syntax: advertise-local on | off

Displaying Information About an Upstream Neighbor Device

You can view information about the upstream neighbor device for a given source IP address for IP PIM packets. The software uses the IP route table or multicast route table to lookup the upstream neighbor device.

The following shows example messages that the Foundry device can display with this command.

```
FastIron SuperX Router# show ip dvmrp rpf 1.1.20.2

directly connected or via an L2 neighbor

FastIron SuperX Router# show ip dvmrp rpf 1.2.3.4

no route

FastIron SuperX Router# show ip dvmrp rpf 1.10.10.24

upstream neighbor=1.1.20.1 on v21 using ip route
```

Syntax: show ip dvmrp rpf <IP address>

where <IP address> is a valid source IP address

NOTE: If there are multiple equal cost paths to the source, the **show ip dvmrp rpf** command output may not be accurate. If your system has multiple equal cost paths, use the command **sh ip dvmrp mcache** to view information about the upstream neighbor. For more information about this command, see the *Foundry Switch and Router Command Line Interface Reference*.

Configuring an IP Tunnel

IP tunnels are used to send traffic through routers that do not support IP multicasting. IP Multicast datagrams are encapsulated within an IP packet and then sent to the remote address. Routers that are not configured for IP Multicast route that packet as a normal IP packet. When the IP Multicast router at the remote end of the tunnel receives the packet, the router strips off the IP encapsulation and forwards the packet as an IP Multicast packet.

NOTE: An IP tunnel must have a remote IP interface at each end. Also, for IP tunneling to work, the remote routers must be reachable by an IP routing protocol.

NOTE: Multiple tunnels configured on a router cannot share the same remote address.

EXAMPLE:

To configure an IP tunnel as seen in Figure 19.6, enter the IP tunnel destination address on an interface of the router.

To configure an IP address on Router A, enter the following:

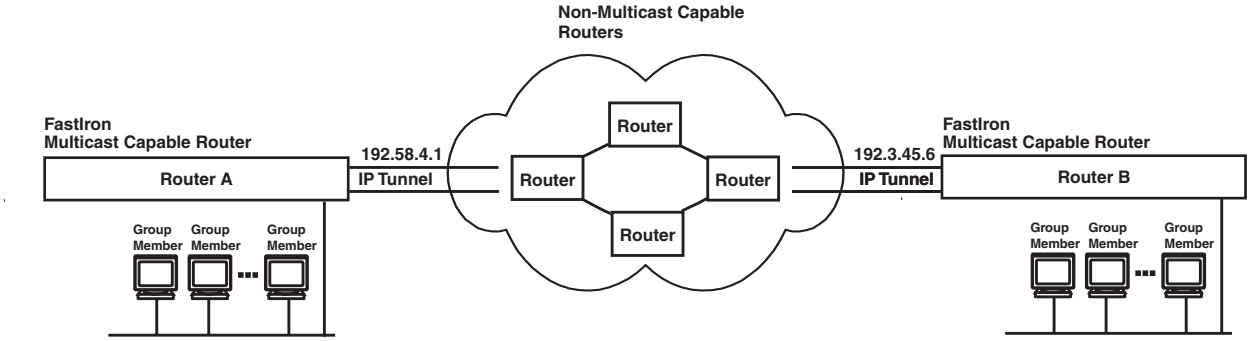
```
FastIron(config)# int e1
FastIron(config-if-1)# ip tunnel 192.3.45.6
```

NOTE: The IP tunnel address represents the configured IP tunnel address of the destination router. In the case of Router A, its destination router is Router B. Router A is the destination router of Router B.

For router B, enter the following:

```
FastIron(config-if-1)# ip tunnel 192.58.4.1
```


Figure 19.6 IP in IP tunneling on multicast packets in a unicast network



Using ACLs to Control Multicast Features

You can use ACLs to control the following multicast features:

- Limit the number of multicast groups that are covered by a static rendezvous point (RP)
- Control which multicast groups for which candidate RPs sends advertisement messages to bootstrap routers
- Identify which multicast group packets will be forwarded or blocked on an interface

Using ACLs to Limit Static RP Groups

You can limit the number of multicast groups covered by a static RP using standard ACLs. In the ACL, you specify the group to which the RP address applies. The following examples set the RP address to be applied to multicast groups with some minor variations.

To configure an RP that covers multicast groups in 239.255.162.x, enter commands such as the following:

```
FastIron SuperX Router(config)# access-list 2 permit 239.255.162.0 0.0.0.255

FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# rp-address 43.43.43.1 2
```

To configure an RP that covers multicast groups in the 239.255.162.x range, except the 239.255.162.2 group, enter commands such as the following:

```
FastIron SuperX Router(config)# access-list 5 deny host 239.255.162.2

FastIron SuperX Router(config)# access-list 5 permit 239.255.0.0 0.0.255.255

FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# bsr-candidate ve 43 32 100
FastIron SuperX Router(config-pim-router)# rp-candidate ve 43
FastIron SuperX Router(config-pim-router)# rp-address 99.99.99.5 5
```

To configure an RP for multicast groups using the override switch, enter commands such as the following:

```
FastIron SuperX Router(config)# access-list 44 permit 239.255.162.0 0.0.0.255

FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# rp-address 43.43.43.1
FastIron SuperX Router(config-pim-router)# rp-address 99.99.99.5 44 override
```

Syntax: [no] rp-address <ip-address> [<access-list-num>] [override]

The access-list-num parameter is the number of the standard ACL that will filter the multicast group.

NOTE: Extended ACLs cannot be used to limit static RP groups.

The **override** parameter directs the Layer 3 Switch to ignore the information learned by a BSR if there is a conflict between the RP configured in this command and the information that is learned by the BSR. In previous releases, static RP configuration precedes the RP address learned from the PIM Bootstrap protocol. With this enhancement, an RP address learned dynamically from PIM Bootstrap protocol takes precedence over static RP configuration unless the override parameter is used.

You can use the **show ip pim rp-set** command to display the ACLs used to filter the static RP groups. For example,

```
FastIron SuperX Router(config) #show ip pim rp-set

Group address    Static-RP-address  Override
--------------------------------------------------
Access-List 44    99.99.99.5           On

Number of group prefixes Learnt from BSR: 1

Group prefix = 224.0.0.0/4 # RPs: 1
      RP 1: 43.43.43.1 priority=0 age=0
```

In the example above, the display shows the following information:

- The Group Address table shows the static RP address that is covered by the access list, and whether or not the override parameter has been enabled.
- The Group prefix line shows the multicast group prefix for the static RP.
- The RP # line shows the configured IP address of the RP candidate.

The **show ip pim rp-map** to show the group-to-RP mapping.

```
FastIron SuperX Router(config)# show ip pim rp-map

Number of group-to-RP mappings: 6

  Group address  RP address
-------------------------------
1 239.255.163.1  43.43.43.1
2 239.255.163.2  43.43.43.1
3 239.255.163.3  43.43.43.1
4 239.255.162.1  99.99.99.5
5 239.255.162.2  99.99.99.5
6 239.255.162.3  99.99.99.5
```

The display shows the multicast group addresses covered by the RP candidate and the IP address of the RP for the listed multicast group. In the example above, you see the following:

- The first three lines show the multicast group addresses that are covered by the RP candidate.
- The last three lines show the multicast group addresses covered by the static RP.

Using ACLs to Limit PIM RP Candidate Advertisement

You can use standard ACLs to control the groups for which the candidate RP will send advertisement messages to the bootstrap router. For example, ACL 5 can be configured to be applied to the multicast groups within the IP address 239.x.x.x range. You can configure the Layer 3 Switch to advertise itself as a candidate RP to the bootstrap router only for groups in the range of 239.x.x.x. Enter commands such as the following:

```
FastIron SuperX Router(config)# interface ethernet 1/1
FastIron SuperX Router(config-if-1/1)# ip address 99.99.99.5 255.255.255.0
```



```
FastIron SuperX Router(config-if-1/1)# ip pim-sparse
FastIron SuperX Router(config-if-1/1)# exit

FastIron SuperX Router(config)# access-list 5 deny host 239.255.162.2
FastIron SuperX Router(config)# access-list 5 permit 239.0.0.0 0.0.255.255

FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# bsr-candidate ethernet 1/1 32 100
FastIron SuperX Router(config-pim-router)# rp-candidate ethernet 1/1 group-list 5
```

The example above shows a configuration for an Ethernet interface. To configure ACLs that are applied to a virtual routing interface, enter commands such as the following:

```
FastIron SuperX Router(config)# interface ve 16
FastIron SuperX Router(config-vif-16)# ip address 16.16.16.1 255.255.255.0
FastIron SuperX Router(config-vif-16)# ip pim-sparse
FastIron SuperX Router(config-vif-16)# exit

FastIron SuperX Router(config)# access-list 5 deny host 239.255.162.2
FastIron SuperX Router(config)# access-list 5 permit 239.255.0.0 0.0.255.255

FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# bsr-candidate ve 16 32 100
FastIron SuperX Router(config-pim-router)# rp-candidate ve 16 group-list 5
```

To configure ACLs that are applied to a loopback interface, enter commands such as the following:

```
FastIron SuperX Router(config)# interface loopback 1
FastIron SuperX Router(config-lbif-1)# ip address 88.88.88.8 255.255.255.0
FastIron SuperX Router(config-lbif-1)# ip pim-sparse
FastIron SuperX Router(config-lbif-1)# exit

FastIron SuperX Router(config)# access-list 5 deny host 239.255.162.2
FastIron SuperX Router(config)# access-list 5 permit 239.255.0.0 0.0.255.255

FastIron SuperX Router(config)# router pim
FastIron SuperX Router(config-pim-router)# bsr-candidate loopback 1 32 100
FastIron SuperX Router(config-pim-router)# rp-candidate loopback 1 group-list 5
```

Syntax: [no] rp-candidate ethernet [<slotnum>/]<portnum> | loopback <num> | ve <num> [group-list <access-list-num>]

The <slotnum> parameter is required on chassis devices.

The <portnum> | **loopback** <num> | **ve** <num> parameter specifies the interface. The Layer 3 Switch will advertise the specified interface's IP address as a candidate RP.

- Enter **ethernet** [<slotnum>/]<portnum> for a physical interface (port).
- Enter **ve** <num> for a virtual interface.
- Enter **loopback** <num> for a loopback interface.

The **group-list** <access-list-num> indicates that a standard ACL is used to filter for which multicast group the advertisement will be made.

NOTE: Extended ACLs cannot be used for group-list.

Using ACLs to Control Multicast Traffic Boundaries

You can create ACLs that determine which multicast traffic packets can be forwarded on an interface in a PIM or DVMRP domain. The ACLs can be create to be applied to a range of multicast group addresses. If an ACL denies the specified multicast group addresses, incoming or outgoing packets from those addresses will not be allowed to flow across the interface.

For example, to set up a boundary, which will deny all multicast group addresses within the 239.x.x.x IP address range, enter commands such as the following:

```
FastIron SuperX Router(config)# access-list 1 deny 239.0.0.0 0.255.255.255
FastIron SuperX Router (config)# access-list 1 permit 234.0.0.0 0.255.255.255

FastIron SuperX Router(config)# interface ethernet 1/1
FastIron SuperX Router(config-if-1/1)# ip pim-sparse
FastIron SuperX Router(config-if-1/1)# ip multicast boundary 1
```

Syntax: [no] ip multicast boundary <access-list-num>

The <access-list-num> parameter defines the ACLs used to set-up the boundaries for multicast traffic packets.

NOTE: Extended ACLs cannot be used in this feature.

Configuring a Static Multicast Route

Static multicast routes allow you to control the network path used by multicast traffic. Static multicast routes are especially useful when the unicast and multicast topologies of a network are different. You can avoid the need to make the topologies similar by instead configuring static multicast routes.

NOTE: This feature is not supported for DVMRP.

You can configure more than one static multicast route. The Layer 3 Switch always uses the most specific route that matches a multicast source address. Thus, if you want to configure a multicast static route for a specific multicast source and also configure another multicast static route for all other sources, you can configure two static routes as shown in the examples below.

To add static routes to multicast router A (see Figure 19.7), enter commands such as the following:

```
PIMRouterA(config)# ip mroute 1 207.95.10.0 255.255.255.0 interface ethernet 1/2
distance 1
PIMRouterA(config)# ip mroute 2 0.0.0.0 0.0.0.0 interface ethernet 2/3 distance 1
PIMRouterA(config)# write memory
```

Syntax: mroute <route-num> <ip-addr> interface ethernet [<slotnum>/]<portnum> | ve <num> [distance <num>]

Or

Syntax: mroute <route-num> <ip-addr> rpf_address <rpf-num>

The <route-num> parameter specifies the route number.

The <ip-addr> command specifies the PIM source for the route.

NOTE: In IP multicasting, a route is handled in terms of its source, rather than its destination.

You can use the **ethernet** [<slotnum>/]<portnum> parameter to specify a physical port or the **ve** <num> parameter to specify a virtual interface.

NOTE: The **ethernet** [<slotnum>/]<portnum> parameter does not apply to PIM SM.

The **distance** <num> parameter sets the administrative distance for the route. When comparing multiple paths for a route, the Layer 3 Switch prefers the path with the lower administrative distance.

NOTE: Regardless of the administrative distances, the Layer 3 Switch always prefers directly connected routes over other routes.

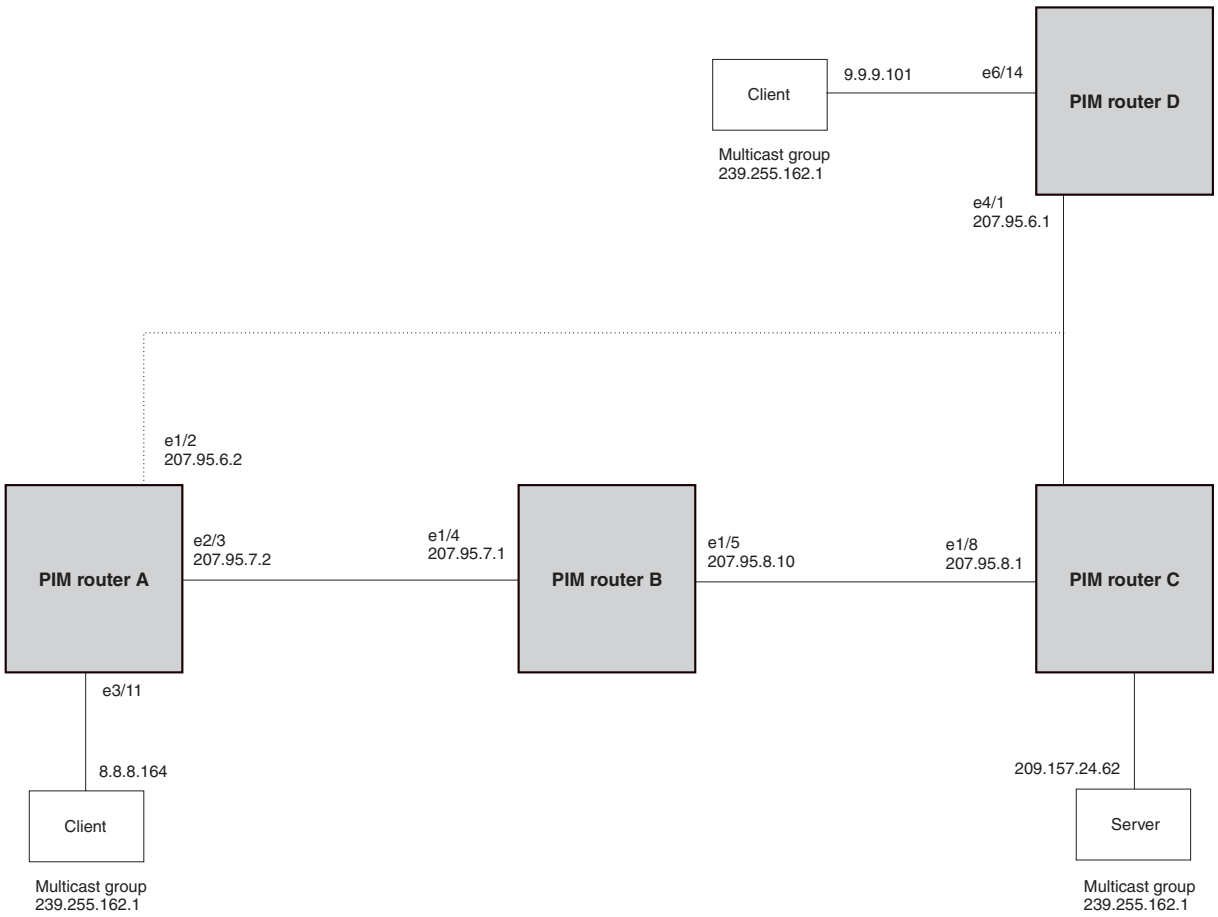
The **rpf_address** <rpf-num> parameter specifies an RPF number.

The example above configures two static multicast routes. The first route is for a specific source network, 207.95.10.0/24. If the Layer 3 Switch receives multicast traffic for network 207.95.10.0/24, the traffic must arrive on port 1/2. The second route is for all other multicast traffic. Traffic from multicast sources other than 207.95.10.0/24 must arrive on port 2/3.

Figure 19.7 shows an example of an IP Multicast network. The two static routes configured in the example above apply to this network. The commands in the example above configure PIM router A to accept PIM packets from 207.95.10.0/24 when they use the path that arrives at port 1/2, and accept all other PIM packets only when they use the path that arrives at port 2/3.

The distance parameter sets the administrative distance. This parameter is used by the software to determine the best path for the route. Thus, to ensure that the Layer 3 Switch uses the default static route, assign a low administrative distance value. When comparing multiple paths for a route, the Layer 3 Switch prefers the path with the lower administrative distance.

Figure 19.7 Example multicast static routes



To add a static route to a virtual interface, enter commands such as the following:

```
FastIron SuperX Router(config)# mroute 3 0.0.0.0 0.0.0.0 int ve 1 distance 1
FastIron SuperX Router(config)# write memory
```

Tracing a Multicast Route

The Foundry implementation of Mtrace is based on “A ‘traceroute’ facility for IP Multicast”, an Internet draft by S. Casner and B. Fenner. To trace a PIM route, use the following CLI method.

NOTE: This feature is not supported for DVMRP.

To trace a PIM route to PIM source 209.157.24.62 in group 239.255.162.1, enter a command such as the following:

```
FastIron SuperX Router# mtrace source 209.157.24.62 group 239.255.162.1

Type Control-c to abort
Tracing the route for tree 209.157.23.188

  0  207.95.7.2
  0  207.95.7.2 Thresh 0
  1  207.95.7.1 Thresh 0
  2  207.95.8.1 Thresh 0
  3  207.157.24.62
```

Syntax: mtrace source <ip-addr> group <multicast-group>

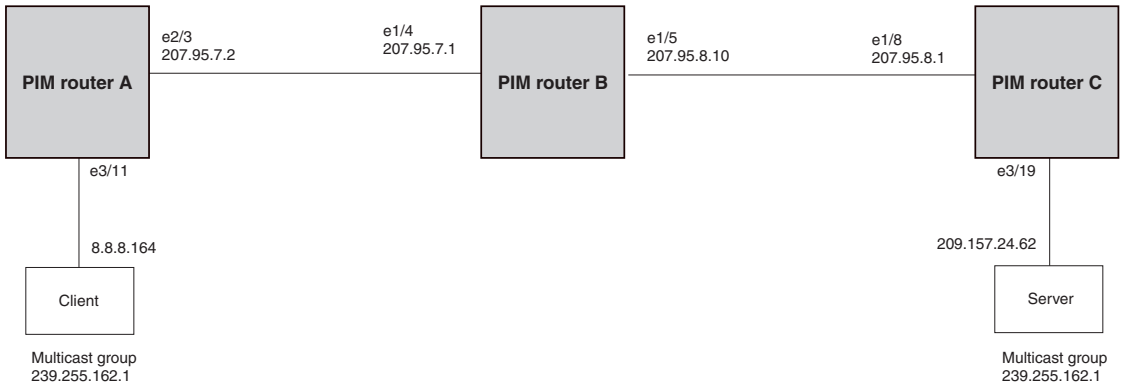
The **source** <ip-addr> parameter specifies the address of the route's source.

NOTE: In IP multicasting, a route is handled in terms of its source, rather than its destination. When you trace an IP route, you specify its destination, but when you trace a PIM route, you specify its source.

The **group** <multicast-group> parameter specifies the PIM group the source IP address is in.

Figure 19.8 shows an example of an IP multicast group. The command example shown above is entered on PIM router A.

Figure 19.8 Example PIM Group



The command example above indicates that the source address 209.157.24.62 is three hops (three PIM routers) away from PIM router A. In PIM terms, each of the three routers has a forwarding state for the specified source address and multicast group. The value following "Thresh" in some of the lines indicates the TTL threshold. The threshold 0 means that all multicast packets are forwarded on the interface. If an administrator has set the TTL threshold to a higher value, only packets whose TTL is higher than the threshold are forwarded on the interface. The threshold is listed only for the PIM router hops between the source and destination.

Displaying Another Multicast Router's Multicast Configuration

The Foundry implementation of Mrinfo is based on the DVMRP Internet draft by T. Pusateri, but applies to PIM and not to DVMRP. To display the PIM configuration of another PIM router, use the following CLI method.

NOTE: This feature is not supported for DVMRP.

To display another PIM router's PIM configuration, enter a command such as the following:

```
FastIron SuperX Router# mrinfo 207.95.8.1
207.95.8.1 -> 207.95.8.10 [PIM/0 /1 ]
207.95.10.2 -> 0.0.0.0 [PIM/0 /1 /leaf]
209.157.25.1 -> 0.0.0.0 [PIM/0 /1 /leaf]
209.157.24.1 -> 0.0.0.0 [PIM/0 /1 /leaf]
207.95.6.1 -> 0.0.0.0 [PIM/0 /1 /leaf]
128.2.0.1 -> 0.0.0.0 [PIM/0 /1 /leaf]
```

Syntax: mrinfo <ip-addr>

The <ip-addr> parameter specifies the IP address of the PIM router.

The output in this example is based on the PIM group shown in Figure 19.8 on page 19-44. The output shows the PIM interfaces configured on PIM router C (207.95.8.1). In this example, the PIM router has six PIM interfaces. One of the interfaces goes to PIM router B. The other interfaces go to leaf nodes, which are multicast end nodes attached to the router's PIM interfaces. (For simplicity, the figure shows only one leaf node.)

When the arrow following an interface in the display points to a router address, this is the address of the next hop PIM router on that interface. In this example, PIM interface 207.95.8.1 on PIM router 207.95.8.1 is connected to PIM router 207.95.8.10. The connection can be a direct one or can take place through non-PIM routers. In this example, the PIM routers are directly connected.

When the arrow following an interface address points to zeros (0.0.0.0), the interface is not connected to a PIM router. The interface is instead connected to a leaf node.

NOTE: This display shows the PIM interface configuration information, but does not show the link states for the interfaces.

The information in brackets indicates the following:

- The multicast interface type (always PIM; this display is not supported for DVMRP)
- The Time-to-Live (TTL) for the interface.
- The metric for the interface
- Whether the interface is connected to a leaf node ("leaf" indicates a leaf node and blank indicates another PIM router)

For example, the information for the first interface listed in the display is "PIM/0 /1". This information indicates that the interface is a PIM interface, has a TTL of 0, and a metric of 1. The interface is not a leaf node interface and thus is an interface to another PIM router.

The information for the second interface in the display is "PIM/0 /1/leaf". This information indicates that the interface is a PIM interface, has a TTL of 0 and a metric of 1, and is connected to a leaf node.

IGMP V3

NOTE: IGMP V3 is supported on Layer 3 devices running software release 02.4.00 or later.

The Internet Group Management Protocol (IGMP) allows an IPV4 interface to communicate IP Multicast group membership information to its neighboring routers. The routers in turn limit the multicast of IP packets with multicast destination addresses to only those interfaces on the router that are identified as IP Multicast group members. This release introduces the support of IGMP version 3 (IGMP V3) on Layer 3 Switches.

In IGMP V2, when a router sent a query to the interfaces, the clients on the interfaces respond with a membership report of multicast groups to the router. The router can then send traffic to these groups, regardless of the traffic source. When an interface no longer needs to receive traffic from a group, it sends a leave message to the router which in turn sends a group-specific query to that interface to see if any other clients on the same interface is still active.

In contrast, IGMP V3 provides selective filtering of traffic based on traffic source. A router running IGMP V3 sends queries to every multicast enabled interface at the specified interval. These queries determine if any interface wants to receive traffic from the router. The queries include the IP address of the traffic source (S) and/or the ID of the multicast group (G).

The interfaces respond to these queries by sending a membership report that contains one or more of the following records that are associated with a specific group:

- Current-State Record that indicates from which sources the interface wants to receive and not receive traffic. The record contains source address of interfaces and whether or not traffic will be received or included (IS_IN) or not received or excluded (IS_EX) from that source.
- Filter-mode-change record. If the interface changes its current state from IS_IN to IS_EX, a TO_EX record is included in the membership report. Likewise, if an interface's current state changes from IS_EX to IS_IN, a TO_IN record appears in the membership report.

  IGMP V2 Leave report is equivalent to a TO_IN(empty) record in IGMP V3. This record means that no traffic from this group will be received regardless of the source.

  An IGMP V2 group report is equivalent to an IS_EX(empty) record in IGMP V3. This record means that all traffic from this group will be received regardless of source.
- Source-List-Change Record. If the interface wants to add or remove traffic sources from its membership report, the membership report can have an ALLOW record, which contains a list of new sources from which the interface wishes to receive traffic. It can also contains a BLOCK record, which lists current traffic sources from which the interfaces wants to stop receiving traffic.

In response to membership reports from the interfaces, the router sends a Group-Specific or a Group-and-Source Specific query to the multicast interfaces. Each query is sent three times with a one-second interval in between each transmission to ensure the interfaces receive the query. For example, a router receives a membership report with a Source-List-Change record to block old sources from an interface. The router sends Group-and-Source Specific Queries to the source and group (S,G) identified in the record. If none of the interfaces is interested in the (S,G), it is removed from (S,G) list for that interface on the router.

Each IGMP V3-enabled router maintains a record of the state of each group and each physical port within a virtual routing interface. This record contains the group, group-timer, filter mode, and source records information for the group or interface. Source records contain information on the source address of the packet and source timer. If the source timer expires when the state of the group or interface is in Include mode, the record is removed.

Default IGMP Version

IGMP V3 is available on devices running software release 02.4.00 and later; however, Foundry devices are shipped with IGMP V2-enabled. You must enable IGMP V3 globally or per interface.

Also, you must specify what version of IGMP you want to run on a device globally, on each interface (physical port or virtual routing interface), and on each physical port within a virtual routing interface. If you do not specify an IGMP version, IGMP V2 will be used.

Compatibility with IGMP V1 and V2

Different multicast groups, interfaces, and routers can run their own version of IGMP. Their version of IGMP is reflected in the membership reports that the interfaces send to the router. Routers and interfaces must be configured to recognized the version of IGMP you want them to process.

An interface or router sends the queries and reports that include its IGMP version specified on it. It may recognize a query or report that has a different version, but it may not process them. For example, an interface running IGMP V2 can recognize IGMP V3 packets, but cannot process them. Also, a router running IGMP V3 can recognize and process IGMP V2 packet, but when that router sends queries to an IGMP V2 interface, the host on that interface may not recognize the IGMP V3 queries. The interface or router does not automatically downgrade the IGMP version running on them to avoid version deadlock.

If an interface continuously receives queries from routers that are running versions of IGMP that are different from what is on the interface, the interface logs warning messages in the syslog every five minutes. Reports sent by interfaces to routers that contain different versions of IGMP do not trigger warning messages; however, you can see the versions of the packets using the **show ip igmp traffic** command.

The version of IGMP can be specified globally, per interface (physical port or virtual routing interface), and per physical port within a virtual routing interface. The IGMP version set on a physical port within a virtual routing interface supersedes the version set on a physical or virtual routing interface. Likewise, the version on a physical or virtual routing interface supersedes the version set globally on the device. The sections below present how to set the version of IGMP.

Globally Enabling the IGMP Version

Using the CLI

To globally identify the IGMP version on a Foundry device, enter the following command:

FESX424 Router`(config)# ip igmp version 3`

Syntax: ip igmp version <version-number>

Enter 1, 2, or 3 for <version-number>. Version 2 is the default version.

Using the Web Management Interface

You cannot set the IGMP version using the Web management interface.

Enabling the IGMP Version Per Interface Setting

Using the CLI

To specify the IGMP version for a physical port, enter a command such as the following:

```
FESX424 Router(config)# interface eth 1/5
FESX424 Router(config-if-1/5)# ip igmp version 3
```

To specify the IGMP version for a virtual routing interface on a physical port, enter a command such as the following:

```
FESX424 Router(config)# interface ve 3
FESX424 Router(config-vif-1) ip igmp version 3
```

Syntax: [no] ip igmp version <version-number>

Enter 1, 2, or 3 for <version-number>. Version 2 is the default version.

Using the Web Management Interface

You cannot set the IGMP version using the Web management interface.

Enabling the IGMP Version on a Physical Port Within a Virtual Routing Interface

Using the CLI

To specify the IGMP version recognized by a physical port that is a member of a virtual routing interface, enter a command such as the following:

```
FESX424 Router(config)# interface ve 3
FESX424 Router(config-vif-3)# ip igmp version 2
FESX424 Router(config-vif-3)# ip igmp port-version 3 e1/3-e1/7 e2/9
```

In this example, the second line sets IGMP V2 on virtual routing interface 3. However, the third line set IGMP V3 on ports 1/3 through 1/7 and port e2/9. All other ports in this virtual routing interface are configured with IGMP V2.

Syntax: ip igmp port-version <version-number> ethernet <port-number>

Enter 1, 2, or 3 for <version-number>. IGMP V2 is the default version.

The **ethernet** <port-number> parameter specifies which physical port within a virtual routing interface is being configured.

Using the Web Management Interface

You cannot set the IGMP version using the Web management interface.

Enabling Membership Tracking and Fast Leave

IGMP V3 provides membership tracking and fast leave to clients. In IGMP V2, only one client on an interface needs to respond to a router's queries; therefore, some of the clients may be invisible to the router, making it impossible for the router to track the membership of all clients in a group. Also, when a client leaves the group, the router sends group specific queries to the interface to see if other clients on that interface need the data stream of the client who is leaving. If no client responds, the router waits three seconds before it stops the traffic.

IGMP V3 contains the tracking and fast leave feature that you enable on virtual routing interfaces. Once enabled, all physical ports on that virtual routing interface will have the feature enabled. IGMP V3 requires all clients to respond to general and group specific queries so that all clients on an interface can be *tracked. Fast leave* allows clients to leave the group without the three second waiting period, if the following conditions are met:

- If the interface, to which the client belongs, has IGMP V3 clients only. Therefore, all physical ports on a virtual routing interface must have IGMP V3 enabled and no IGMP V1 or V2 clients can be on the interface. (Although IGMP V3 can handle V1 and V2 clients, these two clients cannot be on the interface in order for fast leave to take effect.)
- No other client on the interface is receiving traffic from the group to which the client belongs.

  Every group on the physical interface of a virtual routing interface keeps its own tracking record. However, it can track group membership only; it cannot track by (source, group).

For example, two clients (Client A and Client B) belong to group1 but each is receiving traffic streams from different sources. Client A receives a stream from (source_1, group1) and Client B receives it from (source_2, group1). The router still waits for three seconds before it stops the traffic because the two clients are in the same group. If the clients are in different groups, then the three second waiting period is not applied and traffic is stopped immediately. The **show ip igmp group tracking** command displays that clients in a group that are being tracked.

If a client sends a leave message, the client is immediately removed from the group. If a client does not send a report during the the specified group membership time (the default is 140 seconds), that client is removed from the tracking list.

USING THE CLI

To enable the tracking and fast leave feature, enter commands such as the following:

```
FESX424 Router(config)# interface ve 13
FESX424 Router(config-vif-13)# ip igmp tracking
```

Syntax: ip igmp tracking

USING THE WEB MANAGEMENT INTERFACE

You cannot change this parameter using the Web management interface.

Setting the Query Interval

The IGMP query interval period defines how often a router will query an interface for group membership. Possible values are 10 – 3,600 seconds and the default value is 60 seconds, but the value you enter must be a little more than twice the group membership time.

USING THE CLI

To modify the default value for the IGMP query interval, enter the following:

```
FESX424 Router(config)# ip igmp query-interval 120
```

Syntax: ip igmp query-interval <10-3600>

The interval must be a little more than two times the group membership time.

USING THE WEB MANAGEMENT INTERFACE

If available, you can use the Web management interface to configure query interval. For example, on BigIron Chassis devices, log in to the Web management interface and go to the Configure -> DVMRP -> IGMP panel. Enter a value from 10 – 3600 in the Query Interval field. Refer to the *Foundry Enterprise Configuration and Management Guide* for details.

Setting the Group Membership Time

Group membership time defines how long a group will remain active on an interface in the absence of a group report. Possible values are from 20 – 7200 seconds and the default value is 140 seconds.

USING THE CLI

To define an IGMP membership time of 240 seconds, enter the following:

```
FESX424 Router(config)# ip igmp group-membership-time 240
```

Syntax: ip igmp group-membership-time <20-7200>

USING THE WEB MANAGEMENT INTERFACE

If available, you can use the Web management interface to configure group membership time. For example, on BigIron Chassis devices, log in to the Web management interface and go to the Configure -> DVMRP -> IGMP panel. Enter a value from 20 – 7200 in the Group Membership Time field. Refer to the *Foundry Enterprise Configuration and Management Guide* for details.

Setting the Maximum Response Time

The maximum response time defines the maximum number of seconds that a client can wait before it replies to the query sent by the router. Possible values are 1 – 10. The default is 5.

USING THE CLI

To change the IGMP maximum response time, enter a command such as the following at the global CONFIG level of the CLI:

```
FESX424 Router(config)# ip igmp max-response-time 8
```

Syntax: [no] ip igmp max-response-time <num>

The <num> parameter specifies the maximum number of seconds for the response time. Enter a value from 1 – 10. The default is 5.

USING THE WEB MANAGEMENT INTERFACE

You cannot change this parameter using the Web management interface.

IGMP V3 and Source Specific Multicast Protocols

Enabling IGMP V3 enables source specific multicast (SSM) filtering for DVMRP and PIM Dense (PIM-DM) for multicast group addresses in the 224.0.1.0 through 239.255.255.255 address range. However, if PIM Sparse is used as the multicast protocol, the SSM protocol should be enabled if you want to filter unwanted traffic before the Shortest Path Tree protocol switchover occurs for groups in the 232/8 range. Not configuring the SSM protocol in PIM Sparse may cause the switch or router to leak unwanted packets with the same group, but containing undesired sources, to clients. After SPT switch over, the leak stops and source specific multicast works correctly even without configuring the SSM protocol.

If the SSM protocol is not enabled and before the SPT switchover, the multicast router creates one (*, G) entry for the entire multicast group, which can have many sources. If the SSM protocol is enabled, one (S,G) entry is created for every member of the multicast group, even for members with non-existent traffic. For example, if there are 1,000 members in the group, 1,000 (S,G) entries will be created. Therefore, enabling the SSM protocol for PIM-SM requires more resources than leaving the protocol disabled.

Enabling SSM

To enable the SSM protocol on a Foundry device running PIM-SM, enter a command such as the following:

```
FESX424 Router(config)# router pim
FESX424 Router(config-pim-router)# ssm-enable
```

Syntax: [no] ssm-enable

Enter the ssm-enable command under the router pim level to globally enable the SSM protocol on a Layer 3 Switch.

Displaying IGMP V3 Statistics

The sections below present the show commands available for IGMP V3.

Displaying IGMP Group Status

NOTE: This report is available on Layer 3 Switches.

You can display the status of all IGMP multicast groups on a device by entering the following command:

```
FESX424 Router(config)# show ip igmp group
Interface v18 : 1 groups
      group            phy-port static querier life mode    #_src
1     239.0.0.1        e4/20    no     yes           include 19
Interface v110 : 3 groups
      group            phy-port static querier life mode    #_src
2     239.0.0.1        e4/5     no     yes           include 10
3     239.0.0.1        e4/6     no     yes     100   exclude 13
4     224.1.10.1       e4/5     no     yes           include 1
```


To display the status of one IGMP multicast group, enter a command such as the following:

```
FESX424 Router(config)# show ip igmp group 239.0.0.1 detail
Display group 239.0.0.1 in all interfaces.
Interface v18 : 1 groups
      group            phy-port static querier life mode    #_src
1     239.0.0.1        e4/20    no     yes          include 19
    group: 239.0.0.1, include, permit 19 (source, life):
      (3.3.3.1 40) (3.3.3.2 40) (3.3.3.3 40) (3.3.3.4 40) (3.3.3.5 40)
      (3.3.3.6 40) (3.3.3.7 40) (3.3.3.8 40) (3.3.3.9 40) (3.3.3.10 40)
      (3.3.3.11 40) (3.3.3.12 40) (3.3.3.13 40) (3.3.3.14 40) (3.3.3.15 40)
      (3.3.3.16 40) (3.3.3.17 40) (3.3.3.18 40) (3.3.3.19 40)
Interface v110 : 1 groups
      group            phy-port static querier life mode    #_src
2     239.0.0.1        e4/5     no     yes          include 10
    group: 239.0.0.1, include, permit 10 (source, life):
      (2.2.3.0 80) (2.2.3.1 80) (2.2.3.2 80) (2.2.3.3 80) (2.2.3.4 80)
      (2.2.3.5 80) (2.2.3.6 80) (2.2.3.7 80) (2.2.3.8 80) (2.2.3.9 80)
```

If the tracking and fast leave feature is enabled, you can display the list of clients that belong to a particular group by entering commands such as the following:

```
FESX424 Router(config)# show ip igmp group 224.1.10.1 tracking
Display group 224.1.10.1 in all interfaces with tracking enabled.
Interface v13 : 1 groups, tracking_enabled
      group            phy-port static querier life mode    #_src
1     224.1.10.1       e4/15    no     yes          include 3
    receive reports from 3 clients:
      110.110.110.7 110.110.110.8 110.110.110.9
```

Syntax: show ip igmp group [<group-address> [detail] [tracking]]

If you want a report for a specific multicast group, enter that group's address for <group-address>. Omit the <group-address> if you want a report for all multicast groups.

Enter **detail** if you want to display the source list of the multicast group.

Enter **tracking** if you want information on interfaces that have tracking enabled.

IGMP V2 and V3 statistics displayed on the report for each interface.

This Field	Displays
Group	The address of the multicast group
Phy-port	The physical port on which the multicast group was received.
Static	A "yes" entry in this column indicates that the multicast group was configured as a static group; "No" means it was not. Static multicast groups can be configured in IGMP V2 using the **ip igmp static** command. In IGMP V3, static sources cannot be configured in static groups.
Querier	"Yes" means that the port is a querier port; "No" means it is not. A port becomes a non-querier port when it receives a query from a source with a lower source IP address than the port.

This Field	Displays
Life	Shows the number of seconds the interface can remain in exclude mode. An exclude mode changes to include mode if it does not receive an "IS_EX" or "TO_EX" message during a certain period of time. The default is 140 seconds. There is no "life" displayed in include mode.
Mode	Indicates current mode of the interface: Include or Exclude. If the interface is in Include mode, it admits traffic only from the source list. If an interface is in Exclude mode, it denies traffic from the source list and accepts the rest.
#_src	Identifies the source list that will be included or excluded on the interface. If IGMP V2 group is in Exclude mode with a #_src of 0, the group excludes traffic from 0 (zero) source list, which means that all traffic sources are included.
Group:	If you requested a *detailed* report, the following information is displayed: • The multicast group address • The mode of the group • A list of sources from which traffic will be admitted (include) or denied (exclude) on the interface is listed. • The life of each source list. If you requested a *tracking* report, the clients from which reports were received are identified.

Displaying the IGMP Status of an Interface

NOTE: This report is available on Layer 3 Switches.

You can display the status of a multicast enabled port by entering a command such as the following:

```
FESX424 Router(config)# show ip igmp interface
query interval = 60, max response time= 3, group membership time=140
v5: default V2,          PIM dense, addr=1.1.1.2
  e4/12   has    0 groups, non-Querier (age=40), default V2
v18: default V2,         DVMRP, addr=2.2.2.1
  e4/20   has    0 groups, Querier, default V2
v20: configured V3,      PIM dense (port down), addr=1.1.20.1
v110: configured V3,     PIM dense, addr=110.110.110.1
  e4/6    has    2 groups, Querier, default V3
    group: 239.0.0.1, exclude, life=100, deny 13
    group: 224.1.10.1, include, permit 2
  e4/5    has    3 groups, Querier, default V3
    group: 224.2.2.2, include, permit 100
    group: 239.0.0.1, include, permit 10
    group: 224.1.10.1, include, permit 1
```

Syntax: show ip igmp interface [ve | ethernet <number> <group-address>]

Enter **ve** and its <number> or **ethernet** and its <number> to display information for a specific virtual routing interface or ethernet interface.

Entering an address for <group-address> displays information for a specified group on the specified interface.

The report shows the following information:

This Field	Displays
Query interval	Displays how often a querier sends a general query on the interface.
Max response	The maximum number of seconds a client can wait before it replies to the query.
Group membership time	The number of seconds multicast groups can be members of this group before aging out.
(details)	The following is displayed for each interface: • The ID of the interface • The IGMP version that it is running (default IGMP V2 or configured IGMP V3) • The multicast protocol it is running: DVMRP, PIM-DM, PIM-SM • Address of the multicast group on the interface • If the interface is a virtual routing interface, the physical port to which that interface belongs, the number of groups on that physical port, whether or not the port is a querier or a non-querier port, the age of the port, and other multicast information for the port are displayed.

Displaying IGMP Traffic Status

NOTE: This report is available on Layer 3 Switches.

To display the traffic status on each virtual routing interface, enter the following command:

```
FESX424 Router(config)# show ip igmp traffic
Recv  QryV2 QryV3 G-Qry GSQry MbrV2 MbrV3 Leave  IsIN  IsEX ToIN ToEX ALLOW BLK
v5       29     0     0     0     0     0     0     0     0    0    0     0    0
v18      15     0     0     0     0    30     0    60     0    0    0     0    0
v110      0     0     0     0     0    97     0   142    37    2    2     3    2
Send  QryV1 QryV2 QryV3 G-Qry GSQry
v5        0     2     0     0     0
v18       0     0    30    30     0
v110      0     0    30    44    11
```

Syntax: show ip igmp traffic

The report shows the following information:

This Field	Displays
QryV2	Number of general IGMP V2 query received or sent by the virtual routing interface.
QryV3	Number of general IGMP V3 query received or sent by the virtual routing interface.
G-Qry	Number of group specific query received or sent by the virtual routing interface.
GSQry	Number of source specific query received or sent by the virtual routing interface.
MbrV2	The IGMP V2 membership report.
MbrV3	The IGMP V3 membership report.
Leave	Number of IGMP V2 “leave” messages on the interface. (See ToEx for IGMP V3.)
IsIN	Number of source addresses that were included in the traffic.
IsEX	Number of source addresses that were excluded in the traffic.
ToIN	Number of times the interface mode changed from exclude to include.
ToEX	Number of times the interface mode changed from include to exclude.
ALLOW	Number of times that additional source addresses were allowed or denied on the interface:
BLK	Number of times that sources were removed from an interface.

Clearing IGMP Statistics

To clear statistics for IGMP traffic, enter the following command:

```
FESX424 Router# clear igmp traffic
```

Syntax: clear igmp traffic

This command clears all the multicast traffic information on all interfaces on the device.

Chapter 20
Configuring OSPF

This chapter describes how to configure OSPF on Foundry Layer 3 Switches using the CLI.

This chapter contains the following information:

Table 20.1: Chapter Contents

For complete syntax information for the CLI commands shown in this chapter, see the *Foundry Switch and Router Command Line Interface Reference*.

NOTE: If you need to increase the capacity of the IP route table, see "Displaying and Modifying System Parameter Default Settings" on page 4-8.

Overview of OSPF

OSPF is a link-state routing protocol. The protocol uses link-state advertisements (LSA) to update neighboring routers regarding its interfaces and information on those interfaces. The router floods these LSAs to all neighboring routers to update them regarding the interfaces. Each router maintains an identical database that describes its area topology to help a router determine the shortest path between it and any neighboring router.

Foundry Layer 3 Switches support the following types of LSAs, which are described in RFC 1583:

- Router link
- Network link
- Summary link
- Autonomous system (AS) summary link
- AS external link

- Not-So-Stubby Area (NSSA) external link

OSPF is built upon a hierarchy of network components. The highest level of the hierarchy is the ***Autonomous System (AS)***. An autonomous system is defined as a number of networks, all of which share the same routing and administration characteristics.

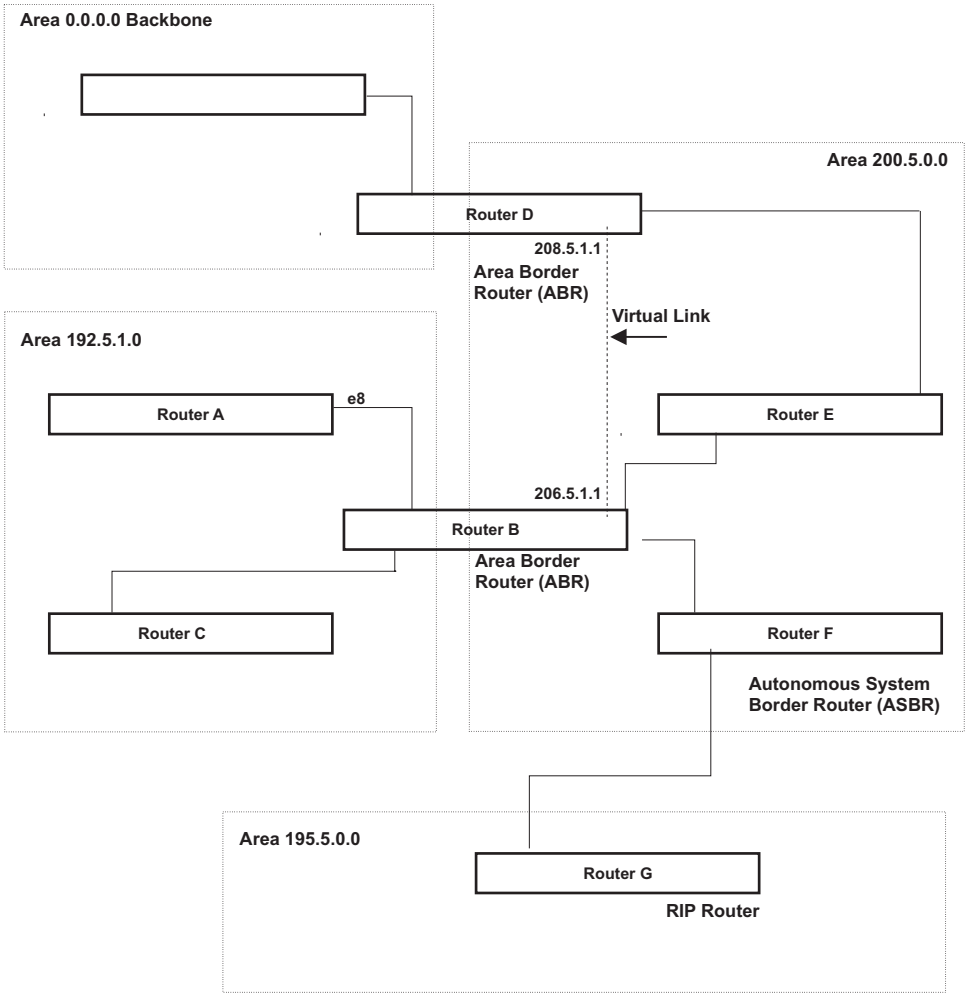
An AS can be divided into multiple ***areas*** as shown in Figure 20.1 on page 20-3. Each area represents a collection of contiguous networks and hosts. Areas limit the area to which link-state advertisements are broadcast, thereby limiting the amount of flooding that occurs within the network. An area is represented in OSPF by either an IP address or a number.

You can further limit the broadcast area of flooding by defining an area range. The area range allows you to assign an aggregate value to a range of IP addresses. This aggregate value becomes the address that is advertised instead all of the individual addresses it represents being advertised. You can assign up to 32 ranges in an OSPF area.

An OSPF router can be a member of multiple areas. Routers with membership in multiple areas are known as ***Area Border Routers (ABRs)***. Each ABR maintains a separate topological database for each area the router is in. Each topological database contains all of the LSA databases for each router within a given area. The routers within the same area have identical topological databases. The ABR is responsible for forwarding routing information or changes between its border areas.

An ***Autonomous System Boundary Router (ASBR)*** is a router that is running multiple protocols and serves as a gateway to routers outside an area and those operating with different protocols. The ASBR is able to import and translate different protocol routes into OSPF through a process known as ***redistribution***. For more details on redistribution and configuration examples, see “Enable Route Redistribution” on page 20-28.

Figure 20.1 OSPF operating in a network



OSPF Point-to-Point Links

OSPF point-to-point links are supported on Gigabit and 10-Gigabit Ethernet interfaces of FESX devices running software release 02.2.00 or later, and on FSX devices running software release 02.3.01 or later.

One important OSPF process is ***Adjacency***. Adjacency occurs when a relationship is formed between neighboring routers for the purpose of exchanging routing information. Adjacent OSPF neighbor routers go beyond the simple Hello packet exchange; they exchange database information. In order to minimize the amount of information exchanged on a particular segment, one of the first steps in creating adjacency is to assign a Designated Router (DR) and a Backup Designated Router (BDR). The Designated Router ensures that there is a central point of contact, thereby improving convergence time within a multi-access segment.

In an OSPF point-to-point network, where a direct Layer 3 connection exists between a single pair of OSPF routers, there is no need for Designated and Backup Designated Routers, as is the case in OSPF multi-access networks. Without the need for Designated and Backup Designated routers, a point-to-point network establishes adjacency and converges faster. The neighboring routers become adjacent whenever they can communicate directly. In contrast, in broadcast and non-broadcast multi-access (NBMA) networks, the Designated Router and Backup Designated Router become adjacent to all other routers attached to the network.

To configure an OSPF point-to-point link, see “Configuring an OSPF Point-to-Point Link” on page 20-36.

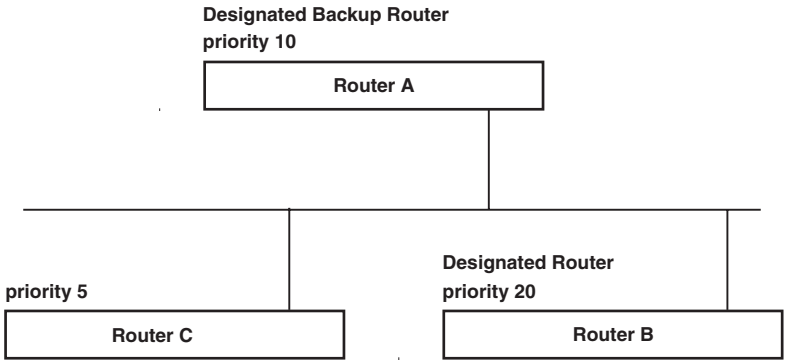
Designated Routers in Multi-Access Networks

In a network that has multiple routers attached, OSPF elects one router to serve as the designated router (DR) and another router on the segment to act as the backup designated router (BDR). This arrangement minimizes the amount of repetitive information that is forwarded on the network by forwarding all messages to the designated router and backup designated routers responsible for forwarding the updates throughout the network.

Designated Router Election in Multi-Access Networks

In a network with no designated router and no backup designated router, the neighboring router with the highest priority is elected as the DR, and the router with the next largest priority is elected as the BDR, as shown in Figure 20.2

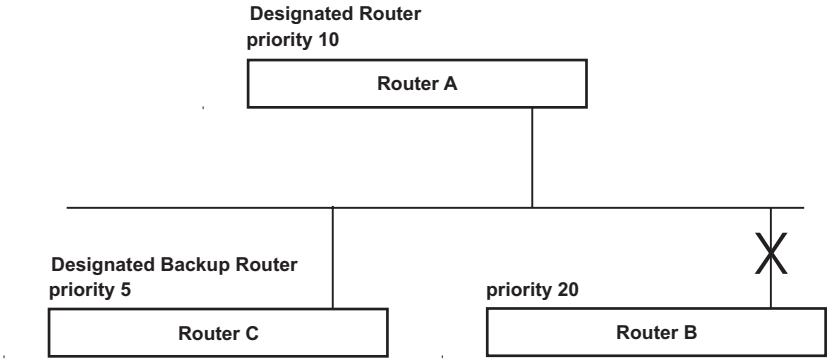
Figure 20.2 Designated and backup router election



If the DR goes off-line, the BDR automatically becomes the DR. The router with the next highest priority becomes the new BDR. This process is shown in Figure 20.3.

NOTE: Priority is a configurable option at the interface level. You can use this parameter to help bias one router as the DR.

Figure 20.3 Backup designated router becomes designated router



If two neighbors share the same priority, the router with the highest router ID is designated as the DR. The router with the next highest router ID is designated as the BDR.

NOTE: By default, the Foundry router ID is the IP address configured on the lowest numbered loopback interface. If the Layer 3 Switch does not have a loopback interface, the default router ID is the lowest numbered IP address configured on the device. For more information or to change the router ID, see “Changing the Router ID” on page 16-23.

When multiple routers on the same network are declaring themselves as DRs, then both priority and router ID are used to select the designated router and backup designated routers.

When only one router on the network claims the DR role despite neighboring routers with higher priorities or router IDs, this router remains the DR. This is also true for BDRs.

The DR and BDR election process is performed when one of the following events occurs:

- an interface is in a waiting state and the wait time expires
- an interface is in a waiting state and a hello packet is received that addresses the BDR
- a change in the neighbor state occurs, such as:
  - a neighbor state transitions from 2 or higher
  - communication to a neighbor is lost
  - a neighbor declares itself to be the DR or BDR for the first time

OSPF RFC 1583 and 2178 Compliance

Foundry routers are configured, by default, to be compliant with the RFC 1583 OSPF V2 specification. Foundry routers can also be configured to operate with the latest OSPF standard, RFC 2178.

NOTE: For details on how to configure the system to operate with the RFC 2178, see “Modify OSPF Standard Compliance Setting” on page 20-36.

Reduction of Equivalent AS External LSAs

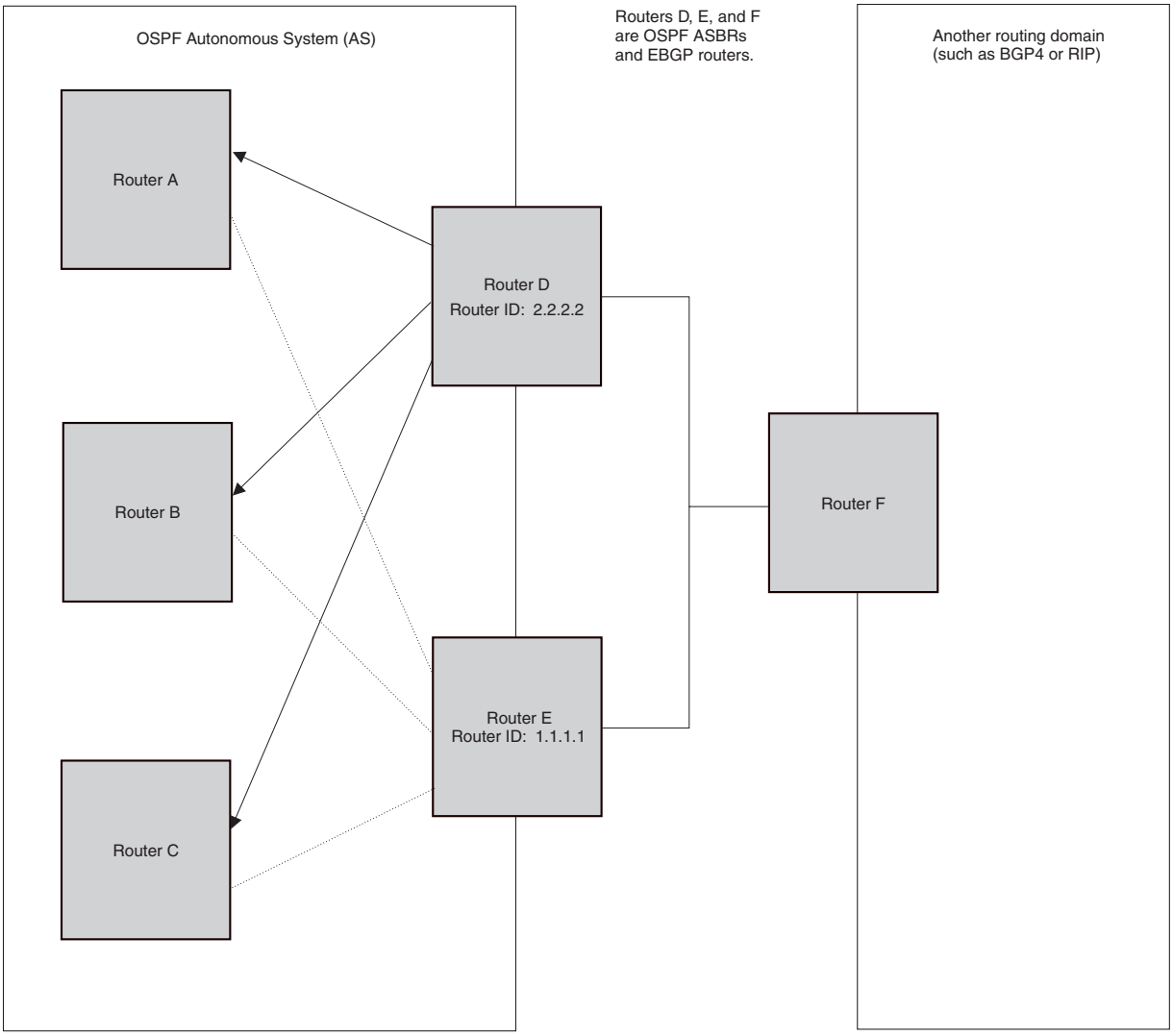
An OSPF ASBR uses AS External link advertisements (AS External LSAs) to originate advertisements of a route to another routing domain, such as a BGP4 or RIP domain. The ASBR advertises the route to the external domain by flooding AS External LSAs to all the other OSPF routers (except those inside stub networks) within the local OSPF Autonomous System (AS).

In some cases, multiple ASBRs in an AS can originate equivalent LSAs. The LSAs are equivalent when they have the same cost, the same next hop, and the same destination. Foundry devices optimize OSPF by eliminating duplicate AS External LSAs in this case. The Layer 3 Switch with the lower router ID flushes the duplicate External LSAs from its database and thus does not flood the duplicate External LSAs into the OSPF AS. AS External LSA reduction therefore reduces the size of the Layer 3 Switch’s link state database.

This enhancement implements the portion of RFC 2328 that describes AS External LSA reduction. This enhancement is enabled by default, requires no configuration, and cannot be disabled.

Figure 20.4 shows an example of the AS External LSA reduction feature. In this example, Foundry Layer 3 Switches D and E are OSPF ASBRs, and thus communicate route information between the OSPF AS, which contains Routers A, B, and C, and another routing domain, which contains Router F. The other routing domain is running another routing protocol, such as BGP4 or RIP. Routers D, E, and F, therefore, are each running both OSPF and either BGP4 or RIP.

Figure 20.4 AS External LSA reduction



Notice that both Router D and Router E have a route to the other routing domain through Router F. In earlier software releases, if Routers D and E have equal-cost routes to Router F, then both Router D and Router E flood AS External LSAs to Routers A, B, and C advertising the route to Router F. Since both routers are flooding equivalent routes, Routers A, B, and C receive multiple routes with the same cost to the same destination (Router F). For Routers A, B, and C, either route to Router F (through Router D or through Router E) is equally good.

OSPF eliminates the duplicate AS External LSAs. When two or more Foundry Layer 3 Switches configured as ASBRs have equal-cost routes to the same next-hop router in an external routing domain, the ASBR with the highest router ID floods the AS External LSAs for the external domain into the OSPF AS, while the other ASBRs flush the equivalent AS External LSAs from their databases. As a result, the overall volume of route advertisement traffic within the AS is reduced and the Layer 3 Switches that flush the duplicate AS External LSAs have more memory for other OSPF data. In Figure 20.4, since Router D has a higher router ID than Router E, Router D floods the AS External LSAs for Router F to Routers A, B, and C. Router E flushes the equivalent AS External LSAs from its database.

Algorithm for AS External LSA Reduction

Figure 20.4 shows an example in which the normal AS External LSA reduction feature is in effect. The behavior changes under the following conditions:

- There is one ASBR advertising (originating) a route to the external destination, but one of the following

happens:

- A second ASBR comes on-line
- A second ASBR that is already on-line begins advertising an equivalent route to the same destination.

In either case above, the router with the higher router ID floods the AS External LSAs and the other router flushes its equivalent AS External LSAs. For example, if Router D is offline, Router E is the only source for a route to the external routing domain. When Router D comes on-line, it takes over flooding of the AS External LSAs to Router F, while Router E flushes its equivalent AS External LSAs to Router F.

- One of the ASBRs starts advertising a route that is no longer equivalent to the route the other ASBR is advertising. In this case, the ASBRs each flood AS External LSAs. Since the LSAs either no longer have the same cost or no longer have the same next-hop router, the LSAs are no longer equivalent, and the LSA reduction feature no longer applies.
- The ASBR with the higher router ID becomes unavailable or is reconfigured so that it is no longer an ASBR. In this case, the other ASBR floods the AS External LSAs. For example, if Router D goes off-line, then Router E starts flooding the AS with AS External LSAs for the route to Router F.

Support for OSPF RFC 2328 Appendix E

Foundry devices provide support for Appendix E in OSPF RFC 2328. Appendix E describes a method to ensure that an OSPF router (such as aFoundry Layer 3 Switch) generates unique link state IDs for type-5 (External) link state advertisements (LSAs) in cases where two networks have the same network address but different network masks.

NOTE: Support for Appendix E of RFC 2328 is enabled automatically and cannot be disabled. No user configuration is required.

Normally, an OSPF router uses the network address alone for the link state ID of the link state advertisement (LSA) for the network. For example, if the router needs to generate an LSA for network 10.1.2.3 255.0.0.0, the router generates ID 10.1.2.3 for the LSA.

However, suppose that an OSPF router needs to generate LSAs for all the following networks:

- 10.0.0.0 255.0.0.0
- 10.0.0.0 255.255.0.0
- 10.0.0.0 255.255.255.0

All three networks have the same network address, 10.0.0.0. Without support for RFC 2328 Appendix E, an OSPF router uses the same link state ID, 10.0.0.0, for the LSAs for all three networks. For example, if the router generates an LSA with ID 10.0.0.0 for network 10.0.0.0 255.0.0.0, this LSA conflicts with the LSA generated for network 10.0.0.0 255.255.0.0 or 10.0.0.0 255.255.255.0. The result is multiple LSAs that have the same ID but that contain different route information.

When appendix E is supported, the router generates the link state ID for a network as follows:

1. Does an LSA with the network address as its ID already exist?
   - No – Use the network address as the ID.
   - Yes – Go to Step 2.
2. Compare the networks that have the same network address, to determine which network is more specific. The more specific network is the one that has more contiguous one bits in its network mask. For example, network 10.0.0.0 255.255.0.0 is more specific than network 10.0.0.0 255.0.0.0, because the first network has 16 ones bits (255.255.0.0) whereas the second network has only 8 ones bits (255.0.0.0).
   - For the less specific network, use the networks address as the ID.
   - For the more specific network, use the network's broadcast address as the ID. The broadcast address is the network address, with all ones bits in the host portion of the address. For example, the broadcast address for network 10.0.0.0 255.255.0.0 is 10.0.0.255.

If this comparison results in a change to the ID of an LSA that has already been generated, the router generates a new LSA to replace the previous one. For example, if the router has already generated an LSA for network with ID 10.0.0.0 for network 10.0.0.0 255.255.255.0, the router must generate a new LSA for the network, if the router needs to generate an LSA for network 10.0.0.0 255.255.0.0 or 10.0.0.0 255.0.0.0.

Dynamic OSPF Activation and Configuration

OSPF is automatically activated when you enable it. The protocol does not require a software reload.

You can configure and save the following OSPF changes without resetting the system:

- all OSPF interface-related parameters (for example: area, hello timer, router dead time cost, priority, re-transmission time, transit delay)
- all area parameters
- all area range parameters
- all virtual-link parameters
- all global parameters
- creation and deletion of an area, interface or virtual link

In addition, you can make the following changes without a system reset by first disabling and then re-enabling OSPF operation:

- changes to address ranges
- changes to global values for redistribution
- addition of new virtual links

You also can change the amount of memory allocated to various types of LSA entries. However, these changes require a system reset or reboot.

Dynamic OSPF Memory

FastIron devices dynamically allocate memory for Link State Advertisements (LSAs) and other OSPF data structures. This eliminates overflow conditions and does not require a reload to change OSPF memory allocation. So long as the Layer 3 Switch has free (unallocated) dynamic memory, OSPF can use the memory.

To display the current allocations of dynamic memory, enter the show memory command. See the *Foundry Switch and Router Command Line Interface Reference*.

Configuring OSPF

To begin using OSPF on the router, perform the steps outlined below:

1. Enable OSPF on the router.
2. Assign the areas to which the router will be attached.
3. Assign individual interfaces to the OSPF areas.
4. Define redistribution filters, if desired.
5. Enable redistribution, if you defined redistribution filters.
6. Modify default global and port parameters as required.
7. Modify OSPF standard compliance, if desired.

NOTE: OSPF is automatically enabled without a system reset.

Configuration Rules

- If a router is to operate as an ASBR, you must enable the ASBR capability at the system level.
- Redistribution must be enabled on routers configured to operate as ASBRs.
- All router ports must be assigned to one of the defined areas on an OSPF router. When a port is assigned to an area, all corresponding sub-nets on that port are automatically included in the assignment.

OSPF Parameters

You can modify or set the following global and interface OSPF parameters.

Global Parameters

- Modify OSPF standard compliance setting.
- Assign an area.
- Define an area range.
- Define the area virtual link.
- Set global default metric for OSPF.
- Change the reference bandwidth for the default cost of OSPF interfaces.
- Disable or re-enable load sharing.
- Enable or disable default-information-originate.
- Modify Shortest Path First (SPF) timers
- Define external route summarization
- Define redistribution metric type.
- Define deny redistribution.
- Define permit redistribution.
- Enable redistribution.
- Change the LSA pacing interval.
- Modify OSPF Traps generated.
- Modify database overflow interval.

Interface Parameters

- Assign interfaces to an area.
- Define the authentication key for the interface.
- Change the authentication-change interval
- Modify the cost for a link.
- Modify the dead interval.
- Modify MD5 authentication key parameters.
- Modify the priority of the interface.
- Modify the retransmit interval for the interface.
- Modify the transit delay of the interface.

NOTE: When using the CLI, you set global level parameters at the OSPF CONFIG Level of the CLI. To reach that level, enter **router ospf…** at the global CONFIG Level. Interface parameters for OSPF are set at the interface CONFIG Level using the CLI command, **ip ospf…**

When using the Web management interface, you set OSPF global parameters using the OSPF configuration panel. All other parameters are accessed through links accessed from the OSPF configuration sheet.

Enable OSPF on the Router

When you enable OSPF on the router, the protocol is automatically activated. To enable OSPF on the router, enter the following CLI command:

```
FESX424 Router(config)# router ospf
```

This command launches you into the OSPF router level where you can assign areas and modify OSPF global parameters.

Note Regarding Disabling OSPF

If you disable OSPF, the Layer 3 Switch removes all the configuration information for the disabled protocol from the running-config. Moreover, when you save the configuration to the startup-config file after disabling one of these protocols, all the configuration information for the disabled protocol is removed from the startup-config file.

NOTE: If you are running software release 02.4.00 or later and do not want to delete the OSPF configuration information, use the CLI command **clear ip ospf process** instead of **no router ospf**. See "Resetting OSPF" on page 20-10.

When you enter the **no router ospf** command, the CLI displays a warning message such as the following:

```
FESX424 Router(config-ospf-router)# no router ospf
router ospf mode now disabled. All ospf config data will be lost when writing to
flash!
```

The Web management interface does not display a warning message.

If you have disabled the protocol but have not yet saved the configuration to the startup-config file and reloaded the software, you can restore the configuration information by re-entering the command to enable the protocol (ex: **router ospf**), or by selecting the Web management option to enable the protocol. If you have already saved the configuration to the startup-config file and reloaded the software, the information is gone.

If you are testing an OSPF configuration and are likely to disable and re-enable the protocol, you might want to make a backup copy of the startup-config file containing the protocol's configuration information. This way, if you remove the configuration information by saving the configuration after disabling the protocol, you can restore the configuration by copying the backup copy of the startup-config file onto the flash memory.

Resetting OSPF

Software release 02.4.00 introduces a new CLI command, **clear ip ospf process**, which globally resets (disables then re-enables) OSPF without deleting the OSPF configuration information. This command is equivalent to entering the commands **no router ospf** followed by **router ospf**. Whereas the **no router ospf** command disables OSPF and removes all the configuration information for the disabled protocol from the running-config, the **router ospf** command re-enables OSPF and restores the OSPF configuration information.

The **clear ip ospf process** command is useful If you are testing an OSPF configuration and are likely to disable and re-enable the protocol. This way, you do not have to save the configuration after disabling the protocol, and you do not have to restore the configuration by copying the backup copy of the startup-config file onto the flash memory.

To reset OSPF without deleting the OSPF configuration, enter the following command at the Global CONFIG level or at the Router OSPF level of the CLI:

```
FESX424 router(config)# clear ip ospf process
```

Syntax: clear ip ospf process

Assign OSPF Areas

Once OSPF is enabled on the system, you can assign areas. Assign an IP address or number as the ***area ID*** for each area. The area ID is representative of all IP addresses (sub-nets) on a router port. Each port on a router can support one area.

An area can be ***normal***, a ***stub***, or a ***Not-So-Stubby Area (NSSA)***.

- Normal – OSPF routers within a normal area can send and receive External Link State Advertisements (LSAs).
- Stub – OSPF routers within a stub area cannot send or receive External LSAs. In addition, OSPF routers in a stub area must use a default route to the area's Area Border Router (ABR) or Autonomous System Boundary Router (ASBR) to send traffic out of the area.
- NSSA – The ASBR of an NSSA can import external route information into the area.
    - ASBRs redistribute (import) external routes into the NSSA as type 7 LSAs. Type-7 External LSAs are a special type of LSA generated only by ASBRs within an NSSA, and are flooded to all the routers within only that NSSA.
    - ABRs translate type 7 LSAs into type 5 External LSAs, which can then be flooded throughout the AS. You can configure address ranges on the ABR of an NSSA so that the ABR converts multiple type-7 External LSAs received from the NSSA into a single type-5 External LSA.

      When an NSSA contains more than one ABR, OSPF elects one of the ABRs to perform the LSA translation for NSSA. OSPF elects the ABR with the highest router ID. If the elected ABR becomes unavailable, OSPF automatically elects the ABR with the next highest router ID to take over translation of LSAs for the NSSA. The election process for NSSA ABRs is automatic.

EXAMPLE:

To set up the OSPF areas shown in Figure 20.1 on page 20-3, enter the following commands.

```
FESX424 Router(config-ospf-router)# area 192.5.1.0
FESX424 Router(config-ospf-router)# area 200.5.0.0
FESX424 Router(config-ospf-router)# area 195.5.0.0
FESX424 Router(config-ospf-router)# area 0.0.0.0
FESX424 Router(config-ospf-router) write memory
```

Syntax: area <num> | <ip-addr>

The <num> | <ip-addr> parameter specifies the area number, which can be a number or in IP address format. If you specify an number, the number can be from 0 – 2,147,483,647.

NOTE: You can assign one area on a router interface. For example, if the system or chassis module has 16 ports, 16 areas are supported on the chassis or module.

Assign a Totally Stubby Area

By default, the Layer 3 Switch sends summary LSAs (LSA type 3) into stub areas. You can further reduce the number of link state advertisements (LSA) sent into a stub area by configuring the Layer 3 Switch to stop sending summary LSAs (type 3 LSAs) into the area. You can disable the summary LSAs when you are configuring the stub area or later after you have configured the area.

This feature disables origination of summary LSAs, but the Layer 3 Switch still accepts summary LSAs from OSPF neighbors and floods them to other neighbors. The Layer 3 Switch can form adjacencies with other routers regardless of whether summarization is enabled or disabled for areas on each router.

When you enter a command or apply a Web management option to disable the summary LSAs, the change takes effect immediately. If you apply the option to a previously configured area, the Layer 3 Switch flushes all of the summary LSAs it has generated (as an ABR) from the area.

NOTE: This feature applies only when the Layer 3 Switch is configured as an Area Border Router (ABR) for the area. To completely prevent summary LSAs from being sent to the area, disable the summary LSAs on each OSPF router that is an ABR for the area.

This feature does not apply to Not So Stubby Areas (NSSAs).

To disable summary LSAs for a stub area, enter commands such as the following:

```
FESX424 Router(config-ospf-router)# area 40 stub 99 no-summary
```

Syntax: area <num> | <ip-addr> stub <cost> [no-summary]

The <num> | <ip-addr> parameter specifies the area number, which can be a number or in IP address format. If you specify a number, the number can be from 0 – 2,147,483,647.

The **stub** <cost> parameter specifies an additional cost for using a route to or from this area and can be from 1 – 16777215. There is no default. Normal areas do not use the cost parameter.

The **no-summary** parameter applies only to stub areas and disables summary LSAs from being sent into the area.

NOTE: You can assign one area on a router interface. For example, if the system or chassis module has 16 ports, 16 areas are supported on the chassis or module.

Assign a Not-So-Stubby Area (NSSA)

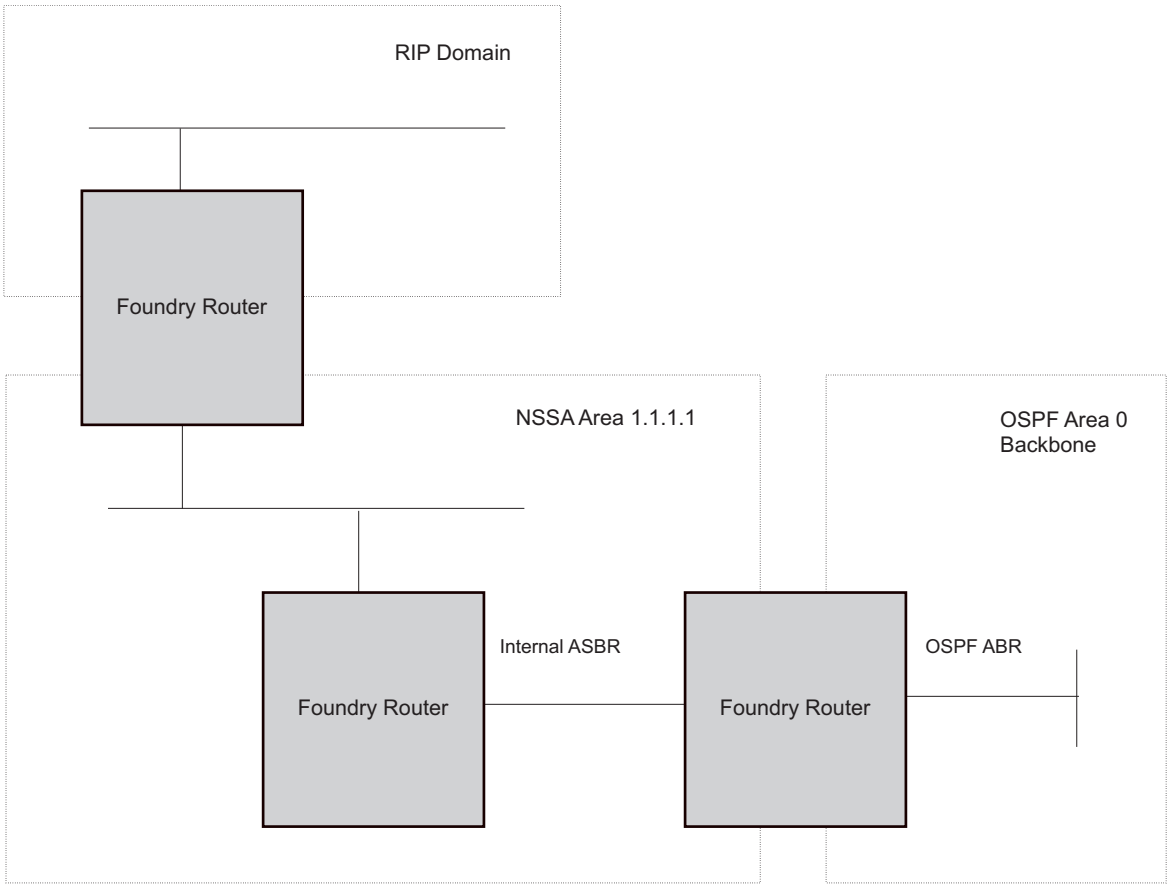
The OSPF Not So Stubby Area (NSSA) feature enables you to configure OSPF areas that provide the benefits of stub areas, but that also are capable of importing external route information. OSPF does not flood external routes from other areas into an NSSA, but does translate and flood route information from the NSSA into other areas such as the backbone.

NSSAs are especially useful when you want to summarize Type-5 External LSAs (external routes) before forwarding them into an OSPF area. The OSPF specification (RFC 2328) prohibits summarization of Type-5 LSAs and requires OSPF to flood Type-5 LSAs throughout a routing domain. When you configure an NSSA, you can specify an address range for aggregating the external routes that the NSSA's ABR exports into other areas.

The Foundry implementation of NSSA is based on RFC 1587.

Figure 20.5 shows an example of an OSPF network containing an NSSA.

Figure 20.5 OSPF network containing an NSSA



This example shows two routing domains, a RIP domain and an OSPF domain. The ASBR inside the NSSA imports external routes from RIP into the NSSA as Type-7 LSAs, which the ASBR floods throughout the NSSA.

The ABR translates the Type-7 LSAs into Type-5 LSAs. If an area range is configured for the NSSA, the ABR also summarizes the LSAs into an aggregate LSA before flooding the Type-5 LSA(s) into the backbone.

Since the NSSA is partially “stubby” the ABR does not flood external LSAs from the backbone into the NSSA. To provide access to the rest of the Autonomous System (AS), the ABR generates a default Type-7 LSA into the NSSA.

Configuring an NSSA

To configure OSPF area 1.1.1.1 as an NSSA, enter the following commands.

```
FESX424 Router(config)# router ospf
FESX424 Router(config-ospf-router)# area 1.1.1.1 nssa 1
FESX424 Router(config-ospf-router)# write memory
```

Syntax: area <num> | <ip-addr> nssa <cost> | default-information-originate

The <num> | <ip-addr> parameter specifies the area number, which can be a number or in IP address format. If you specify an number, the number can be from 0 – 2,147,483,647.

The **nssa** <cost> | **default-information-originate** parameter specifies that this is a Not-So-Stubby-Area (NSSA). The <cost> specifies an additional cost for using a route to or from this NSSA and can be from 1 – 16777215. There is no default. Normal areas do not use the cost parameter. Alternatively, you can use the **default-information-originate** parameter causes the Layer 3 Switch to inject the default route into the NSSA.

NOTE: The Layer 3 Switch does not inject the default route into an NSSA by default.

NOTE: You can assign one area on a router interface. For example, if the system or chassis module has 16 ports, 16 areas are supported on the chassis or module.

To configure additional parameters for OSPF interfaces in the NSSA, use the **ip ospf area…** command at the interface level of the CLI.

Configuring an Address Range for the NSSA

If you want the ABR that connects the NSSA to other areas to summarize the routes in the NSSA before translating them into Type-5 LSAs and flooding them into the other areas, configure an address range. The ABR creates an aggregate value based on the address range. The aggregate value becomes the address that the ABR advertises instead of advertising the individual addresses represented by the aggregate. You can configure up to 32 ranges in an OSPF area.

To configure an address range in NSSA 1.1.1.1, enter the following commands. This example assumes that you have already configured NSSA 1.1.1.1.

```
FESX424 Router(config)# router ospf
FESX424 Router(config-ospf-router)# area 1.1.1.1 range 209.157.22.1 255.255.0.0
FESX424 Router(config-ospf-router)# write memory
```

Syntax: [no] area <num> | <ip-addr> range <ip-addr> <ip-mask> [advertise | not-advertise]

The <num> | <ip-addr> parameter specifies the area number, which can be in IP address format. If you specify a number, the number can be from 0 – 2,147,483,647.

The **range** <ip-addr> parameter specifies the IP address portion of the range. The software compares the address with the significant bits in the mask. All network addresses that match this comparison are summarized in a single route advertised by the router.

The <ip-mask> parameter specifies the portions of the IP address that a route must contain to be summarized in the summary route. In the example above, all networks that begin with 209.157 are summarized into a single route.

The **advertise | not-advertise** parameter specifies whether you want the Layer 3 Switch to send type 3 LSAs for the specified range in this area. The default is **advertise**.

Assigning an Area Range (optional)

You can assign a ***range*** for an area, but it is not required. Ranges allow a specific IP address and mask to represent a range of IP addresses within an area, so that only that reference range address is advertised to the network, instead of all the addresses within that range. Each area can have up to 32 range addresses.

EXAMPLE:

To define an area range for sub-nets on 193.45.5.1 and 193.45.6.2, enter the following command:

```
FESX424 Router(config)# router ospf
FESX424 Router(config-ospf-router)# area 192.45.5.1 range 193.45.0.0 255.255.0.0
FESX424 Router(config-ospf-router)# area 193.45.6.2 range 193.45.0.0 255.255.0.0
```

Syntax: area <num> | <ip-addr> range <ip-addr> <ip-mask>

The <num> | <ip-addr> parameter specifies the area number, which can be in IP address format.

The **range** <ip-addr> parameter specifies the IP address portion of the range. The software compares the address with the significant bits in the mask. All network addresses that match this comparison are summarized in a single route advertised by the router.

The <ip-mask> parameter specifies the portions of the IP address that a route must contain to be summarized in the summary route. In the example above, all networks that begin with 193.45 are summarized into a single route.

Assigning Interfaces to an Area

Once you define OSPF areas, you can assign interfaces to the areas. All router ports must be assigned to one of the defined areas on an OSPF router. When a port is assigned to an area, all corresponding sub-nets on that port are automatically included in the assignment.

To assign interface 1/8 to area 192.5.0.0 and then save the changes, enter the following commands:

```
FastIron SuperX Router(config-ospf-router)# interface e 1/8
FastIron SuperX Router(config-if-1/8)# ip ospf area 192.5.0.0
FastIron SuperX Router(config-if-1/8)# write memory
```

Modify Interface Defaults

OSPF has interface parameters that you can configure. For simplicity, each of these parameters has a default value. No change to these default values is required except as needed for specific network configurations.

Port default values can be modified using the following CLI commands at the interface configuration level of the CLI:

- ip ospf area <ip-addr>
- ip ospf auth-change-wait-time <secs>
- ip ospf authentication-key [0 | 1] <string>
- ip ospf cost <num>
- ip ospf dead-interval <value>
- ip ospf hello-interval <value>
- ip ospf md5-authentication key-activation-wait-time <num> | key-id <num> [0 | 1] key <string>
- ip ospf passive
- ip ospf priority <value>
- ip ospf retransmit-interval <value>
- ip ospf transmit-delay <value>

For a complete description of these parameters, see the summary of OSPF port parameters in the next section.

OSPF Interface Parameters

The following parameters apply to OSPF interfaces.

Area: Assigns an interface to a specific area. You can assign either an IP address or number to represent an OSPF Area ID. If you assign a number, it can be any value from 0 – 2,147,483,647.

Auth-change-wait-time: OSPF gracefully implements authentication changes to allow all routers to implement the change and thus prevent disruption to neighbor adjacencies. During the authentication-change interval, both the old and new authentication information is supported. The default authentication-change interval is 300 seconds (5 minutes). You change the interval to a value from 0 – 14400 seconds.

Authentication-key: OSPF supports three methods of authentication for each interface—none, simple password, and MD5. Only one method of authentication can be active on an interface at a time. The default authentication value is none, meaning no authentication is performed.

- The simple password method of authentication requires you to configure an alphanumeric password on an interface. The simple password setting takes effect immediately. All OSPF packets transmitted on the interface contain this password. Any OSPF packet received on the interface is checked for this password. If the password is not present, then the packet is dropped. The password can be up to eight characters long.
- The MD5 method of authentication requires you to configure a key ID and an MD5 Key. The key ID is a number from 1 – 255 and identifies the MD5 key that is being used. The MD5 key can be up to sixteen alphanumeric characters long.

Cost: Indicates the overhead required to send a packet across an interface. You can modify the cost to differentiate between 100 Mbps and 1000 Mbps (1 Gbps) links. The default cost is calculated by dividing 100 million by the bandwidth. For 10 Mbps links, the cost is 10. The cost for both 100 Mbps and 1000 Mbps links is 1, because the speed of 1000 Mbps was not in use at the time the OSPF cost formula was devised.

Dead-interval: Indicates the number of seconds that a neighbor router waits for a hello packet from the current router before declaring the router down. The value can be from 1 – 65535 seconds. The default is 40 seconds.

Hello-interval: Represents the length of time between the transmission of hello packets. The value can be from 1 – 65535 seconds. The default is 10 seconds.

MD5-authentication activation wait time: The number of seconds the Layer 3 Switch waits until placing a new MD5 key into effect. The wait time provides a way to gracefully transition from one MD5 key to another without disturbing the network. The wait time can be from 0 – 14400 seconds. The default is 300 seconds (5 minutes).

MD5-authentication key ID and key: A method of authentication that requires you to configure a key ID and an MD5 key. The key ID is a number from 1 – 255 and identifies the MD5 key that is being used. The MD5 key consists of up to 16 alphanumeric characters. The MD5 is encrypted and included in each OSPF packet transmitted.

Passive: When you configure an OSPF interface to be passive, that interface does not send or receive OSPF route updates. By default, all OSPF interfaces are active and thus can send and receive OSPF route information. Since a passive interface does not send or receive route information, the interface is in effect a stub network. OSPF interfaces are active by default.

NOTE: This option affects all IP sub-nets configured on the interface. If you want to disable OSPF updates only on some of the IP sub-nets on the interface, use the **ospf-ignore** or **ospf-passive** parameter with the **ip address** command. See “Assigning an IP Address to an Ethernet Port” on page 16-17.

Priority: Allows you to modify the priority of an OSPF router. The priority is used when selecting the designated router (DR) and backup designated routers (BDRs). The value can be from 0 – 255. The default is 1. If you set the priority to 0, the Layer 3 Switch does not participate in DR and BDR election.

Retransmit-interval: The time between retransmissions of link-state advertisements (LSAs) to adjacent routers for this interface. The value can be from 0 – 3600 seconds. The default is 5 seconds.

Transit-delay: The time it takes to transmit Link State Update packets on this interface. The value can be from 0 – 3600 seconds. The default is 1 second.

Encrypted Display of the Authentication String or MD5 Authentication Key

The optional **0 | 1** parameter with the **authentication-key** and **md5-authentication key-id** parameters affects encryption.

For added security, FastIron devices encrypt display of the password or authentication string. Encryption is enabled by default. The software also provides an optional parameter to disable encryption of a password or authentication string, on an individual OSPF area or OSPF interface basis.

When encryption of the passwords or authentication strings is enabled, they are encrypted in the CLI regardless of the access level you are using. In the Web management interface, the passwords or authentication strings are encrypted at the read-only access level but are visible at the read-write access level.

The encryption option can be omitted (the default) or can be one of the following.

- **0** – Disables encryption for the password or authentication string you specify with the command. The password or string is shown as clear text in the running-config and the startup-config file. Use this option of you do not want display of the password or string to be encrypted.
- **1** – Assumes that the password or authentication string you enter is the encrypted form, and decrypts the value before using it.

NOTE: If you want the software to assume that the value you enter is the clear-text form, and to encrypt display of that form, do not enter **0** or **1**. Instead, omit the encryption option and allow the software to use the default behavior.

If you specify encryption option **1**, the software assumes that you are entering the encrypted form of the password or authentication string. In this case, the software decrypts the password or string you enter before using the value for authentication. If you accidentally enter option **1** followed by the clear-text version of the password or string, authentication will fail because the value used by the software will not match the value you intended to use.

Change the Timer for OSPF Authentication Changes

When you make an OSPF authentication change, the software uses the authentication-change timer to gracefully implement the change. The software implements the change in the following ways:

- Outgoing OSPF packets – After you make the change, the software continues to use the old authentication to send packets, during the remainder of the current authentication-change interval. After this, the software uses the new authentication for sending packets.
- Inbound OSPF packets – The software accepts packets containing the new authentication and continues to accept packets containing the older authentication for two authentication-change intervals. After the second interval ends, the software accepts packets only if they contain the new authentication key.

The default authentication-change interval is 300 seconds (5 minutes). You change the interval to a value from 0 – 14400 seconds.

OSPF provides graceful authentication change for all the following types of authentication changes in OSPF:

- Changing authentication methods from one of the following to another of the following:
  - Simple text password
  - MD5 authentication
  - No authentication
- Configuring a new simple text password or MD5 authentication key
- Changing an existing simple text password or MD5 authentication key

To change the authentication-change interval, enter a command such as the following at the interface configuration level of the CLI:

```
FastIron SuperX Switch(config-if-2/5)# ip ospf auth-change-wait-time 400
```

Syntax: [no] ip ospf auth-change-wait-time <secs>

The <secs> parameter specifies the interval and can be from 0 – 14400 seconds. The default is 300 seconds (5 minutes).

NOTE: For backward compatibility, the **ip ospf md5-authentication key-activation-wait-time** <seconds> command is still supported.

Block Flooding of Outbound LSAs on Specific OSPF Interfaces

By default, the Layer 3 Switch floods all outbound LSAs on all the OSPF interfaces within an area. You can configure a filter to block outbound LSAs on an OSPF interface. This feature is particularly useful when you want to block LSAs from some, but not all, of the interfaces attached to the area.

After you apply filters to block the outbound LSAs, the filtering occurs during the database synchronization and flooding.

If you remove the filters, the blocked LSAs are automatically re-flooded. You do not need to reset OSPF to re-flood the LSAs.

NOTE: You cannot block LSAs on virtual links.

To apply a filter to an OSPF interface to block flooding of outbound LSAs on the interface, enter the following command at the Interface configuration level for that interface.

```
FastIron SuperX Switch(config-if-1/1)# ip ospf database-filter all out
```

The command in this example blocks all outbound LSAs on the OSPF interface configured on port 1/1.

Syntax: [no] ip ospf database-filter all out

To remove the filter, enter a command such as the following:

```
FastIron SuperX Switch(config-if-1/1)# no ip ospf database-filter all out
```

Configuring an OSPF Non-Broadcast Interface

Starting with release 02.3.01, the FESX and FSX Layer 3 switches support Non-Broadcast Multi-Access (NBMA) networks. This feature enables you to configure an interface on a Foundry device to send OSPF traffic to its neighbor as unicast packets rather than broadcast packets.

OSPF routers generally use broadcast packets to establish neighbor relationships and broadcast route updates on Ethernet and virtual interfaces (VEs). In this release, as an alternative, you can configure the Foundry device to use unicast packets for this purpose. This can be useful in situations where multicast traffic is not feasible (for example when a firewall does not allow multicast packets).

On a non-broadcast interface, the routers at the other end of this interface must also be configured as non-broadcast and neighbor routers. There is no restriction on the number of routers sharing a non-broadcast interface (for example, through a hub/switch).

NOTE: Only Ethernet interfaces or VEs can be configured as non-broadcast interfaces.

To configure an OSPF interface as a non-broadcast interface, enable the feature on a physical interface or a VE, following the **ip ospf area** statement, and then specify the IP address of the neighbor in the OSPF configuration. The non-broadcast interface configuration must be done on the OSPF routers on both ends of the link.

For example, the following commands configure VE 20 as a non-broadcast interface:

```
FESX424 router(config)# int ve 20
FESX424 router(config-vif-20)# ip ospf area 0
FESX424 router(config-vif-20)# ip ospf network non-broadcast
FESX424 router(config-vif-20)# exit
```

Syntax: [no] ip ospf network non-broadcast

The following commands specify 1.1.20.1 as an OSPF neighbor address. The address specified must be in the same sub-net as a non-broadcast interface.

```
FESX424 router(config)# router ospf
FESX424 router(config-ospf-router)# neighbor 1.1.20.1
```

For example, to configure the feature in a network with three routers connected by a hub or switch, each router must have the linking interface configured as a non-broadcast interface, and both of the other routers must be specified as neighbors.

The output of the **show ip ospf interface** command has been enhanced to display information about non-broadcast interfaces and neighbors that are configured in the same sub-net. For example:

```
FESX424 router# show ip ospf interface
v20,OSPF enabled
     IP Address 1.1.20.4, Area 0
     OSPF state BD, Pri 1, Cost 1, Options 2, Type non-broadcast Events 6
     Timers(sec): Transit 1, Retrans 5, Hello 10, Dead 40
```



```
DR:  Router ID 1.1.13.1          Interface Address 1.1.20.5
BDR: Router ID 2.2.2.1           Interface Address 1.1.20.4
Neighbor Count = 1, Adjacent Neighbor Count= 2
Non-broadcast neighbor config: 1.1.20.1, 1.1.20.2, 1.1.20.3, 1.1.20.5,
Neighbor:         1.1.20.5
Authentication-Key:None
MD5 Authentication: Key None, Key-Id None, Auth-change-wait-time 300
```

In the Type field, “non-broadcast” indicates that this is a non-broadcast interface. When the interface type is non-broadcast, the Non-broadcast neighbor config field displays the neighbors that are configured in the same sub-net. If no neighbors are configured in the same sub-net, a message such as the following is displayed:

```
***Warning! no non-broadcast neighbor config in 1.1.100.1 255.255.255.0
```

Assign Virtual Links

All ABRs (area border routers) must have either a direct or indirect link to the OSPF backbone area (0.0.0.0 or 0). If an ABR does not have a physical link to the area backbone, the ABR can configure a ***virtual link*** to another router within the same area, which has a physical connection to the area backbone.

The path for a virtual link is through an area shared by the neighbor ABR (router with a physical backbone connection), and the ABR requiring a logical connection to the backbone.

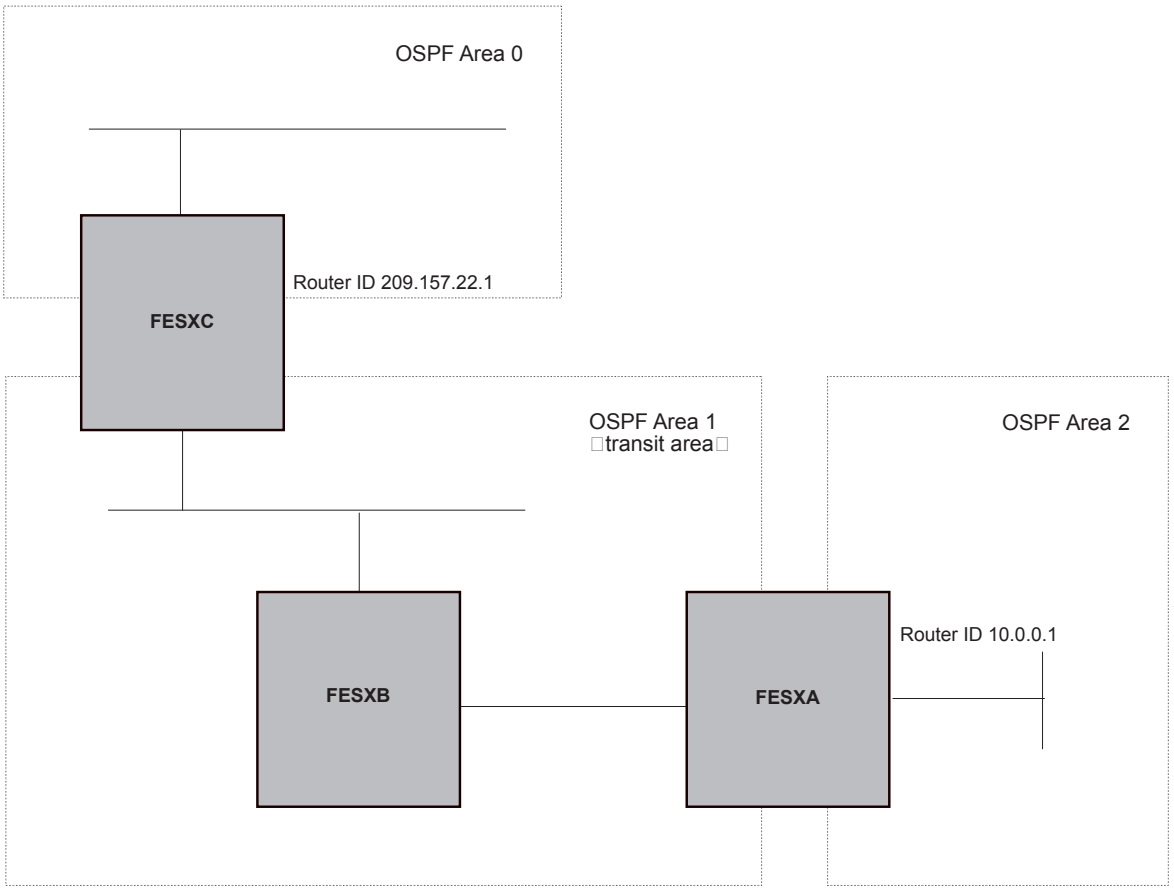
Two parameters fields must be defined for all virtual links—transit area ID and neighbor router.

- The ***transit area ID*** represents the shared area of the two ABRs and serves as the connection point between the two routers. This number should match the area ID value.
- The ***neighbor router*** field is the router ID (IP address) of the router that is physically connected to the backbone, when assigned from the router interface requiring a logical connection. When assigning the parameters from the router with the physical connection, the router ID is the IP address of the router requiring a logical connection to the backbone.

NOTE: By default, the Foundry router ID is the IP address configured on the lowest numbered loopback interface. If the Layer 3 Switch does not have a loopback interface, the default router ID is the lowest numbered IP address configured on the device. For more information or to change the router ID, see “Changing the Router ID” on page 16-23.

NOTE: When you establish an area virtual link, you must configure it on both of the routers (both ends of the virtual link).

Figure 20.6 Defining OSPF virtual links within a network



EXAMPLE:

Figure 20.6 shows an OSPF area border router, FastIronA, that is cut off from the backbone area (area 0). To provide backbone access to FastIronA, you can add a virtual link between FastIronA and FastIronC using area 1 as a transit area. To configure the virtual link, you define the link on the router that is at each end of the link. No configuration for the virtual link is required on the routers in the transit area.

To define the virtual link on FastIronA, enter the following commands:

```
FESX424 RouterA(config-ospf-router)# area 1 virtual-link 209.157.22.1
FESX424 RouterA(config-ospf-router)# write memory
```

Enter the following commands to configure the virtual link on FastIronC:

```
FESX424 RouterC(config-ospf-router)# area 1 virtual-link 10.0.0.1
FESX424 RouterC(config-ospf-router)# write memory
```

Syntax: area <ip-addr> | <num> virtual-link <router-id>
[authentication-key | dead-interval | hello-interval | retransmit-interval | transmit-delay <value>]

The **area** <ip-addr> | <num> parameter specifies the transit area.

The <router-id> parameter specifies the router ID of the OSPF router at the remote end of the virtual link. To display the router ID on a Foundry Layer 3 Switch, enter the **show ip** command.

See “Modify Virtual Link Parameters” on page 20-21 for descriptions of the optional parameters.

Modify Virtual Link Parameters

OSPF has some parameters that you can modify for virtual links. Notice that these are the same parameters as the ones you can modify for physical interfaces.

You can modify default values for virtual links using the following CLI command at the ***OSPF router level*** of the CLI, as shown in the following syntax:

Syntax: area <num> | <ip-addr> virtual-link <ip-addr> [authentication-key [0 | 1] <string>] [dead-interval <num>] [hello-interval <num>] [md5-authentication key-activation-wait-time <num> | key-id <num> [0 | 1] key <string>] [retransmit-interval <num>] [transmit-delay <num>]

The parameters are described below. For syntax information, see the *Foundry Switch and Router Command Line Interface Reference*.

Virtual Link Parameter Descriptions

You can modify the following virtual link interface parameters:

Authentication Key: This parameter allows you to assign different authentication methods on a port-by-port basis. OSPF supports three methods of authentication for each interface—none, simple password, and MD5. Only one method of authentication can be active on an interface at a time.

The simple password method of authentication requires you to configure an alphanumeric password on an interface. The password can be up to eight characters long. The simple password setting takes effect immediately. All OSPF packets transmitted on the interface contain this password. All OSPF packets received on the interface are checked for this password. If the password is not present, then the packet is dropped.

The MD5 method of authentication encrypts the authentication key you define. The authentication is included in each OSPF packet transmitted.

MD5 Authentication Key: When simple authentication is enabled, the key is an alphanumeric password of up to eight characters. When MD5 is enabled, the key is an alphanumeric password of up to 16 characters that is later encrypted and included in each OSPF packet transmitted. You must enter a password in this field when the system is configured to operate with either simple or MD5 authentication.

MD5 Authentication Key ID: The Key ID is a number from 1 – 255 and identifies the MD5 key that is being used. This parameter is required to differentiate among multiple keys defined on a router.

MD5 Authentication Wait Time: This parameter determines when a newly configured MD5 authentication key is valid. This parameter provides a graceful transition from one MD5 key to another without disturbing the network. All new packets transmitted after the key activation wait time interval use the newly configured MD5 Key. OSPF packets that contain the old MD5 key are accepted for up to five minutes after the new MD5 key is in operation.

The range for the key activation wait time is from 0 – 14400 seconds. The default value is 300 seconds.

Hello Interval: The length of time between the transmission of hello packets. The range is 1 – 65535 seconds. The default is 10 seconds.

Retransmit Interval: The interval between the re-transmission of link state advertisements to router adjacencies for this interface. The range is 0 – 3600 seconds. The default is 5 seconds.

Transmit Delay: The period of time it takes to transmit Link State Update packets on the interface. The range is 0 – 3600 seconds. The default is 1 second.

Dead Interval: The number of seconds that a neighbor router waits for a hello packet from the current router before declaring the router down. The range is 1 – 65535 seconds. The default is 40 seconds.

Encrypted Display of the Authentication String or MD5 Authentication Key

The optional **0 | 1** parameter with the **authentication-key** and **md5-authentication key-id** parameters affects encryption.

For added security, FastIron devices encrypt display of the password or authentication string. Encryption is enabled by default. The software also provides an optional parameter to disable encryption of a password or authentication string, on an individual OSPF area or OSPF interface basis.

When encryption of the passwords or authentication strings is enabled, they are encrypted in the CLI regardless of the access level you are using. In the Web management interface, the passwords or authentication strings are encrypted at the read-only access level but are visible at the read-write access level.

The encryption option can be omitted (the default) or can be one of the following.

- **0** – Disables encryption for the password or authentication string you specify with the command. The password or string is shown as clear text in the running-config and the startup-config file. Use this option of you do not want display of the password or string to be encrypted.
- **1** – Assumes that the password or authentication string you enter is the encrypted form, and decrypts the value before using it.

NOTE: If you want the software to assume that the value you enter is the clear-text form, and to encrypt display of that form, do not enter **0** or **1**. Instead, omit the encryption option and allow the software to use the default behavior.

If you specify encryption option **1**, the software assumes that you are entering the encrypted form of the password or authentication string. In this case, the software decrypts the password or string you enter before using the value for authentication. If you accidentally enter option **1** followed by the clear-text version of the password or string, authentication will fail because the value used by the software will not match the value you intended to use.

Changing the Reference Bandwidth for the Cost on OSPF Interfaces

Each interface on which OSPF is enabled has a cost associated with it. The Layer 3 Switch advertises its interfaces and their costs to OSPF neighbors. For example, if an interface has an OSPF cost of ten, the Layer 3 Switch advertises the interface with a cost of ten to other OSPF routers.

By default, an interface's OSPF cost is based on the port speed of the interface. The cost is calculated by dividing the reference bandwidth by the port speed. The default reference bandwidth is 100 Mbps, which results in the following default costs:

- 10 Mbps port – 10
- All other port speeds – 1

You can change the reference bandwidth, to change the costs calculated by the software.

The software uses the following formula to calculate the cost:

Cost = reference-bandwidth/interface-speed

If the resulting cost is less than 1, the software rounds the cost up to 1. The default reference bandwidth results in the following costs:

- 10 Mbps port's cost = 100/10 = 10
- 100 Mbps port's cost = 100/100 = 1
- 1000 Mbps port's cost = 100/1000 = 0.10, which is rounded up to 1
- 155 Mbps port's cost = 100/155 = 0.65, which is rounded up to 1
- 622 Mbps port's cost = 100/622 = 0.16, which is rounded up to 1
- 2488 Mbps port's cost = 100/2488 = 0.04, which is rounded up to 1

The bandwidth for interfaces that consist of more than one physical port is calculated as follows:

- Trunk group – The combined bandwidth of all the ports.
- Virtual interface – The combined bandwidth of all the ports in the port-based VLAN that contains the virtual interface.

The default reference bandwidth is 100 Mbps. You can change the reference bandwidth to a value from 1 – 4294967.

If a change to the reference bandwidth results in a cost change to an interface, the Layer 3 Switch sends a link-state update to update the costs of interfaces advertised by the Layer 3 Switch.

NOTE: If you specify the cost for an individual interface, the cost you specify overrides the cost calculated by the software.

Interface Types To Which the Reference Bandwidth Does Not Apply

Some interface types are not affected by the reference bandwidth and always have the same cost regardless of the reference bandwidth in use:

- The cost of a loopback interface is always 0.
- The cost of a virtual link is calculated using the Shortest Path First (SPF) algorithm and is not affected by the auto-cost feature.
- The bandwidth for tunnel interfaces is 9 Kbps and is not affected by the auto-cost feature.

Changing the Reference Bandwidth

To change the reference bandwidth, enter a command such as the following at the OSPF configuration level of the CLI:

```
FESX424 Router(config-ospf-router)# auto-cost reference-bandwidth 500
```

The reference bandwidth specified in this example results in the following costs:

- 10 Mbps port's cost = 500/10 = 50
- 100 Mbps port's cost = 500/100 = 5
- 1000 Mbps port's cost = 500/1000 = 0.5, which is rounded up to 1
- 155 Mbps port's cost = 500/155 = 3.23, which is rounded up to 4
- 622 Mbps port's cost = 500/622 = 0.80, which is rounded up to 1
- 2488 Mbps port's cost = 500/2488 = 0.20, which is rounded up to 1

The costs for 10 Mbps, 100 Mbps, and 155 Mbps ports change as a result of the changed reference bandwidth. Costs for higher-speed interfaces remain the same.

Syntax: [no] auto-cost reference-bandwidth <num>

The <num> parameter specifies the reference bandwidth and can be a value from 1 – 4294967. The default is 100, which results in the same costs as previous software releases.

To restore the reference bandwidth to its default value and thus restore the default costs of interfaces to their default values, enter the following command:

```
FESX424 Router(config-ospf-router)# no auto-cost reference-bandwidth
```

Define Redistribution Filters

Route redistribution imports and translates different protocol routes into a specified protocol type. On Foundry routers, redistribution is supported for static routes, OSPF, RIP, and BGP4. When you configure redistribution for RIP, you can specify that static, OSPF, or BGP4 routes are imported into RIP routes. Likewise, OSPF redistribution supports the import of static, RIP, and BGP4 routes into OSPF routes. BGP4 supports redistribution of static, RIP, and OSPF routes into BGP4.

NOTE: The Layer 3 Switch advertises the default route into OSPF even if redistribution is not enabled, and even if the default route is learned through an IBGP neighbor. IBGP routes (including the default route) are not redistributed into OSPF by OSPF redistribution (for example, by the OSPF **redistribute** command).

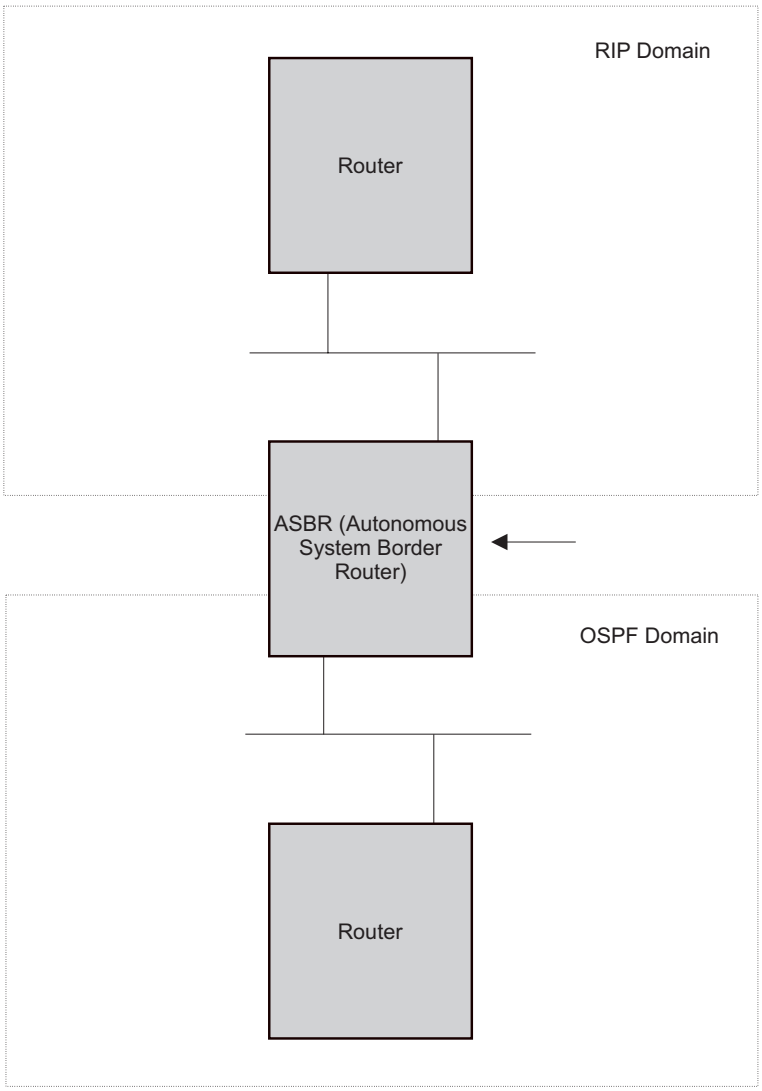
In Figure 20.7 on page 20-24, an administrator wants to configure the FastIron Layer 3 Switch acting as the ASBR (Autonomous System Boundary Router) between the RIP domain and the OSPF domain to redistribute routes between the two domains.

NOTE: The ASBR must be running both RIP and OSPF protocols to support this activity.

To configure for redistribution, define the redistribution tables with deny and permit redistribution filters. Use the **deny** and **permit** redistribute commands for OSPF at the OSPF router level.

NOTE: Do not enable redistribution until you have configured the redistribution filters. If you enable redistribution before you configure the redistribution filters, the filters will not take affect and all routes will be distributed.

Figure 20.7 Redistributing OSPF and static routes to RIP routes



EXAMPLE:

To configure the FastIron Layer 3 Switch acting as an ASBR in Figure 20.7 to redistribute OSPF, BGP4, and static routes into RIP, enter the following commands:

```
FESX424 RouterASBR(config)# router rip
FESX424 RouterASBR(config-rip-router)# permit redistribute 1 all
FESX424 RouterASBR(config-rip-router)# write memory
```

NOTE: Redistribution is permitted for all routes by default, so the **permit redistribute 1 all** command in the example above is shown for clarity but is not required.

You also have the option of specifying import of just OSPF, BGP4, or static routes, as well as specifying that only routes for a specific network or with a specific cost (metric) be imported, as shown in the command syntax below:

Syntax: deny | permit redistribute <filter-num> all | bgp | connected | rip | static
[address <ip-addr> <ip-mask> [match-metric <value> [set-metric <value>]]]

EXAMPLE:

To redistribute RIP, static, and BGP4 routes into OSPF, enter the following commands on the Layer 3 Switch acting as an ASBR:

```
FESX424 RouterASBR(config)# router ospf
FESX424 RouterASBR(config-ospf-router)# permit redistribute 1 all
FESX424 RouterASBR(config-ospf-router)# write memory
```

Syntax: deny | permit redistribute <filter-num> all | bgp | connected | rip | static
address <ip-addr> <ip-mask>
[match-metric <value> | set-metric <value>]

NOTE: Redistribution is permitted for all routes by default, so the **permit redistribute 1 all** command in the example above is shown for clarity but is not required.

You also have the option of specifying import of just OSPF, BGP4, or static routes, as well as specifying that only routes for a specific network or with a specific cost (metric) be imported, as shown in the command syntax below:

Syntax: [no] redistribution bgp | connected | rip | static [route-map <map-name>]

For example, to enable redistribution of RIP and static IP routes into OSPF, enter the following commands.

```
FESX424 Router(config)# router ospf
FESX424 Router(config-ospf-router)# redistribution rip
FESX424 Router(config-ospf-router)# redistribution static
FESX424 Router(config-ospf-router)# write memory
```

NOTE: The **redistribution** command does not perform the same function as the **permit redistribute** and **deny redistribute** commands. The **redistribute** commands allow you to control redistribution of routes by filtering on the IP address and network mask of a route. The **redistribution** commands enable redistribution for routes of specific types (static, directly connected, and so on). Configure all your redistribution filters before enabling redistribution.

NOTE: Do not enable redistribution until you have configured the redistribution filters. If you enable redistribution before you configure the redistribution filters, the filters will not take affect and all routes will be distributed.

Prevent Specific OSPF Routes from Being Installed in the IP Route Table

By default, all OSPF routes in the OSPF route table are eligible for installation in the IP route table. You can configure a distribution list to explicitly deny specific routes from being eligible for installation in the IP route table.

NOTE: This feature does not block receipt of LSAs for the denied routes. The Layer 3 Switch still receives the routes and installs them in the OSPF database. The feature only prevents the software from installing the denied OSPF routes into the IP route table.

To configure an OSPF distribution list:

- Configure a standard or extended ACL that identifies the routes you want to deny. Using a standard ACL lets

you deny routes based on the destination network, but does not filter based on the network mask. To also filter based on the destination network's network mask, use an extended ACL.

- Configure an OSPF distribution list that uses the ACL as input.

NOTE: If you change the ACL after you configure the OSPF distribution list, you must clear the IP route table to place the changed ACL into effect. To clear the IP route table, enter the **clear ip route** command at the Privileged EXEC level of the CLI.

The following sections show how to use the CLI to configure an OSPF distribution list. Separate examples are provided for standard and extended ACLs.

NOTE: The examples show named ACLs. However, you also can use a numbered ACL as input to the OSPF distribution list.

Using a Standard ACL as Input to the Distribution List

To use a standard ACL to configure an OSPF distribution list for denying specific routes, enter commands such as the following:

```
FESX424 Router(config)# ip access-list standard no_ip
FESX424 Router(config-std-nacl)# deny 4.0.0.0 0.255.255.255
FESX424 Router(config-std-nacl)# permit any any
FESX424 Router(config-std-nacl)# exit
FESX424 Router(config)# router ospf
FESX424 Router(config-ospf-router)# distribute-list no_ip in
```

The first three commands configure a standard ACL that denies routes to any 4.x.x.x destination network and allows all other routes for eligibility to be installed in the IP route table. The last three commands change the CLI to the OSPF configuration level and configure an OSPF distribution list that uses the ACL as input. The distribution list prevents routes to any 4.x.x.x destination network from entering the IP route table. The distribution list does not prevent the routes from entering the OSPF database.

Syntax: [no] distribute-list <acl-name> | <acl-id> in [<interface type>] [<interface number>]

Syntax: [no] ip access-list standard <acl-name> | <acl-id>

Syntax: deny | permit <source-ip> <wildcard>

The <acl-name> | <acl-id> parameter specifies the ACL name or ID.

The **in** command applies the ACL to incoming route updates.

The <interface number> parameter specifies the interface number on which to apply the ACL. Enter only one valid interface number. If necessary, use the **show interface brief** command to display a list of valid interfaces. If you do not specify an interface, the Foundry device applies the ACL to all incoming route updates.

If you do not specify an interface type and interface number, the device applies the OSPF distribution list to all incoming route updates.

The **deny | permit** parameter indicates whether packets that match the policy are dropped or forwarded.

The <source-ip> parameter specifies the source address for the policy. Since this ACL is input to an OSPF distribution list, the <source-ip> parameter actually is specifying the destination network of the route.

The <wildcard> parameter specifies the portion of the source address to match against. The <wildcard> is a four-part value in dotted-decimal notation (IP address format) consisting of ones and zeros. Zeros in the mask mean the packet's source address must match the <source-ip>. Ones mean any value matches. For example, the <source-ip> and <wildcard> values 4.0.0.0 0.255.255.255 mean that all 4.x.x.x networks match the ACL.

If you want the policy to match on all destination networks, enter **any any**.

If you prefer to specify the wildcard (mask value) in Classless Interdomain Routing (CIDR) format, you can enter a forward slash after the IP address, then enter the number of significant bits in the mask. For example, you can

enter the CIDR equivalent of “4.0.0.0 0.255.255.255” as “4.0.0.0/8”. The CLI automatically converts the CIDR number into the appropriate ACL mask (where zeros instead of ones are the significant bits) and changes the non-significant portion of the IP address into zeros.

NOTE: If you enable the software to display IP sub-net masks in CIDR format, the mask is saved in the file in “/<mask-bits>” format. To enable the software to display the CIDR masks, enter the **ip show-subnet-length** command at the global CONFIG level of the CLI. You can use the CIDR format to configure the ACL entry regardless of whether the software is configured to display the masks in CIDR format.

If you use the CIDR format, the ACL entries appear in this format in the running-config and startup-config files, but are shown with sub-net mask in the display produced by the **show ip access-list** command.

Using an Extended ACL as Input to the Distribution List

To use an extended ACL to configure an OSPF distribution list for denying specific routes, enter commands such as the following:

```
FESX424 Router(config)# ip access-list extended no_ip
FESX424 Router(config-ext-nacl)# deny ip 4.0.0.0 0.255.255.255 255.255.0.0
0.0.255.255
FESX424 Router(config-ext-nacl)# permit ip any any
FESX424 Router(config-ext-nacl)# exit
FESX424 Router(config)# router ospf
FESX424 Router(config-ospf-router)# distribute-list no_ip in
```

The first three commands configure an extended ACL that denies routes to any 4.x.x.x destination network with a 255.255.0.0 network mask and allows all other routes for eligibility to be installed in the IP route table. The last three commands change the CLI to the OSPF configuration level and configure an OSPF distribution list that uses the ACL as input. The distribution list prevents routes to any 4.x.x.x destination network with network mask 255.255.0.0 from entering the IP route table. The distribution list does not prevent the routes from entering the OSPF database.

Syntax: [no] ip access-list extended <acl-name> | <acl-id>

Syntax: deny | permit <ip-protocol> <source-ip> <wildcard> <destination-ip> <wildcard>

The <acl-name> | <acl-id> parameter specifies the ACL name or ID.

The **deny | permit** parameter indicates whether packets that match the policy are dropped or forwarded.

The <ip-protocol> parameter indicates the type of IP packet you are filtering. When using an extended ACL as input for an OSPF distribution list, specify **ip**.

The <source-ip> <wildcard> parameter specifies the source address for the policy. Since this ACL is input to an OSPF distribution list, the <source-ip> parameter actually is specifying the destination network of the route.

The <wildcard> parameter specifies the portion of the source address to match against. The <wildcard> is a four-part value in dotted-decimal notation (IP address format) consisting of ones and zeros. Zeros in the mask mean the packet’s source address must match the <source-ip>. Ones mean any value matches. For example, the <source-ip> and <wildcard> values 4.0.0.0 0.255.255.255 mean that all 4.x.x.x networks match the ACL.

If you want the policy to match on all network addresses, enter **any any**.

If you prefer to specify the wildcard (mask value) in Classless Interdomain Routing (CIDR) format, you can enter a forward slash after the IP address, then enter the number of significant bits in the mask. For example, you can enter the CIDR equivalent of “4.0.0.0 0.255.255.255” as “4.0.0.0/8”. The CLI automatically converts the CIDR number into the appropriate ACL mask (where zeros instead of ones are the significant bits) and changes the non-significant portion of the IP address into zeros.

NOTE: If you enable the software to display IP sub-net masks in CIDR format, the mask is saved in the file in "/<mask-bits>" format. To enable the software to display the CIDR masks, enter the **ip show-subnet-length** command at the global CONFIG level of the CLI. You can use the CIDR format to configure the ACL entry regardless of whether the software is configured to display the masks in CIDR format.

If you use the CIDR format, the ACL entries appear in this format in the running-config and startup-config files, but are shown with sub-net mask in the display produced by the **show ip access-list** commands.

The <destination-ip> <wildcard> parameter specifies the destination address for the policy. Since this ACL is input to an OSPF distribution list, the <destination-ip> parameter actually is specifying the network mask of the destination. The <wildcard> parameter specifies the portion of the destination address to match against. If you want the policy to match on all network masks, enter **any any**.

Modify Default Metric for Redistribution

The default metric is a global parameter that specifies the cost applied to all OSPF routes by default. The default value is 10. You can assign a cost from 1 – 15.

NOTE: You also can define the cost on individual interfaces. The interface cost overrides the default cost.

To assign a default metric of 4 to all routes imported into OSPF, enter the following commands:

```
FESX424 Router(config)# router ospf
FESX424 Router(config-ospf-router)# default-metric 4
```

Syntax: default-metric <value>

The <value> can be from 1 – 16,777,215. The default is 10.

Enable Route Redistribution

To enable route redistribution, use one of the following methods.

NOTE: Do not enable redistribution until you have configured the redistribution filters. Otherwise, you might accidentally overload the network with routes you did not intend to redistribute.

To enable redistribution of RIP and static IP routes into OSPF, enter the following commands.

```
FESX424 Router(config)# router ospf
FESX424 Router(config-ospf-router)# redistribution rip
FESX424 Router(config-ospf-router)# redistribution static
FESX424 Router(config-ospf-router)# write memory
```


Example Using a Route Map

To configure a route map and use it for redistribution of routes into OSPF, enter commands such as the following:

```
FESX424 Router(config)# ip route 1.1.0.0 255.255.0.0 207.95.7.30
FESX424 Router(config)# ip route 1.2.0.0 255.255.0.0 207.95.7.30
FESX424 Router(config)# ip route 1.3.0.0 255.255.0.0 207.95.7.30
FESX424 Router(config)# ip route 4.1.0.0 255.255.0.0 207.95.6.30
FESX424 Router(config)# ip route 4.2.0.0 255.255.0.0 207.95.6.30
FESX424 Router(config)# ip route 4.3.0.0 255.255.0.0 207.95.6.30
FESX424 Router(config)# ip route 4.4.0.0 255.255.0.0 207.95.6.30 5
FESX424 Router(config)# route-map abc permit 1
FESX424 Router(config-routemap abc)# match metric 5
FESX424 Router(config-routemap abc)# set metric 8
FESX424 Router(config-routemap abc)# router ospf
FESX424 Router(config-ospf-router)# redistribute static route-map abc
```

The commands in this example configure some static IP routes, then configure a route map and use the route map for redistributing static IP routes into OSPF.

The **ip route** commands configure the static IP routes. The **route-map** command begins configuration of a route map called “abc”. The number indicates the route map entry (called the “instance”) you are configuring. A route map can contain multiple entries. The software compares packets to the route map entries in ascending numerical order and stops the comparison once a match is found.

The **match** command in the route map matches on routes that have 5 for their metric value (cost). The **set** command changes the metric in routes that match the route map to 8.

The **redistribute static** command enables redistribution of static IP routes into OSPF, and uses route map “abc“ to control the routes that are redistributed. In this example, the route map allows a static IP route to be redistributed into OSPF only if the route has a metric of 5, and changes the metric to 8 before placing the route into the OSPF route table.

The following command shows the result of the redistribution filter. Since only one of the static IP routes configured above matches the route map, only one route is redistributed. Notice that the route’s metric is 5 before redistribution but is 8 after redistribution.

```
FESX424 Router(config-ospf-router)# show ip ospf database external extensive

Index Aging  LS ID           Router          Netmask  Metric    Flag
1     2      4.4.0.0         10.10.10.60     ffff0000 80000008  0000
```

Syntax: [no] redistribution bgp | connected | rip | static [route-map <map-name>]

The **bgp | connected | rip | static** parameter specifies the route source.

The **route-map** <map-name> parameter specifies the route map name. The following match parameters are valid for OSPF redistribution:

- **match ip address | next-hop** <acl-num>
- **match metric** <num>
- **match tag** <tag-value>

The following set parameters are valid for OSPF redistribution:

- **set ip next hop** <ip-addr>
- **set metric [+ | -]**<num> **| none**
- **set metric-type type-1 | type-2**

- **set tag** <tag-value>

NOTE: You must configure the route map before you configure a redistribution filter that uses the route map.

NOTE: When you use a route map for route redistribution, the software disregards the permit or deny action of the route map.

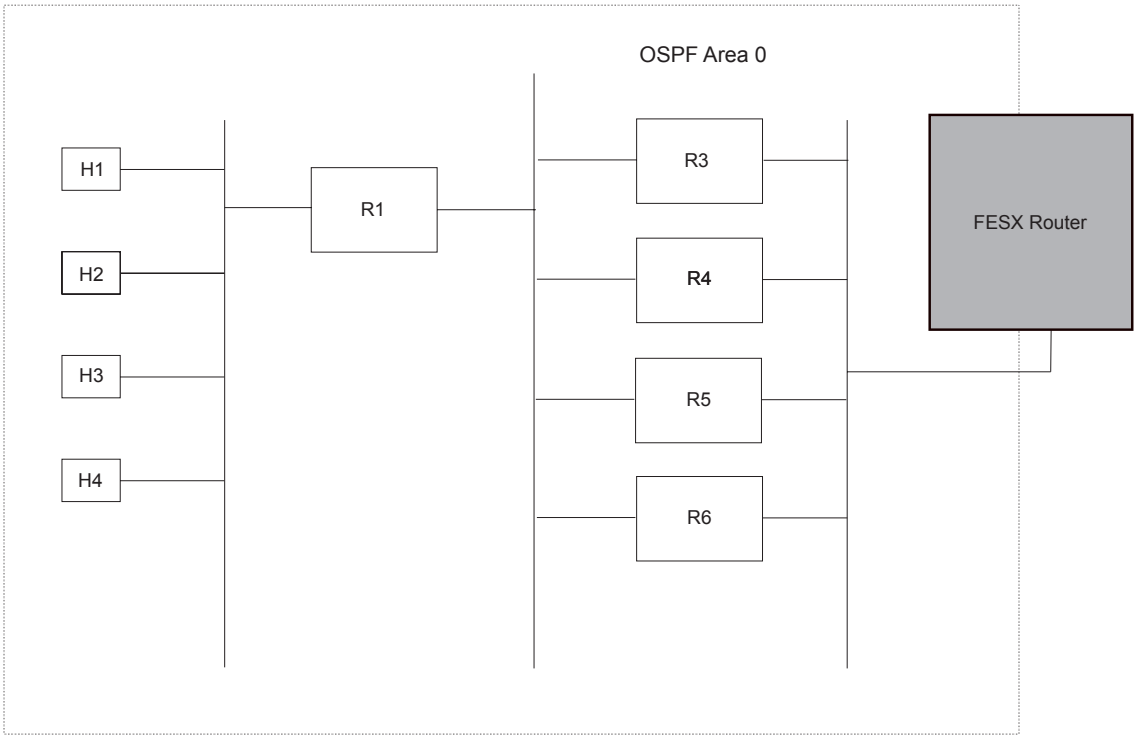
NOTE: For an external route that is redistributed into OSPF through a route map, the metric value of the route remains the same unless the metric is set by a **set metric** command inside the route map. The **default-metric** <num> command has no effect on the route. This behavior is different from a route that is redistributed without using a route map. For a route redistributed without using a route map, the metric is set by the default-metric <num> command.

Disable or Re-enable Load Sharing

Foundry routers can load share among up to eight equal-cost IP routes to a destination. By default, IP load sharing is enabled. The default is 4 equal-cost paths but you can specify from 2 – 6 paths.

The router software can use the route information it learns through OSPF to determine the paths and costs. Figure 20.8 shows an example of an OSPF network containing multiple paths to a destination (in this case, R1).

Figure 20.8 Example OSPF network with four equal-cost paths



In the example in Figure 20.8, the Foundry router has four paths to R1:

- FI->R3
- FI->R4
- FI->R5
- FI->R6

Normally, the Foundry router will choose the path to the R1 with the lower metric. For example, if R3's metric is 1400 and R4's metric is 600, the Foundry router will always choose R4.

However, suppose the metric is the same for all four routers in this example. If the costs are the same, the router now has four equal-cost paths to R1. To allow the router to load share among the equal cost routes, enable IP load sharing. The software supports four equal-cost OSPF paths by default when you enable load sharing. You can specify from 2 – 6 paths.

NOTE: The Foundry router is not source routing in these examples. The router is concerned only with the paths to the next-hop routers, not the entire paths to the destination hosts.

OSPF load sharing is enabled by default when IP load sharing is enabled. To configure IP load sharing parameters, see "Configuring IP Load Sharing" on page 16-41.

Configure External Route Summarization

When the Layer 3 Switch is an OSPF Autonomous System Boundary Router (ASBR), you can configure it to advertise one external route as an aggregate for all redistributed routes that are covered by a specified address range.

When you configure an address range, the range takes effect immediately. All the imported routes are summarized according to the configured address range. Imported routes that have already been advertised and that fall within the range are flushed out of the AS and a single route corresponding to the range is advertised.

If a route that falls within a configured address range is imported by the Layer 3 Switch, no action is taken if the Layer 3 Switch has already advertised the aggregate route; otherwise the Layer 3 Switch advertises the aggregate route. If an imported route that falls with in a configured address range is removed by the Layer 3 Switch, no action is taken if there are other imported route(s) that fall with in the same address range; otherwise the aggregate route is flushed.

You can configure up to 32 address ranges. The Layer 3 Switch sets the forwarding address of the aggregate route to zero and sets the tag to zero.

If you delete an address range, the advertised aggregate route is flushed and all imported routes that fall within the range are advertised individually.

If an external LSDB overflow condition occurs, all aggregate routes are flushed out of the AS, along with other external routes. When the Layer 3 Switch exits the external LSDB overflow condition, all the imported routes are summarized according to the configured address ranges.

NOTE: If you use redistribution filters in addition to address ranges, the Layer 3 Switch applies the redistribution filters to routes first, then applies them to the address ranges.

NOTE: If you disable redistribution, all the aggregate routes are flushed, along with other imported routes.

NOTE: This option affects only imported, type 5 external routes. A single type 5 LSA is generated and flooded throughout the AS for multiple external routes. Type 7-route redistribution is not affected by this feature. All type 7 routes will be imported (if redistribution is enabled). To summarize type 7 LSAs or exported routes, use NSSA address range summarization.

To configure a summary address for OSPF routes, enter commands such as the following:

```
FESX424 Router(config-ospf-router)# summary-address 10.1.0.0 255.255.0.0
```

The command in this example configures summary address 10.1.0.0, which includes addresses 10.1.1.0, 10.1.2.0, 10.1.3.0, and so on. For all of these networks, only the address 10.1.0.0 is advertised in external LSAs.

Syntax: summary-address <ip-addr> <ip-mask>

The <ip-addr> parameter specifies the network address.

The <ip-mask> parameter specifies the network mask.

To display the configured summary addresses, enter the following command at any level of the CLI:

```
FESX424 Router(config-ospf-router)# show ip ospf config
OSPF Redistribution Address Ranges currently defined:
Range-Address    Subnetmask
1.0.0.0          255.0.0.0
1.0.1.0          255.255.255.0
1.0.2.0          255.255.255.0
```

Syntax: show ip ospf config

Configure Default Route Origination

When the Layer 3 Switch is an OSPF Autonomous System Boundary Router (ASBR), you can configure it to automatically generate a default external route into an OSPF routing domain. This feature is called “default route origination” or “default information origination”.

By default, Foundry Layer 3 Switches do not advertise the default route into the OSPF domain. If you want the Layer 3 Switch to advertise the OSPF default route, you must explicitly enable default route origination.

When you enable OSPF default route origination, the Layer 3 Switch advertises a type 5 default route that is flooded throughout the AS (except stub areas and NSSAs). In addition, internal NSSA ASBRs advertise their default routes as translatable type 7 default routes.

The Layer 3 Switch advertises the default route into OSPF even if OSPF route redistribution is not enabled, and even if the default route is learned through an IBGP neighbor.

NOTE: Foundry Layer 3 Switches never advertise the OSPF default route, regardless of other configuration parameters, unless you explicitly enable default route origination using the following method.

If the Layer 3 Switch is an ASBR, you can use the “always” option when you enable the default route origination. The always option causes the ASBR to create and advertise a default route if it does not already have one configured.

If default route origination is enabled and you disable it, the default route originated by the Layer 3 Switch is flushed. Default routes generated by other OSPF routers are not affected. If you re-enable the feature, the feature takes effect immediately and thus does not require you to reload the software.

NOTE: The ABR (Layer 3 Switch) will not inject the default route into an NSSA by default and the command described in this section will not cause the Layer 3 Switch to inject the default route into the NSSA. To inject the default route into an NSSA, use the **area** <num> | <ip-addr> **nssa default-information-originate** command. See “Assign a Not-So-Stubby Area (NSSA)” on page 20-12.

To enable default route origination, enter the following command:

```
FESX424 Router(config-ospf-router)# default-information-originate
```

To disable the feature, enter the following command:

```
FESX424 Router(config-ospf-router)# no default-information-originate
```

Syntax: [no] default-information-originate [always] [metric <value>] [metric-type <type>]

The **always** parameter advertises the default route regardless of whether the router has a default route. This option is disabled by default.

The **metric** <value> parameter specifies a metric for the default route. If this option is not used, the default metric is used for the route.

The **metric-type** <type> parameter specifies the external link type associated with the default route advertised into the OSPF routing domain. The <type> can be one of the following:

- 1 – Type 1 external route
- 2 – Type 2 external route

If you do not use this option, the default redistribution metric type is used for the route type.

NOTE: If you specify a metric and metric type, the values you specify are used even if you do not use the **always** option.

Modify SPF Timers

The Layer 3 Switch uses the following timers when calculating the shortest path for OSPF routes:

- SPF delay – When the Layer 3 Switch receives a topology change, the software waits before it starts a Shortest Path First (SPF) calculation. By default, the software waits five seconds. You can configure the SPF delay to a value from 0 – 65535 seconds. If you set the SPF delay to 0 seconds, the software immediately begins the SPF calculation after receiving a topology change.
- SPF hold time – The Layer 3 Switch waits for a specific amount of time between consecutive SPF calculations. By default, the Layer 3 Switch waits ten seconds. You can configure the SPF hold time to a value from 0 – 65535 seconds. If you set the SPF hold time to 0 seconds, the software does not wait between consecutive SPF calculations.

You can set the delay and hold time to lower values to cause the Layer 3 Switch to change to alternate paths more quickly in the event of a route failure. Note that lower values require more CPU processing time.

You can change one or both of the timers. To do so, enter commands such as the following:

```
FESX424 Router(config-ospf-router)# timers spf 10 20
```

The command in this example changes the SPF delay to 10 seconds and changes the SPF hold time to 20 seconds.

Syntax: timers spf <delay> <hold-time>

The <delay> parameter specifies the SPF delay.

The <hold-time> parameter specifies the SPF hold time.

To set the timers back to their default values, enter a command such as the following:

```
FESX424 Router(config-ospf-router)# no timers spf 10 20
```

Modify Redistribution Metric Type

The redistribution metric type is used by default for all routes imported into OSPF unless you specify different metrics for individual routes using redistribution filters. Type 2 specifies a big metric (three bytes). Type 1 specifies a small metric (two bytes). The default value is type 2.

To modify the default value to type 1, enter the following command:

```
FESX424 Router(config-ospf-router)# metric-type type1
```

Syntax: metric-type type1 | type2

The default is **type2**.

Modify Administrative Distance

Foundry Layer 3 Switches can learn about networks from various protocols, including Border Gateway Protocol version 4 (BGP4), RIP, and OSPF. Consequently, the routes to a network may differ depending on the protocol from which the routes were learned. The default administrative distance for OSPF routes is 110. See "Changing Administrative Distances" on page 21-29 for a list of the default distances for all route sources.

The router selects one route over another based on the source of the route information. To do so, the router can use the administrative distances assigned to the sources. You can bias the Layer 3 Switch's decision by changing the default administrative distance for RIP routes.

Configuring Administrative Distance Based on Route Type

You can configure a unique administrative distance for each type of OSPF route. For example, you can use this feature to prefer a static route over an OSPF inter-area route but you also want to prefer OSPF intra-area routes to static routes.

The distance you specify influences the choice of routes when the Layer 3 Switch has multiple routes for the same network from different protocols. The Layer 3 Switch prefers the route with the lower administrative distance.

You can specify unique default administrative distances for the following route types:

- Intra-area routes
- Inter-area routes
- External routes

The default for all these OSPF route types is 110.

NOTE: This feature does not influence the choice of routes within OSPF. For example, an OSPF intra-area route is always preferred over an OSPF inter-area route, even if the intra-area route's distance is greater than the inter-area route's distance.

To change the default administrative distances for inter-area routes, intra-area routes, and external routes, enter the following command:

```
FESX424 Router(config-ospf-router)# distance external 100
FESX424 Router(config-ospf-router)# distance inter-area 90
FESX424 Router(config-ospf-router)# distance intra-area 80
```

Syntax: distance external | inter-area | intra-area <distance>

The **external | inter-area | intra-area** parameter specifies the route type for which you are changing the default administrative distance.

The <distance> parameter specifies the new distance for the specified route type. Unless you change the distance for one of the route types using commands such as those shown above, the default is 110.

To reset the administrative distance to its system default (110), enter a command such as the following:

```
FESX424 Router(config-ospf-router)# no distance external 100
```

Configure OSPF Group Link State Advertisement (LSA) Pacing

The Layer 3 Switch paces LSA refreshes by delaying the refreshes for a specified time interval instead of performing a refresh each time an individual LSA's refresh timer expires. The accumulated LSAs constitute a group, which the Layer 3 Switch refreshes and sends out together in one or more packets.

The pacing interval, which is the interval at which the Layer 3 Switch refreshes an accumulated group of LSAs, is configurable to a range from 10 – 1800 seconds (30 minutes). The default is 240 seconds (four minutes). Thus, every four minutes, the Layer 3 Switch refreshes the group of accumulated LSAs and sends the group together in the same packet(s).

Usage Guidelines

The pacing interval is inversely proportional to the number of LSAs the Layer 3 Switch is refreshing and aging. For example, if you have approximately 10,000 LSAs, decreasing the pacing interval enhances performance. If you have a very small database (40 – 100 LSAs), increasing the pacing interval to 10 – 20 minutes might enhance performance slightly.

Changing the LSA Pacing Interval

To change the LSA pacing interval to two minutes (120 seconds), enter the following command:

```
FESX424 Router(config-ospf-router)# timers lsa-group-pacing 120
```

Syntax: [no] timers lsa-group-pacing <secs>

The <secs> parameter specifies the number of seconds and can be from 10 – 1800 (30 minutes). The default is 240 seconds (four minutes).

To restore the pacing interval to its default value, enter the following command:

```
FESX424 Router(config-ospf-router)# no timers lsa-group-pacing
```

Modify OSPF Traps Generated

OSPF traps as defined by RFC 1850 are supported on Foundry routers. OSPF trap generation is enabled on the router, by default.

When using the CLI, you can disable all or specific OSPF trap generation by entering the following CLI command:

```
FESX424 Router(config-ospf-router)# no snmp-server trap ospf
```

To later re-enable the trap feature, enter **snmp-server trap ospf**.

To disable a specific OSPF trap, enter the command as **no snmp-server trap ospf** <ospf-trap>.

These commands are at the OSPF router Level of the CLI.

Here is a summary of OSPF traps supported on Foundry routers, their corresponding CLI commands, and their associated MIB objects from RFC 1850:

- **interface-state-change-trap** – [MIB object: OspfIfstateChange]
- **virtual-interface-state-change-trap** – [MIB object: OspfVirtIfStateChange
- **neighbor-state-change-trap** – [MIB object:ospfNbrStateChange]
- **virtual-neighbor-state-change-trap** – [MIB object: ospfVirtNbrStateChange]
- **interface-config-error-trap** – [MIB object: ospfIfConfigError]
- **virtual-interface-config-error-trap** – [MIB object: ospfVirtIfConfigError]
- **interface-authentication-failure-trap** – [MIB object: ospfIfAuthFailure]
- **virtual-interface-authentication-failure-trap** – [MIB object: ospfVirtIfAuthFailure]
- **interface-receive-bad-packet-trap** – [MIB object: ospfIfrxBadPacket]
- **virtual-interface-receive-bad-packet-trap** – [MIB object: ospfVirtIfRxBadPacket]
- **interface-retransmit-packet-trap** – [MIB object: ospfTxRetransmit]
- **virtual-interface-retransmit-packet-trap** – [MIB object: ospfVirtIfTxRetransmit]
- **originate-lsa-trap** – [MIB object: ospfOriginateLsa]
- **originate-maxage-lsa-trap** – [MIB object: ospfMaxAgeLsa]
- **link-state-database-overflow-trap** – [MIB object: ospfLsdbOverflow]
- **link-state-database-approaching-overflow-trap** – [MIB object: ospfLsdbApproachingOverflow

EXAMPLE:

To stop an OSPF trap from being collected, use the CLI command: **no trap** <ospf-trap>, at the Router OSPF level of the CLI. To disable reporting of the neighbor-state-change-trap, enter the following command:

```
FESX424 Router(config-ospf-router)# no trap neighbor-state-change-trap
```

EXAMPLE:

To reinstate the trap, enter the following command:

```
FESX424 Router(config-ospf-router)# trap neighbor-state-change-trap
```

Syntax: [no] snmp-server trap ospf <ospf-trap>

Modify OSPF Standard Compliance Setting

Foundry routers are configured, by default, to be compliant with the RFC 1583 OSPF V2 specification.

To configure a router to operate with the latest OSPF standard, RFC 2178, enter the following commands:

```
FESX424 Router(config)# router ospf
FESX424 Router(config-ospf-router)# no rfc1583-compatibility
```

Syntax: [no] rfc1583-compatibility

Modify Exit Overflow Interval

If a database overflow condition occurs on a router, the router eliminates the condition by removing entries that originated on the router. The exit overflow interval allows you to set how often a Layer 3 Switch checks to see if the overflow condition has been eliminated. The default value is 0. The range is 0 – 86400 seconds (24 hours). If the configured value of the database overflow interval is zero, then the router never leaves the database overflow condition.

NOTE: FastIron devices dynamically allocate OSPF memory as needed. See “Dynamic OSPF Memory” on page 20-8.

To modify the exit overflow interval to 60 seconds, enter the following command:

```
FESX424 Router(config-ospf-router)# data-base-overflow-interval 60
```

Syntax: database-overflow-interval <value>

The <value> can be from 0 – 86400 seconds. The default is 0 seconds.

Configuring an OSPF Point-to-Point Link

In an OSPF point-to-point link, a direct Layer 3 connection exists between a single pair of OSPF routers, without the need for Designated and Backup Designated routers. In a point-to-point link, neighboring routers become adjacent whenever they can communicate directly. In contrast, in broadcast and non-broadcast multi-access (NBMA) networks, the Designated Router and the Backup Designated Router become adjacent to all other routers attached to the network.

Configuration Notes and Limitations

- This feature is supported on FESX devices running software release 02.2.00 or later.
- This feature is supported on Gigabit Ethernet and 10-Gigabit Ethernet interfaces.
- This feature is supported on physical interfaces. It is not supported on virtual interfaces.
- Foundry supports numbered point-to-point networks, meaning the OSPF router must have an IP interface address which uniquely identifies the router over the network. Foundry does not support unnumbered point-to-point networks.

Configuring an OSPF Point-to-Point Link

To configure an OSPF point-to-point link, enter commands such as the following:

```
FastIron SuperX Switch(config)# interface eth 1/5
FastIron SuperX Switch(config-if-1/5)# ip ospf network point-to-point
```

This command configures an OSPF point-to-point link on Interface 5 in slot 1.

Syntax: [no] ip ospf network point-to-point

Viewing Configured OSPF Point-to-Point Links

See “Displaying OSPF Neighbor Information” on page 20-41 and “Displaying OSPF Interface Information” on page 20-43.

Specify Types of OSPF Syslog Messages to Log

You can specify which kinds of OSPF-related Syslog messages are logged. By default, the only OSPF messages that are logged are those indicating possible system errors. If you want other kinds of OSPF messages to be logged, you can configure the Foundry device to log them.

For example, to specify that all OSPF-related Syslog messages be logged, enter the following commands.

```
FESX424 Router(config)# router ospf
FESX424 Router(config-ospf-router)# log all
```

Syntax: [no] log all | adjacency | bad_packet [checksum] | database | memory | retransmit

The **log** command has the following options:

The **all** option causes all OSPF-related Syslog messages to be logged. If you later disable this option with the **no log all** command, the OSPF logging options return to their default settings.

The **adjacency** option logs essential OSPF neighbor state changes, especially on error cases. This option is disabled by default.

The **bad_packet checksum** option logs all OSPF packets that have checksum errors. This option is enabled by default.

The **bad_packet** option logs all other bad OSPF packets. This option is disabled by default.

The **database** option logs OSPF LSA-related information. This option is disabled by default.

The **memory** option logs abnormal OSPF memory usage. This option is enabled by default.

The **retransmit** option logs OSPF retransmission activities. This option is disabled by default.

Displaying OSPF Information

You can use CLI commands and Web management options to display the following OSPF information:

- Trap, area, and interface information – see “Displaying General OSPF Configuration Information” on page 20-38.
- CPU utilization statistics – see “Displaying CPU Utilization Statistics” on page 20-39.
- Area information – see “Displaying OSPF Area Information” on page 20-40.
- Neighbor information – see “Displaying OSPF Neighbor Information” on page 20-41.
- Interface information – see “Displaying OSPF Interface Information” on page 20-43.
- Route information – see “Displaying OSPF Route Information” on page 20-44.
- External link state information – see “Displaying OSPF External Link State Information” on page 20-46.
- Link state information – see “Displaying OSPF Link State Information” on page 20-47.
- Virtual Neighbor information – see “Displaying OSPF Virtual Neighbor Information” on page 20-49.

- Virtual Link information – see “Displaying OSPF Virtual Link Information” on page 20-49.
- ABR and ASBR information – see “Displaying OSPF ABR and ASBR Information” on page 20-49.
- Trap state information – see “Displaying OSPF Trap Status” on page 20-49.

Displaying General OSPF Configuration Information

To display general OSPF configuration information, enter the following command at any CLI level:

```
FESX424 Router> show ip ospf config

Router OSPF: Enabled
Redistribution: Disabled
Default OSPF Metric: 10
OSPF Redistribution Metric: Type2

OSPF External LSA Limit: 25000

OSPF Database Overflow Interval: 0

RFC 1583 Compatibility: Enabled

Router id: 207.95.11.128

Interface State Change Trap:                        Enabled
Virtual Interface State Change Trap:                Enabled
Neighbor State Change Trap:                         Enabled
Virtual Neighbor State Change Trap:                 Enabled
Interface Configuration Error Trap:                 Enabled
Virtual Interface Configuration Error Trap:         Enabled
Interface Authentication Failure Trap:              Enabled
Virtual Interface Authentication Failure Trap:      Enabled
Interface Receive Bad Packet Trap:                  Enabled
Virtual Interface Receive Bad Packet Trap:          Enabled
Interface Retransmit Packet Trap:                   Enabled
Virtual Interface Retransmit Packet Trap:           Enabled
Originate LSA Trap:                                 Enabled
Originate MaxAge LSA Trap:                          Enabled
Link State Database Overflow Trap:                  Enabled
Link State Database Approaching Overflow Trap:      Enabled

OSPF Area currently defined:
Area-ID          Area-Type Cost
0                normal    0

OSPF Interfaces currently defined:
Ethernet Interface: 3/1-3/2
ip ospf md5-authentication-key-activation-wait-time 300
ip ospf cost 0
ip ospf area 0

Ethernet Interface: v1
ip ospf md5-authentication-key-activation-wait-time 300
ip ospf cost 0
ip ospf area 0
```

Syntax: show ip ospf config

Displaying CPU Utilization Statistics

You can display CPU utilization statistics for OSPF and other IP protocols.

To display CPU utilization statistics for OSPF for the previous one-second, one-minute, five-minute, and fifteen-minute intervals, enter the following command at any level of the CLI:

```
FESX424 Router# show process cpu
Process Name    5Sec(%)    1Min(%)    5Min(%)    15Min(%)    Runtime(ms)
ARP               0.01       0.03       0.09       0.22              9
BGP               0.04       0.06       0.08       0.14             13
GVRP              0.00       0.00       0.00       0.00              0
ICMP              0.00       0.00       0.00       0.00              0
IP                0.00       0.00       0.00       0.00              0
OSPF              0.03       0.06       0.09       0.12             11
RIP               0.00       0.00       0.00       0.00              0
STP               0.00       0.00       0.00       0.00              0
VRRP              0.00       0.00       0.00       0.00              0
```

If the software has been running less than 15 minutes (the maximum interval for utilization statistics), the command indicates how long the software has been running. Here is an example:

```
FESX424 Router# show process cpu
The system has only been up for 6 seconds.
Process Name    5Sec(%)    1Min(%)    5Min(%)    15Min(%)    Runtime(ms)
ARP               0.01       0.00       0.00       0.00              0
BGP               0.00       0.00       0.00       0.00              0
GVRP              0.00       0.00       0.00       0.00              0
ICMP              0.01       0.00       0.00       0.00              1
IP                0.00       0.00       0.00       0.00              0
OSPF              0.00       0.00       0.00       0.00              0
RIP               0.00       0.00       0.00       0.00              0
STP               0.00       0.00       0.00       0.00              0
VRRP              0.00       0.00       0.00       0.00              0
```

To display utilization statistics for a specific number of seconds, enter a command such as the following:

```
FESX424 Router# show process cpu 2
Statistics for last 1 sec and 80 ms
Process Name    Sec(%)    Time(ms)
ARP               0.00           0
BGP               0.00           0
GVRP              0.00           0
ICMP              0.01           1
IP                0.00           0
OSPF              0.00           0
RIP               0.00           0
STP               0.01           0
VRRP              0.00           0
```

When you specify how many seconds' worth of statistics you want to display, the software selects the sample that most closely matches the number of seconds you specified. In this example, statistics are requested for the previous two seconds. The closest sample available is actually for the previous 1 second plus 80 milliseconds.

Syntax: show process cpu [<num>]

The <num> parameter specifies the number of seconds and can be from 1 – 900. If you use this parameter, the command lists the usage statistics only for the specified number of seconds. If you do not use this parameter, the

command lists the usage statistics for the previous one-second, one-minute, five-minute, and fifteen-minute intervals.

Displaying OSPF Area Information

To display OSPF area information, enter the following command at any CLI level:

```
FESX424 Router> show ip ospf area
Indx  Area       Type   Cost   SPFR ABR ASBR LSA Chksum(Hex)
1    0.0.0.0     normal  0     1     0     0    1    0000781f
2   192.147.60.0 normal  0     1     0     0    1    0000fee6
3   192.147.80.0 stub    1     1     0     0    2    000181cd
```

Syntax: show ip ospf area [<area-id>] | [<num>]

The <area-id> parameter shows information for the specified area.

The <num> parameter displays the entry that corresponds to the entry number you enter. The entry number identifies the entry's position in the area table.

This display shows the following information.

Table 20.2: CLI Display of OSPF Area Information

This Field...	Displays...
Indx	The row number of the entry in the router's OSPF area table.
Area	The area number.
Type	The area type, which can be one of the following: • nssa • normal • stub
Cost	The area's cost.
SPFR	The SPFR value.
ABR	The ABR number.
ASBR	The ABSR number.
LSA	The LSA number.
Chksum(Hex)	The checksum for the LSA packet. The checksum is based on all the fields in the packet except the age field. The Layer 3 Switch uses the checksum to verify that the packet is not corrupted.

Displaying OSPF Neighbor Information

To display OSPF neighbor information, enter the following command at any CLI level:

```
FESX424 Router> show ip ospf neighbor

Port Address          Pri State      Neigh Address   Neigh ID
8    212.76.7.251     1   full       212.76.7.200    173.35.1.220
```

To display detailed OSPF neighbor information, enter the following command at any CLI level:

```
FESX424 Router# show ip ospf neighbor detail

Port        Address         Pri State      Neigh Address   Neigh ID     Ev Op Cnt
9/1         20.2.0.2        1   FULL/DR    20.2.0.1        2.2.2.2       6  2  0
    Second-to-dead:39
10/1        20.3.0.2        1   FULL/BDR   20.3.0.1        3.3.3.3       5  2  0
    Second-to-dead:36
1/1-1/8     23.5.0.1        1   FULL/DR    23.5.0.2        16.16.16.16   6  2  0
    Second-to-dead:33
2/1-2/2     23.2.0.1        1   FULL/DR    23.2.0.2        15.15.15.15   6  2  0
    Second-to-dead:33
```

Syntax: show ip ospf neighbor [router-id <ip-addr>] | [<num>] | [detail]

The **router-id** <ip-addr> parameter displays only the neighbor entries for the specified router.

The <num> parameter displays only the entry in the specified index position in the neighbor table. For example, if you enter “1”, only the first entry in the table is displayed.

The **detail** parameter displays detailed information about the neighbor routers.

These displays show the following information.

Table 20.3: CLI Display of OSPF Neighbor Information

Field	Description
Port	The port through which the Layer 3 Switch is connected to the neighbor. The port on which an OSPF point-to-point link is configured.
Address	The IP address of this Layer 3 Switch’s interface with the neighbor.

Pri	The OSPF priority of the neighbor.
	• For multi-access networks, the priority is used during election of the Designated Router (DR) and Backup designated Router (BDR).
	• For point-to-point links, this field shows one of the following values:
	  • 1 = point-to-point link
	  • 3 = point-to-point link with assigned subnet

Table 20.3: CLI Display of OSPF Neighbor Information (Continued)

Field	Description
State	The state of the conversation between the Layer 3 Switch and the neighbor. This field can have one of the following values: • Down – The initial state of a neighbor conversation. This value indicates that there has been no recent information received from the neighbor. • Attempt – This state is only valid for neighbors attached to non-broadcast networks. It indicates that no recent information has been received from the neighbor. • Init – A Hello packet has recently been seen from the neighbor. However, bidirectional communication has not yet been established with the neighbor. (The router itself did not appear in the neighbor's Hello packet.) All neighbors in this state (or higher) are listed in the Hello packets sent from the associated interface. • 2-Way – Communication between the two routers is bidirectional. This is the most advanced state before beginning adjacency establishment. The Designated Router and Backup Designated Router are selected from the set of neighbors in the 2-Way state or greater. • ExStart – The first step in creating an adjacency between the two neighboring routers. The goal of this step is to decide which router is the master, and to decide upon the initial Database Description (DD) sequence number. Neighbor conversations in this state or greater are called adjacencies. • Exchange – The router is describing its entire link state database by sending Database Description packets to the neighbor. Each Database Description packet has a DD sequence number, and is explicitly acknowledged. Only one Database Description packet can be outstanding at any time. In this state, Link State Request packets can also be sent asking for the neighbor's more recent advertisements. All adjacencies in Exchange state or greater are used by the flooding procedure. In fact, these adjacencies are fully capable of transmitting and receiving all types of OSPF routing protocol packets. • Loading – Link State Request packets are sent to the neighbor asking for the more recent advertisements that have been discovered (but not yet received) in the Exchange state. • Full – The neighboring routers are fully adjacent. These adjacencies will now appear in router links and network link advertisements.
Neigh Address	The IP address of the neighbor. For point-to-point links, the value is as follows: • If the **Pri** field is "1", this value is the IP address of the neighbor router's interface. • If the **Pri** field is "3", this is the subnet IP address of the neighbor router's interface.
Neigh ID	The neighbor router's ID.
Ev	The number of times the neighbor's state changed.
Opt	The sum of the option bits in the Options field of the Hello packet. This information is used by Foundry technical support. See Section A.2 in RFC 2178 for information about the Options field in Hello packets.
Cnt	The number of LSAs that were retransmitted.
Second-to-dead	The amount of time the Foundry device will wait for a HELLO message from each OSPF neighbor before assuming the neighbor is dead.

Displaying OSPF Interface Information

To display OSPF interface information, enter the following command at any CLI level:

```
FastIron SuperX Router# show ip ospf interface 192.168.1.1

Ethernet 2/1,OSPF enabled
     IP Address 192.168.1.1, Area 0
     OSPF state ptr2ptr, Pri 1, Cost 1, Options 2, Type pt-2-pt Events 1
     Timers(sec): Transit 1, Retrans 5, Hello 10, Dead 40
     DR:  Router ID 0.0.0.0           Interface Address 0.0.0.0
     BDR: Router ID 0.0.0.0           Interface Address 0.0.0.0
     Neighbor Count = 0, Adjacent Neighbor Count= 1
     Neighbor: 2.2.2.2
     Authentication-Key:None
     MD5 Authentication: Key None, Key-Id None, Auth-change-wait-time 300
```

Syntax: show ip ospf interface [<ip-addr>]

The <ip-addr> parameter displays the OSPF interface information for the specified IP address.

The following table defines the highlighted fields shown in the above example output of the **show ip ospf interface** command.

Table 20.4: Output of the show ip ospf interface command

This field	Displays
IP Address	The IP address of the interface.
OSPF state	ptr2ptr (point to point)
Pri	The link ID as defined in the router-LSA. This value can be one of the following: 1 = point-to-point link 3 = point-to-point link with an assigned subnet
Cost	The configured output cost for the interface.
Options	OSPF Options (Bit7 - Bit0): • unused:1 • opaque:1 • summary:1 • dont_propagate:1 • nssa:1 • multicast:1 • externals:1 • tos:1

Table 20.4: Output of the show ip ospf interface command

This field	Displays
Type	The area type, which can be one of the following: • Broadcast = 0x01 • NBMA = 0x02 • Point to Point = 0x03 • Virtual Link = 0x04 • Point to Multipoint = 0x05
Events	OSPF Interface Event: • Interface_Up = 0x00 • Wait_Timer = 0x01 • Backup_Seen = 0x02 • Neighbor_Change = 0x03 • Loop_Indication = 0x04 • Unloop_Indication = 0x05 • Interface_Down = 0x06 • Interface_Passive = 0x07
Adjacent Neighbor Count	The number of adjacent neighbor routers.
Neighbor:	The neighbor router's ID.

Displaying OSPF Route Information

To display OSPF route information for the router, enter the following command at any CLI level:

```
FastIron SuperX Router> show ip ospf routes

Index Destination     Mask            Path_Cost Type2_Cost Path_Type
1     212.95.7.0      255.255.255.0   1         0          Intra
      Adv_Router      Link_State      Dest_Type State      Tag       Flags
      173.35.1.220    212.95.7.251    Network   Valid      00000000  7000
      Paths Out_Port  Next_Hop        Type      Arp_Index  State
      1     5/6       209.95.7.250    OSPF      8          84 00

Index Destination     Mask            Path_Cost Type2_Cost Path_Type
2     11.3.63.0       255.255.255.0   11        0          Inter
      Adv_Router      Link_State      Dest_Type State      Tag       Flags
      209.95.7.250    11.3.63.0       Network   Valid      00000000  0000
      Paths Out_Port  Next_Hop        Type      Arp_Index  State
      1     5/6       209.95.7.250    OSPF      8          84 00
```

Syntax: show ip ospf routes [<ip-addr>]

The <ip-addr> parameter specifies a destination IP address. If you use this parameter, only the route entries for that destination are shown.

This display shows the following information.

Table 20.5: CLI Display of OSPF Route Information

This Field...	Displays...
Index	The row number of the entry in the router's OSPF route table.
Destination	The IP address of the route's destination.
Mask	The network mask for the route.
Path_Cost	The cost of this route path. (A route can have multiple paths. Each path represents a different exit port for the Layer 3 Switch.)
Type2_Cost	The type 2 cost of this path.
Path_Type	The type of path, which can be one of the following: • Inter – The path to the destination passes into another area. • Intra – The path to the destination is entirely within the local area. • External1 – The path to the destination is a type 1 external route. • External2 – The path to the destination is a type 2 external route.
Adv_Router	The OSPF router that advertised the route to this Foundry Layer 3 Switch.
Link-State	The link state from which the route was calculated.
Dest_Type	The destination type, which can be one of the following: • ABR – Area Border Router • ASBR – Autonomous System Boundary Router • Network – the network
State	The route state, which can be one of the following: • Changed • Invalid • Valid This information is used by Foundry technical support.
Tag	The external route tag.
Flags	State information for the route entry. This information is used by Foundry technical support.
Paths	The number of paths to the destination.
Out_Port	The router port through which the Layer 3 Switch reaches the next hop for this route path.
Next_Hop	The IP address of the next-hop router for this path.

Table 20.5: CLI Display of OSPF Route Information (Continued)

This Field...	Displays...
Type	The route type, which can be one of the following: • OSPF • Static Replaced by OSPF
Arp_Index	The index position in the ARP table of the ARP entry for this path's IP address.
State	State information for the path. This information is used by Foundry technical support.

Displaying the Routes that Have Been Redistributed into OSPF

You can display the routes that have been redistributed into OSPF. To display the redistributed routes, enter the following command at any level of the CLI:

```
FESX424 Router# show ip ospf redistribute route
  4.3.0.0 255.255.0.0 static
  3.1.0.0 255.255.0.0 static
  10.11.61.0 255.255.255.0 connected
  4.1.0.0 255.255.0.0 static
```

In this example, four routes have been redistributed. Three of the routes were redistributed from static IP routes and one route was redistributed from a directly connected IP route.

Syntax: show ip ospf redistribute route [<ip-addr> <ip-mask>]

The <ip-addr> <ip-mask> parameter specifies a network prefix and network mask. Here is an example:

```
FESX424 Router# show ip ospf redistribute route 3.1.0.0 255.255.0.0
  3.1.0.0 255.255.0.0 static
```

Displaying OSPF External Link State Information

To display external link state information, enter the following command at any CLI level:

```
FESX424 Router> show ip ospf database external-link-state

Ospf ext link-state by router ID 130.130.130.241 are in the following:

Area ID         Aging  LS ID           Router          Seq(hex) Chksum   Type
0.0.0.0         279    130.132.75.48   130.130.130.241 80000004 00000ace EXTR
0.0.0.0         278    130.132.88.112  130.130.130.241 80000004 0000f793 EXTR
0.0.0.0         279    130.132.81.208  130.130.130.241 80000004 000081b0 EXTR
0.0.0.0         284    130.132.46.224  130.130.130.241 80000004 000063e1 EXTR
0.0.0.0         285    130.132.40.64   140.140.140.243 80000004 0000ebff EXTR
0.0.0.0         286    130.132.33.160  150.150.150.245 80000004 0000751d EXTR
0.0.0.0         296    130.131.241.16  150.150.150.245 80000004 00002e25 EXTR
```


Syntax: show ip ospf database external-link-state [advertise <num>] | [extensive] | [link-state-id <ip-addr>] | [router-id <ip-addr>] | [sequence-number <num(Hex)>] | [status <num>]

The **advertise** <num> parameter displays the hexadecimal data in the specified LSA packet. The <num> parameter identifies the LSA packet by its position in the router's External LSA table. To determine an LSA packet's position in the table, enter the **show ip ospf external-link-state** command to display the table. See "Displaying the Data in an LSA" on page 20-48 for an example.

The **extensive** option displays the LSAs in decrypted format.

NOTE: You cannot use the **extensive** option in combination with other display options. The entire database is displayed.

The **link-state-id** <ip-addr> parameter displays the External LSAs for the LSA source specified by <IP-addr>.

The **router-id** <ip-addr> parameter shows the External LSAs for the specified OSPF router.

The **sequence-number** <num(Hex)> parameter displays the External LSA entries for the specified hexadecimal LSA sequence number.

The **status** <num> option shows status information.

This display shows the following information.

Table 20.6: CLI Display of OSPF External Link State Information

This Field...	Displays...
Area ID	The OSPF area the router is in.
Aging	The age of the LSA, in seconds.
LS ID	The ID of the link-state advertisement from which the Layer 3 Switch learned this route.
Router	The router IP address.
Seq(hex)	The sequence number of the LSA. The OSPF neighbor that sent the LSA stamps it with a sequence number to enable the Layer 3 Switch and other OSPF routers to determine which LSA for a given route is the most recent.
Chksum	A checksum for the LSA packet. The checksum is based on all the fields in the packet except the age field. The Layer 3 Switch uses the checksum to verify that the packet is not corrupted.
Type	The route type, which is always EXTR (external).

Displaying OSPF Link State Information

To display link state information, enter the following command at any CLI level:

```
FESX424 Router> show ip ospf database link-state
```

Syntax: show ip ospf database link-state [advertise <num>] | [asbr] | [extensive] | [link-state-id <ip-addr>] | [network] | [nssa] | [opaque-area] | [router] | [router-id <ip-addr>] | [sequence-number <num(Hex)>] | [status <num>] | [summary]

The **advertise** <num> parameter displays the hexadecimal data in the specified LSA packet. The <num> parameter identifies the LSA packet by its position in the router's External LSA table. To determine an LSA

packet's position in the table, enter the **show ip ospf external-link-state** command to display the table. See "Displaying the Data in an LSA" on page 20-48 for an example.

The **asbr** option shows ASBR information.

The **extensive** option displays the LSAs in decrypted format.

NOTE: You cannot use the **extensive** option in combination with other display options. The entire database is displayed.

The **link-state-id** <ip-addr> parameter displays the External LSAs for the LSA source specified by <IP-addr>.

The **network** option shows network information.

The **nssa** option shows network information.

The **opaque-area** option shows information for opaque areas.

The **router-id** <ip-addr> parameter shows the External LSAs for the specified OSPF router.

The **sequence-number** <num(Hex)> parameter displays the External LSA entries for the specified hexadecimal LSA sequence number.

The **status** <num> option shows status information.

The **summary** option shows summary information.

Displaying the Data in an LSA

You can use the CLI to display the data the Layer 3 Switch received in a specific External LSA packet or other type of LSA packet. For example, to display the LSA data in entry 3 in the External LSA table, enter the following command:

```
FESX424 Router> show ip ospf database external-link-state advertise 3
05 84 02 05 82 83 0d 60 82 82 82 f1 80 00 00 02 e4 05
00 24 ff ff ff f0 80 00 00 0a 00 00 00 00 00 00 00 00
```

Syntax: show ip ospf database external-link-state [advertise <num>] | [link-state-id <ip-addr>] | [router-id <ip-addr>] | [sequence-number <num(Hex)>] | [status <num>]

To determine an external LSA's or other type of LSA's index number, enter one of the following commands to display the appropriate LSA table:

- **show ip ospf database link-state advertise** <num> – This command displays the data in the packet for the specified LSA.
- **show ip ospf database external-link-state advertise** <num> – This command displays the data in the packet for the specified external LSA.

For example, to determine an external LSA's index number, enter the following command:

```
FESX424 Router> show ip ospf external-link-state
Index Aging  LS ID           Router          Seq(hex) Chksum
1     1332   130.132.81.208  130.130.130.241 80000002 000085ae
2     1325   130.132.116.192 130.130.130.241 80000002 0000a37d
3     1330   130.132.88.112  130.130.130.241 80000002 0000fb91
4     1333   130.132.75.48   130.130.130.241 80000002 00000ecc
5     1338   130.132.46.224  130.130.130.241 80000002 000067df
```

additional entries omitted for brevity...

Displaying OSPF Virtual Neighbor Information

To display OSPF virtual neighbor information, enter the following command at any CLI level:

```
FESX424 Router> show ip ospf virtual-neighbor
```

Syntax: show ip ospf virtual-neighbor [<num>]

The <num> parameter displays the table beginning at the specified entry number.

Displaying OSPF Virtual Link Information

To display OSPF virtual link information, enter the following command at any CLI level:

```
FESX424 Router> show ip ospf virtual-link
```

Syntax: show ip ospf virtual-link [<num>]

The <num> parameter displays the table beginning at the specified entry number.

Displaying OSPF ABR and ASBR Information

To display OSPF ABR and ASBR information, enter the following command at any CLI level:

```
FESX424 Router> show ip ospf border-routers
```

Syntax: show ip ospf border-routers [<ip-addr>]

The <ip-addr> parameter displays the ABR and ASBR entries for the specified IP address.

Displaying OSPF Trap Status

All traps are enabled by default when you enable OSPF. To disable or re-enable an OSPF trap, see “Modify OSPF Traps Generated” on page 20-35.

To display the state of each OSPF trap, enter the following command at any CLI level:

```
FESX424 Router> show ip ospf trap

Interface State Change Trap:                        Enabled
Virtual Interface State Change Trap:                Enabled
Neighbor State Change Trap:                         Enabled
Virtual Neighbor State Change Trap:                 Enabled
Interface Configuration Error Trap:                 Enabled
Virtual Interface Configuration Error Trap:         Enabled
Interface Authentication Failure Trap:              Enabled
Virtual Interface Authentication Failure Trap:      Enabled
Interface Receive Bad Packet Trap:                  Enabled
Virtual Interface Receive Bad Packet Trap:          Enabled
Interface Retransmit Packet Trap:                   Enabled
Virtual Interface Retransmit Packet Trap:           Enabled
Originate LSA Trap:                                 Enabled
Originate MaxAge LSA Trap:                          Enabled
Link State Database Overflow Trap:                  Enabled
Link State Database Approaching Overflow Trap:      Enabled
```

Syntax: show ip ospf trap

Chapter 21
Configuring BGP4

This chapter provides details on how to configure ***Border Gateway Protocol version 4 (BGP4)*** on Foundry products using the CLI.

BGP4 is supported in the following configurations:

- FESX Layer 3 switches running software release 02.1.01 or later
- FSX Layer 3 switches running software release 02.2.00 or later

This chapter contains the following information:

Table 21.1: Chapter Contents

Table 21.1: Chapter Contents

BGP4 is described in RFC 1771. The Foundry implementation fully complies with RFC 1771. The Foundry BGP4 implementation also supports the following RFCs:

- RFC 1745 (OSPF Interactions)
- RFC 1997 (BGP Communities Attributes)
- RFC 2385 (TCP MD5 Signature Option)
- RFC 2439 (Route Flap Dampening)
- RFC 2796 (Route Reflection)
- RFC 2842 (Capability Advertisement)
- RFC 3065 (BGP4 Confederations)

To display BGP4 configuration information and statistics, see “Displaying BGP4 Information” on page 21-65.

This chapter shows the commands you need in order to configure the Foundry Layer 3 Switch for BGP4. For a detailed list of all CLI commands, including syntax and possible values, see the *Foundry Switch and Router Command Line Interface Reference*.

NOTE: Your Layer 3 Switch’s management module must have 32MB or higher to run BGP4.

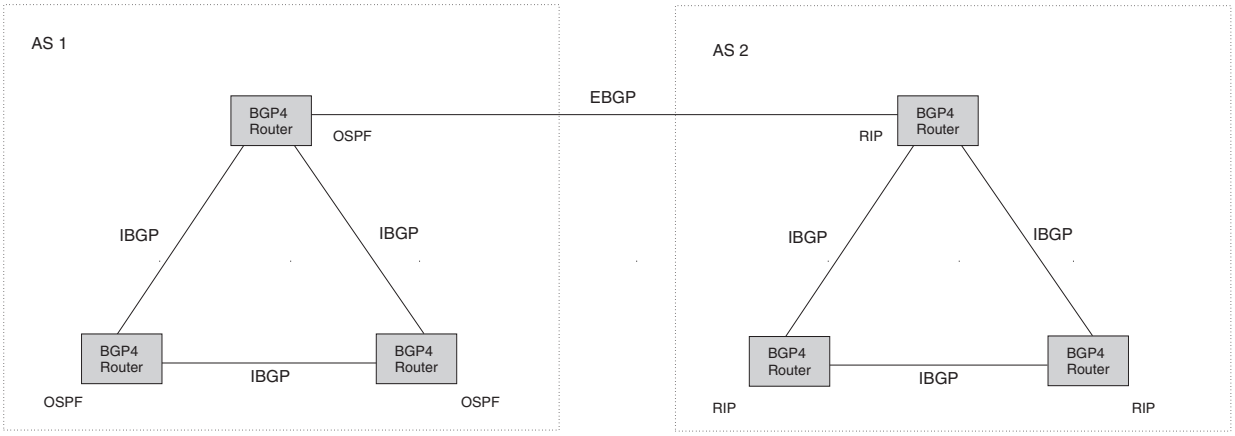
Overview of BGP4

BGP4 is the standard Exterior Gateway Protocol (EGP) used on the Internet to route traffic between ***Autonomous Systems (AS)*** and to maintain loop-free routing. An autonomous system is a collection of networks that share the same routing and administration characteristics. For example, a corporate intranet consisting of several networks under common administrative control might be considered an AS. The networks in an AS can but do not need to run the same routing protocol to be in the same AS, nor do they need to be geographically close.

Routers within an AS can use different Interior Gateway Protocols (IGPs) such as RIP and OSPF to communicate with one another. However, for routers in different ASs to communicate, they need to use an EGP. BGP4 is the standard EGP used by Internet routers and therefore is the EGP implemented on Foundry Layer 3 Switches.

Figure 21.1 on page 21-3 shows a simple example of two BGP4 ASs. Each AS contains three BGP4 routers. All of the BGP4 routers within an AS communicate using IBGP. BGP4 routers communicate with other ASs using EBGP. Notice that each of the routers also is running an Interior Gateway Protocol (IGP). The routers in AS1 are running OSPF and the routers in AS2 are running RIP. Foundry Layer 3 Switches can be configured to redistribute routes among BGP4, RIP, and OSPF. They also can redistribute static routes.

Figure 21.1 Example BGP4 ASs



Relationship Between the BGP4 Route Table and the IP Route Table

The Foundry Layer 3 Switch's BGP4 route table can have multiple routes to the same destination, which are learned from different BGP4 neighbors. A BGP4 neighbor is another router that also is running BGP4. BGP4 neighbors communicate using Transmission Control Protocol (TCP) port 179 for BGP communication. When you configure the Foundry Layer 3 Switch for BGP4, one of the configuration tasks you perform is to identify the Layer 3 Switch's BGP4 neighbors.

Although a router's BGP4 route table can have multiple routes to the same destination, the BGP4 protocol evaluates the routes and chooses only one of the routes to send to the IP route table. The route that BGP4 chooses and sends to the IP route table is the ***preferred route*** and will be used by the Foundry Layer 3 Switch. If the preferred route goes down, BGP4 updates the route information in the IP route table with a new BGP4 preferred route.

NOTE: If IP load sharing is enabled and you enable multiple equal-cost paths for BGP4, BGP4 can select more than one equal-cost path to a destination.

A BGP4 route consists of the following information:

- Network number (prefix) – A value comprised of the network mask bits and an IP address (<IP address>/<mask bits>); for example, 192.215.129.0/18 indicates a network mask of 18 bits applied to the IP address 192.215.129.0. When a BGP4 Layer 3 Switch advertises a route to one of its neighbors, the route is expressed in this format.
- AS-path – A list of the other ASs through which a route passes. BGP4 routers can use the AS-path to detect and eliminate routing loops. For example, if a route received by a BGP4 router contains the AS that the router is in, the router does not add the route to its own BGP4 table. (The BGP4 RFCs refer to the AS-path as "AS_PATH".)
- Additional path attributes – A list of additional parameters that describe the route. The route origin and next hop are examples of these additional path attributes.

NOTE: The Layer 3 Switch re-advertises a learned best BGP4 route to the Layer 3 Switch's neighbors even when the software does not select that route for installation in the IP route table. The best BGP4 route is the route that the software selects based on comparison of the BGP4 route path's attributes.

After a Foundry Layer 3 Switch successfully negotiates a BGP4 session with a neighbor (a BGP4 peer), the Foundry Layer 3 Switch exchanges complete BGP4 route tables with the neighbor. After this initial exchange, the Foundry Layer 3 Switch and all other RFC 1771-compliant BGP4 routers send UPDATE messages to inform neighbors of new, changed, or no longer feasible routes. BGP4 routers do not send regular updates. However, if configured to do so, a BGP4 router does regularly send KEEPALIVE messages to its peers to maintain BGP4

sessions with them if the router does not have any route information to send in an UPDATE message. See "BGP4 Message Types" on page 21-5 for information about BGP4 messages.

How BGP4 Selects a Path for a Route

When multiple paths for the same route are known to a BGP4 router, the router uses the following algorithm to weigh the paths and determine the optimal path for the route. The optimal path depends on various parameters, which can be modified. (See "Optional Configuration Tasks" on page 21-21.)

1. Is the next hop accessible though an Interior Gateway Protocol (IGP) route? If not, ignore the route.

   NOTE: The device does not use the default route to resolve BGP4 next hop. Also see "Enabling Next-Hop Recursion" on page 21-26.

2. Use the path with the largest weight.
3. If the weights are the same, prefer the route with the largest local preference.
4. If the routes have the same local preference, prefer the route that was originated locally (by this BGP4 Layer 3 Switch).
5. If the local preferences are the same, prefer the route with the shortest AS-path. An AS-SET counts as 1. A confederation path length, if present, is not counted as part of the path length.
6. If the AS-path lengths are the same, prefer the route with the lowest origin type. From low to high, route origin types are valued as follows:
   - IGP is lowest
   - EGP is higher than IGP but lower than INCOMPLETE
   - INCOMPLETE is highest
7. If the routes have the same origin type, prefer the route with the lowest MED. For a definition of MED, see "Configuring the Layer 3 Switch To Always Compare Multi-Exit Discriminators (MEDs)" on page 21-31.

   0BGP4 compares the MEDs of two otherwise equivalent paths ***if and only if*** the routes were learned from the same neighboring AS. This behavior is called ***deterministic MED***. 0Deterministic MED is always enabled and cannot be disabled.

   In addition, you can enable the Layer 3 Switch to always compare the MEDs, regardless of the AS information in the paths. To enable this comparison, enter the **always-compare-med** command at the BGP4 configuration level of the CLI. This option is disabled by default.

   NOTE: By default, value 0 (most favorable) is used in MED comparison when the MED attribute is not present. The default MED comparison results in the Layer 3 Switch favoring the route paths that are missing their MEDs. 0You can use the **med-missing-as-worst** command to make the Layer 3 Switch regard a BGP route with a missing MED attribute as the least favorable route, when comparing the MEDs of the routes.

   NOTE: MED comparison is not performed for internal routes originated within the local AS or confederation.

8. Prefer routes in the following order:
   - Routes received through EBGP from a BGP4 neighbor outside of the confederation
   - Routes received through EBGP from a BGP4 router within the confederation
   - Routes received through IBGP
9. If all the comparisons above are equal, prefer the route with the lowest IGP metric to the BGP4 next hop. This is the closest internal path inside the AS to reach the destination.

10. If the internal paths also are the same and BGP4 load sharing is enabled, load share among the paths. Otherwise, prefer the route that comes from the BGP4 router with the lowest router ID.

NOTE: Foundry Layer 3 Switches support BGP4 load sharing among multiple equal-cost paths. BGP4 load sharing enables the Layer 3 Switch to balance the traffic across the multiple paths instead of choosing just one path based on router ID. For EBGP routes, load sharing applies only when the paths are from neighbors within the same remote AS. EBGP paths from neighbors in different ASs are not compared.

BGP4 Message Types

BGP4 routers communicate with their neighbors (other BGP4 routers) using the following types of messages:

- OPEN
- UPDATE
- KEEPALIVE
- NOTIFICATION

OPEN Message

After a BGP4 router establishes a TCP connection with a neighboring BGP4 router, the routers exchange OPEN messages. An OPEN message indicates the following:

- BGP version – Indicates the version of the protocol that is in use on the router. BGP version 4 supports Classless Interdomain Routing (CIDR) and is the version most widely used in the Internet. Version 4 also is the only version supported on Foundry Layer 3 Switches.
- AS number – A two-byte number that identifies the AS to which the BGP4 router belongs.
- Hold Time – The number of seconds a BGP4 router will wait for an UPDATE or KEEPALIVE message (described below) from a BGP4 neighbor before assuming that the neighbor is dead. BGP4 routers exchange UPDATE and KEEPALIVE messages to update route information and maintain communication. If BGP4 neighbors are using different Hold Times, the lowest Hold Time is used by the neighbors. If the Hold Time expires, the BGP4 router closes its TCP connection to the neighbor and clears any information it has learned from the neighbor and cached.

  You can configure the Hold Time to be 0, in which case a BGP4 router will consider its neighbors to always be up. For directly-attached neighbors, you can configure the Foundry Layer 3 Switch to immediately close the TCP connection to the neighbor and clear entries learned from an EBGP neighbor if the interface to that neighbor goes down. This capability is provided by the fast external fallover feature, which is disabled by default.
- BGP Identifier – The router ID. The BGP Identifier (router ID) identifies the BGP4 router to other BGP4 routers. Foundry Layer 3 Switches use the same router ID for OSPF and BGP4. If you do not set a router ID, the software uses the IP address on the lowest numbered loopback interface configured on the router. If the Layer 3 Switch does not have a loopback interface, the default router ID is the lowest numbered IP address configured on the device. For more information or to change the router ID, see “Changing the Router ID” on page 16-23.
- Parameter list – An optional list of additional parameters used in peer negotiation with BGP4 neighbors.

UPDATE Message

After BGP4 neighbors establish a BGP4 connection over TCP and exchange their BGP4 routing tables, they do not send periodic routing updates. Instead, a BGP4 neighbor sends an update to its neighbor when it has a new route to advertise or routes have changed or become unfeasible. An UPDATE message can contain the following information:

- Network Layer Reachability Information (NLRI) – The mechanism by which BGP4 supports Classless Interdomain Routing (CIDR). An NLRI entry consists of an IP prefix that indicates a network being advertised by the UPDATE message. The prefix consists of an IP network number and the length of the network portion of the number. For example, an UPDATE message with the NLRI entry 192.215.129.0/18 indicates a route to IP network 192.215.129.0 with network mask 255.255.192.0. The binary equivalent of this mask is 18 consecutive one bits, thus “18” in the NLRI entry.
- Path attributes – Parameters that indicate route-specific information such as path information, route preference, next hop values, and aggregation information. BGP4 uses the path attributes to make filtering and routing decisions.
- Unreachable routes – A list of routes that have been in the sending router’s BGP4 table but are no longer feasible. The UPDATE message lists unreachable routes in the same format as new routes: <IP address>/<CIDR prefix>.

KEEPALIVE Message

BGP4 routers do not regularly exchange UPDATE messages to maintain the BGP4 sessions. For example, if a Layer 3 Switch configured to perform BGP4 routing has already sent the latest route information to its peers in UPDATE messages, the router does not send more UPDATE messages. Instead, BGP4 routers send KEEPALIVE messages to maintain the BGP4 sessions. KEEPALIVE messages are 19 bytes long and consist only of a message header; they contain no routing data.

BGP4 routers send KEEPALIVE messages at a regular interval, the Keep Alive Time. The default Keep Alive Time on Foundry Layer 3 Switches is 60 seconds.

A parameter related to the Keep Alive Time is the Hold Time. A BGP4 router’s Hold Time determines how many seconds the router will wait for a KEEPALIVE or UPDATE message from a BGP4 neighbor before deciding that the neighbor is dead. The Hold Time is negotiated when BGP4 routers exchange OPEN messages; the lower Hold Time is then used by both neighbors. For example, if BGP4 Router A sends a Hold Time of 5 seconds and BGP4 Router B sends a Hold Time of 4 seconds, both routers use 4 seconds as the Hold Time for their BGP4 session. The default Hold Time is 180 seconds. Generally, the Hold Time is configured to three times the value of the Keep Alive Time.

If the Hold Time is 0, a BGP4 router assumes that its neighbor is alive regardless of how many seconds pass between receipt of UPDATE or KEEPALIVE messages.

NOTIFICATION Message

When you close the router’s BGP4 session with a neighbor, or the router detects an error in a message received from the neighbor, or an error occurs on the router, the router sends a NOTIFICATION message to the neighbor. No further communication takes place between the BGP4 router that sent the NOTIFICATION and the neighbor(s) that received the NOTIFICATION.

Basic Configuration and Activation for BGP4

BGP4 is disabled by default. To enable BGP4 and place your Foundry Layer 3 Switch into service as a BGP4 router, you must perform at least the following steps:

1. Enable the BGP4 protocol.
2. Set the local AS number.

   NOTE: You must specify the local AS number. BGP4 is not functional until you specify the local AS number.

3. Add each BGP4 neighbor (peer BGP4 router) and identify the AS the neighbor is in.
4. Save the BGP4 configuration information to the system configuration file.

NOTE: By default, the Foundry router ID is the IP address configured on the lowest numbered loopback interface. If the Layer 3 Switch does not have a loopback interface, the default router ID is the lowest numbered IP interface address configured on the device. For more information or to change the router ID, see “Changing the Router ID” on page 16-23. If you change the router ID, all current BGP4 sessions are cleared.

```
FESX424 Router> enable
FESX424 Router# configure terminal
FESX424 Router(config)# router bgp
BGP4: Please configure 'local-as' parameter in order to enable BGP4.
FESX424 Router(config-bgp-router)# local-as 10
FESX424 Router(config-bgp-router)# neighbor 209.157.23.99 remote-as 100
FESX424 Router(config-bgp-router)# write memory
```

NOTE: When BGP4 is enabled on a Foundry Layer 3 Switch, you do not need to reset the system. The protocol is activated as soon as you enable it. Moreover, the router begins a BGP4 session with a BGP4 neighbor as soon as you add the neighbor.

Note Regarding Disabling BGP4

If you disable BGP4, the Layer 3 Switch removes all the running configuration information for the disabled protocol from the running-config. To restore the BGP4 configuration, you must reload the software to load the configuration from the startup-config. Moreover, when you save the configuration to the startup-config file after disabling the protocol, all the configuration information for the disabled protocol is removed from the startup-config file.

The CLI displays a warning message such as the following:

```
FESX424 Router(config-bgp-router)# no router bgp
router bgp mode now disabled. All bgp config data will be lost when writing to
flash!
```

If you are testing a BGP4 configuration and are likely to disable and re-enable the protocol, you might want to make a backup copy of the startup-config file containing the protocol’s configuration information. This way, if you remove the configuration information by saving the configuration after disabling the protocol, you can restore the configuration by copying the backup copy of the startup-config file onto the flash memory.

NOTE: To disable BGP4 without losing the BGP4 configuration information, remove the local AS (for example, by entering the **no local-as** <num> command). In this case, BGP4 retains the other configuration information but is not operational until you set the local AS again.

BGP4 Parameters

You can modify or set the following BGP4 parameters.

- Optional – Define the router ID. (The same router ID also is used by OSPF.)
- Required – Specify the local AS number.
- Optional – Add a loopback interface for use with neighbors.
- Required – Identify BGP4 neighbors.
- Optional – Change the Keep Alive Time and Hold Time.
- Optional – Change the update timer for route changes.
- Optional – Enable fast external fallover.
- Optional – Specify a list of individual networks in the local AS to be advertised to remote ASs using BGP4.
- Optional – Change the default local preference for routes.

- Optional – Enable the default route (default-information-originate).
- Optional – Enable use of a default route to resolve a BGP4 next-hop route.
- Optional – Change the default MED (metric).
- Optional – Enable next-hop recursion.
- Optional – Change the default administrative distances for EBGP, IBGP, and locally originated routes.
- Optional – Require the first AS in an Update from an EBGP neighbor to be the neighbor's AS.
- Optional – Change MED comparison parameters.
- Optional – Disable comparison of the AS-Path length.
- Optional – Enable comparison of the router ID.
- Optional – Enable auto summary to summarize routes at an IP class boundary (A, B, or C).
- Optional – Aggregate routes in the BGP4 route table into CIDR blocks.
- Optional – Configure the router as a BGP4 router reflector.
- Optional – Configure the Layer 3 Switch as a member of a BGP4 confederation.
- Optional – Change the default metric for routes that BGP4 redistributes into RIP or OSPF.
- Optional – Change the parameters for RIP, OSPF, or static routes redistributed into BGP4.
- Optional – Change the number of paths for BGP4 load sharing.
- Optional – Change other load-sharing parameters
- Optional – Define BGP4 address filters.
- Optional – Define BGP4 AS-path filters.
- Optional – Define BGP4 community filters.
- Optional – Define IP prefix lists.
- Optional – Define neighbor distribute lists.
- Optional – Define BGP4 route maps for filtering routes redistributed into RIP and OSPF.
- Optional – Define route flap dampening parameters.

NOTE: When using CLI, you set global level parameters at the BGP CONFIG Level of the CLI. You can reach the BGP CONFIG level by entering **router bgp…** at the global CONFIG level.

When Parameter Changes Take Effect

Some parameter changes take effect immediately while others do not take full effect until the router's sessions with its neighbors are reset. Some parameters do not take effect until the router is rebooted.

Immediately

The following parameter changes take effect immediately:

- Enable or disable BGP.
- Set or change the local AS.
- Add neighbors.
- Change the update timer for route changes.
- Disable or enable fast external fallover.
- Specify individual networks that can be advertised.

- Change the default local preference, default information originate setting, or administrative distance.
- Enable or disable use of a default route to resolve a BGP4 next-hop route.
- Enable or disable MED (metric) comparison.
- Require the first AS in an Update from an EBGP neighbor to be the neighbor's AS.
- Change MED comparison parameters.
- Disable comparison of the AS-Path length.
- Enable comparison of the router ID.
- Enable next-hop recursion.
- Enable or disable auto summary.
- Change the default metric.
- Disable or re-enable route reflection.
- Configure confederation parameters.
- Disable or re-enable load sharing.
- Change the maximum number of load-sharing paths.
- Change other load-sharing parameters.
- Define route flap dampening parameters.
- Add, change, or negate redistribution parameters (except changing the default MED; see below).
- Add, change, or negate route maps (when used by the **network** command or a redistribution command).

After Resetting Neighbor Sessions

The following parameter changes take effect only after the router's BGP4 sessions are cleared, or reset using the "soft" clear option. (See "Closing or Resetting a Neighbor Session" on page 21-105.)

- Change the Hold Time or Keep Alive Time.
- Aggregate routes.
- Add, change, or negate filter tables.

After Disabling and Re-Enabling Redistribution

The following parameter change takes effect only after you disable and then re-enable redistribution:

- Change the default MED (metric).

Memory Considerations

BGP4 handles a very large number of routes and therefore requires a lot of memory. For example, in a typical configuration with just a single BGP4 neighbor, a BGP4 router may need to be able to hold up to 80,000 routes. Many configurations, especially those involving more than one neighbor, can require the router to hold even more routes. Foundry Layer 3 Switches and NetIron Internet Backbone routers provide dynamic memory allocation for BGP4 data. These devices automatically allocate memory when needed to support BGP4 neighbors, routes, and route attribute entries. Dynamic memory allocation is performed automatically by the software and does not require a reload.

Table 21.2 lists the maximum total amount of system memory (DRAM) BGP4 can use. The maximum depends on the total amount of system memory on the device.

Table 21.2: Maximum Memory Usage

Platform	Maximum Memory BGP4 Can Use
FESX with 128 MB	30 MB
FSX with Management 1 module with 256 MB	130 MB
FSX with Management 2 module with 512 MB	400 MB

The memory amounts listed in the table are for all BGP4 data, including routes received from neighbors, BGP route advertisements (routes sent to neighbors), and BGP route attribute entries. The routes sent to and received from neighbors use the most BGP4 memory. Generally, the actual limit to the number of neighbors, routes, or route attribute entries the device can accommodate depends on how many routes the Layer 3 Switch sends to and receives from the neighbors.

In some cases, where most of the neighbors do not send or receive a full BGP route table (about 80,000 routes), the memory can support a larger number of BGP4 neighbors. However, if most of the BGP4 neighbors send or receive full BGP route tables, the number of BGP neighbors the memory can support is less than in configurations where the neighbors send smaller route tables.

As a guideline, Layer 3 Switches with a 512 MB Management 4 module can accommodate 150 – 200 neighbors, with the assumption that the Layer 3 Switch receives about one million routes total from all neighbors and sends about eight million routes total to neighbors. For each additional one million incoming routes, the capacity for outgoing routes decreases by around two million.

Memory Configuration Options Obsoleted by Dynamic Memory

Devices that support dynamic BGP4 memory allocation do not require or even support static configuration of memory for BGP4 neighbors, routes, or route attributes. Consequently, the following CLI commands and equivalent Web management options are not supported on these devices:

- **max-neighbors** <num>
- **max-routes** <num>
- **max-attribute-entries** <num>

If you boot a device that has a startup-config file that contains these commands, the software ignores the commands and uses dynamic memory allocation for BGP4. The first time you save the device's running configuration (running-config) to the startup-config file, the commands are removed from the file.

Basic Configuration Tasks

The following sections describe how to perform the configuration tasks that are required to use BGP4 on the Foundry Layer 3 Switch. You can modify many parameters in addition to the ones described in this section. See "Optional Configuration Tasks" on page 21-21.

Enabling BGP4 on the Router

When you enable BGP4 on the router, BGP4 is automatically activated. To enable BGP4 on the router, enter the following commands:

```
FESX424 Router> enable
FESX424 Router# configure terminal
FESX424 Router(config)# router bgp
BGP4: Please configure 'local-as' parameter in order to enable BGP4.
```



```
FESX424 Router(config-bgp-router)# local-as 10
FESX424 Router(config-bgp-router)# neighbor 209.157.23.99 remote-as 100
FESX424 Router(config-bgp-router)# write memory
```

Changing the Router ID

The OSPF and BGP4 protocols use router IDs to identify the routers that are running the protocols. A router ID is a valid, unique IP address and sometimes is an IP address configured on the router. The router ID cannot be an IP address in use by another device.

By default, the router ID on a Foundry Layer 3 Switch is one of the following:

- If the router has loopback interfaces, the default router ID is the IP address configured on the lowest numbered loopback interface configured on the Layer 3 Switch. For example, if you configure loopback interfaces 1, 2, and 3 as follows, the default router ID is 9.9.9.9/24:
  - Loopback interface 1, 9.9.9.9/24
  - Loopback interface 2, 4.4.4.4/24
  - Loopback interface 3, 1.1.1.1/24
- If the device does not have any loopback interfaces, the default router ID is the lowest numbered IP interface address configured on the device.

NOTE: Foundry Layer 3 Switches use the same router ID for both OSPF and BGP4. If the router is already configured for OSPF, you may want to use the router ID that is already in use on the router rather than set a new one. To display the router ID, enter the **show ip** CLI command at any CLI level.

To change the router ID, enter a command such as the following:

```
FESX424 Router(config)# ip router-id 209.157.22.26
```

Syntax: ip router-id <ip-addr>

The <ip-addr> can be any valid, unique IP address.

NOTE: You can specify an IP address used for an interface on the Foundry Layer 3 Switch, but do not specify an IP address in use by another device.

Setting the Local AS Number

The local AS number identifies the AS the Foundry BGP4 router is in. The AS number can be from 1 – 65535. There is no default. AS numbers 64512 – 65535 are the well-known private BGP4 AS numbers and are not advertised to the Internet community.

To set the local AS number, enter commands such as the following:

```
FESX424 Router(config)# router bgp
BGP4: Please configure 'local-as' parameter in order to enable BGP4.
FESX424 Router(config-bgp-router)# local-as 10
FESX424 Router(config-bgp-router)# write memory
```

Syntax: [no] local-as <num>

The <num> parameter specifies the local AS number.

Adding a Loopback Interface

You can configure the router to use a loopback interface instead of a specific port or virtual routing interface to communicate with a BGP4 neighbor. A loopback interface adds stability to the network by working around route flap problems that can occur due to unstable links between the router and its neighbors.

Loopback interfaces are always up, regardless of the states of physical interfaces. Loopback interfaces are especially useful for IBGP neighbors (neighbors in the same AS) that are multiple hops away from the router. When you configure a BGP4 neighbor on the router, you can specify whether the router uses the loopback interface to communicate with the neighbor. As long as a path exists between the router and its neighbor, BGP4 information can be exchanged. The BGP4 session is not associated with a specific link but instead is associated with the virtual interfaces.

You can add up to 24 IP addresses to each loopback interface.

NOTE: If you configure the Foundry Layer 3 Switch to use a loopback interface to communicate with a BGP4 neighbor, the peer IP address on the remote router pointing to your loopback address must be configured.

To add a loopback interface, enter commands such as those shown in the following example:

```
FESX424 Router(config-bgp-router)# exit
FESX424 Router(config)# int loopback 1
FESX424 Router(config-lbif-1)# ip address 10.0.0.1/24
```

Syntax: interface loopback <num>

The <num> value can be from 1 – 8 on Chassis Layer 3 Switches and the TurboIron/8 Layer 3 Switch. The value can be from 1 – 4 on the NetIron Stackable Layer 3 Switch.

Adding BGP4 Neighbors

The BGP4 protocol does not contain a peer discovery process. Therefore, for each of the router's BGP4 neighbors (peers), you must indicate the neighbor's IP address and the AS each neighbor is in. Neighbors that are in different ASs communicate using EBGP. Neighbors within the same AS communicate using IBGP.

NOTE: If the Layer 3 Switch has multiple neighbors with similar attributes, you can simplify configuration by configuring a peer group, then adding individual neighbors to it. The configuration steps are similar, except you specify a peer group name instead of a neighbor IP address when configuring the neighbor parameters, then add individual neighbors to the peer group. See "Adding a BGP4 Peer Group" on page 21-17.

NOTE: The Layer 3 Switch attempts to establish a BGP4 session with a neighbor as soon as you enter a command specifying the neighbor's IP address. If you want to completely configure the neighbor parameters before the Layer 3 Switch establishes a session with the neighbor, you can administratively shut down the neighbor. See "Administratively Shutting Down a Session with a BGP4 Neighbor" on page 21-20.

To add a BGP4 neighbor with IP address 209.157.22.26, enter the following command:

```
FESX424 Router(config-bgp-router)# neighbor 209.157.22.26
```

The neighbor's <ip-addr> must be a valid IP address.

The **neighbor** command has some additional parameters, as shown in the following syntax:

Syntax: [no] neighbor <ip-addr> | <peer-group-name>
[advertisement-interval <num>]
[capability orf prefixlist [send | receive]]
[default-originate [route-map <map-name>]]
[description <string>]
[distribute-list in | out <num,num,...> | <acl-num> in | out]
[ebgp-multihop [<num>]]
[filter-list in | out <num,num,...> | <acl-num> in | out | weight]
[maximum-prefix <num> [<threshold>] [teardown]]
[next-hop-self]
[nlri multicast | unicast | multicast unicast]
[password [0 | 1] <string>]
[prefix-list <string> in | out]
[remote-as <as-number>]
[remove-private-as]
[route-map in | out <map-name>]
[route-reflector-client]
[send-community]
[soft-reconfiguration inbound]
[shutdown]
[timers keep-alive <num> hold-time <num>]
[unsuppress-map <map-name>]
[update-source <ip-addr> | ethernet [<slotnum>/]<portnum> | loopback <num> | ve <num>]
[weight <num>]

The <ip-addr> | <peer-group-name> parameter indicates whether you are configuring an individual neighbor or a peer group. If you specify a neighbor's IP address, you are configuring that individual neighbor. If you specify a peer group name, you are configuring a peer group. See "Adding a BGP4 Peer Group" on page 21-17.

advertisement-interval <num> specifies the minimum delay (in seconds) between messages to the specified neighbor. The default is 30 for EBGP neighbors (neighbors in other ASs). The default is 5 for IBGP neighbors (neighbors in the same AS). The range is 0 – 600.

NOTE: The Layer 3 Switch applies the advertisement interval only under certain conditions. The Layer 3 Switch does not apply the advertisement interval when sending initial updates to a BGP4 neighbor. As a result, the Layer 3 Switch sends the updates one immediately after another, without waiting for the advertisement interval.

capability orf prefixlist [**send** | **receive**] configures cooperative router filtering. The **send** | **receive** parameter specifies the support you are enabling:

- **send** – The Layer 3 Switch sends the IP prefix lists as Outbound Route Filters (ORFs) to the neighbor.
- **receive** – The Layer 3 Switch accepts filters as Outbound Route Filters (ORFs) from the neighbor.

If you do not specify the capability, both capabilities are enabled. The **prefixlist** parameter specifies the type of filter you want to send to the neighbor.

For more information, see "Configuring Cooperative BGP4 Route Filtering" on page 21-55.

NOTE: The current release supports cooperative filtering only for filters configured using IP prefix lists.

default-originate [**route-map** <map-name>] configures the Layer 3 Switch to send the default route 0.0.0.0 to the neighbor. If you use the route-map <map-name> parameter, the route map injects the default route conditionally, based on the match conditions in the route map.

description <string> specifies a name for the neighbor. You can enter an alphanumeric text string up to 80 characters long.

distribute-list in | out <num,num,...> specifies a distribute list to be applied to updates to or from the specified neighbor. The **in | out** keyword specifies whether the list is applied on updates received from the neighbor or sent to the neighbor. The <num,num,...> parameter specifies the list of address-list filters. The router applies the filters in the order in which you list them and stops applying the filters in the distribute list when a match is found.

Alternatively, you can specify **distribute-list** <acl-num> **in | out** to use an IP ACL instead of a distribute list. In this case, <acl-num> is an IP ACL.

NOTE: By default, if a route does not match any of the filters, the Layer 3 Switch denies the route. To change the default behavior, configure the last filter as "permit any any".

NOTE: The address filter must already be configured. See "Filtering Specific IP Addresses" on page 21-40.

ebgp-multihop [<num>] specifies that the neighbor is more than one hop away and that the session type with the neighbor is thus EBGP-multihop. This option is disabled by default. The <num> parameter specifies the TTL you are adding for the neighbor. You can specify a number from 0 – 255. The default is 0. If you leave the EBGP TTL value set to 0, the software uses the IP TTL value.

filter-list in | out <num,num,...> specifies an AS-path filter list or a list of AS-path ACLs. The **in | out** keyword specifies whether the list is applied on updates received from the neighbor or sent to the neighbor. If you specify **in** or **out**, The <num,num,...> parameter specifies the list of AS-path filters. The router applies the filters in the order in which you list them and stops applying the filters in the AS-path filter list when a match is found. The **weight** <num> parameter specifies a weight that the Layer 3 Switch applies to routes received from the neighbor that match the AS-path filter or ACL. You can specify a number from 0 – 65535.

Alternatively, you can specify **filter-list** <acl-num> **in | out | weight** to use an AS-path ACL instead of an AS-path filter list. In this case, <acl-num> is an AS-path ACL.

NOTE: By default, if an AS-path does not match any of the filters or ACLs, the Layer 3 Switch denies the route. To change the default behavior, configure the last filter or ACL as "permit any any".

NOTE: The AS-path filter or ACL must already be configured. See "Filtering AS-Paths" on page 21-41.

maximum-prefix <num> specifies the maximum number of IP network prefixes (routes) that can be learned from the specified neighbor or peer group. You can specify a value from 0 – 4294967295. The default is 0 (unlimited).

- The <num> parameter specifies the maximum number. You can specify a value from 0 – 4294967295. The default is 0 (unlimited).
- The <threshold> parameter specifies the percentage of the value you specified for the **maximum-prefix** <num>, at which you want the software to generate a Syslog message. You can specify a value from 1 (one percent) to 100 (100 percent). The default is 100.
- The **teardown** parameter tears down the neighbor session if the maximum-prefix limit is exceeded. The session remains shutdown until you clear the prefixes using the **clear ip bgp neighbor all** or **clear ip bgp neighbor** <ip-addr> command, or change the neighbor's maximum-prefix configuration. The software also generates a Syslog message.

next-hop-self specifies that the router should list itself as the next hop in updates sent to the specified neighbor. This option is disabled by default.

The **nlri multicast | unicast | multicast unicast** parameter specifies whether the neighbor is a multicast neighbor or a unicast neighbor. For MBGP, you must specify **multicast**. Optionally, you also can specify **unicast** if you want the Layer 3 Switch to exchange unicast (BGP4) routes as well as multicast routes with the neighbor. The default is **unicast** only.

password [**0 | 1**] <string> specifies an MD5 password for securing sessions between the Layer 3 Switch and the neighbor. You can enter a string up to 80 characters long. The string can contain any alphanumeric characters, but the first character cannot be a number. If the password contains a number, do not enter a space following the number.

The **0 | 1** parameter is the encryption option, which you can omit (the default) or which can be one of the following.

- **0** – Disables encryption for the authentication string you specify with the command. The password or string is shown as clear text in the output of commands that display neighbor or peer group configuration information.

- **1** – Assumes that the authentication string you enter is the encrypted form, and decrypts the value before using it.

For more information, see "Encryption of BGP4 MD5 Authentication Keys" on page 21-16.

NOTE: If you want the software to assume that the value you enter is the clear-text form, and to encrypt display of that form, do not enter 0 or 1. Instead, omit the encryption option and allow the software to use the default behavior.

If you specify encryption option 1, the software assumes that you are entering the encrypted form of the password or authentication string. In this case, the software decrypts the password or string you enter before using the value for authentication. If you accidentally enter option 1 followed by the clear-text version of the password or string, authentication will fail because the value used by the software will not match the value you intended to use.

prefix-list <string> **in | out** specifies an IP prefix list. You can use IP prefix lists to control routes to and from the neighbor. IP prefix lists are an alternative method to AS-path filters. The **in | out** keyword specifies whether the list is applied on updates received from the neighbor or sent to the neighbor. You can configure up to 1000 prefix list filters. The filters can use the same prefix list or different prefix lists. To configure an IP prefix list, see "Defining IP Prefix Lists" on page 21-47.

remote-as <as-number> specifies the AS the remote neighbor is in. The <as-number> can be a number from 1 – 65535. There is no default.

remove-private-as configures the router to remove private AS numbers from UPDATE messages the router sends to this neighbor. The router will remove AS numbers 64512 – 65535 (the well-known BGP4 private AS numbers) from the AS-path attribute in UPDATE messages the Layer 3 Switch sends to the neighbor. This option is disabled by default.

route-map in | out <map-name> specifies a route map the Layer 3 Switch will apply to updates sent to or received from the specified neighbor. The **in | out** keyword specifies whether the list is applied on updates received from the neighbor or sent to the neighbor.

NOTE: The route map must already be configured. See "Defining Route Maps" on page 21-48.

route-reflector-client specifies that this neighbor is a route-reflector client of the router. Use the parameter only if this router is going to be a route reflector. For information, see "Configuring Route Reflection Parameters" on page 21-32. This option is disabled by default.

send-community enables sending the community attribute in updates to the specified neighbor. By default, the router does not send the community attribute.

shutdown administratively shuts down the session with this neighbor. Shutting down the session allows you to completely configure the neighbor and save the configuration without actually establishing a session with the neighbor. This option is disabled by default.

soft-reconfiguration inbound enables the soft reconfiguration feature, which stores all the route updates received from the neighbor. If you request a soft reset of inbound routes, the software performs the reset by comparing the policies against the stored route updates, instead of requesting the neighbor's BGP4 route table or resetting the session with the neighbor. See "Using Soft Reconfiguration" on page 21-100.

timers keep-alive <num> **hold-time** <num> overrides the global settings for the Keep Alive Time and Hold Time. For the Keep Alive Time, you can specify from 0 – 65535 seconds. For the Hold Time, you can specify 0 or 3 – 65535 (1 and 2 are not allowed). If you set the Hold Time to 0, the router waits indefinitely for messages from a neighbor without concluding that the neighbor is dead. The defaults for these parameters are the currently configured global Keep Alive Time and Hold Time. For more information about these parameters, see "Changing the Keep Alive Time and Hold Time" on page 21-21.

unsuppress-map <map-name> removes route dampening from a neighbor's routes when those routes have been dampened due to aggregation. See "Removing Route Dampening from a Neighbor's Routes Suppressed Due to Aggregation" on page 21-61.

update-source <ip-addr> | **ethernet** [<slotnum>/]<portnum> | **loopback** <num> | **ve** <num> configures the router to communicate with the neighbor through the specified interface. There is no default.

weight <num> specifies a weight the Layer 3 Switch will add to routes received from the specified neighbor. BGP4 prefers larger weights over smaller weights. The default weight is 0.

Encryption of BGP4 MD5 Authentication Keys

When you configure a BGP4 neighbor or neighbor peer group, you can specify an MD5 authentication string for authenticating packets exchanged with the neighbor or peer group of neighbors.

For added security, the software encrypts display of the authentication string by default. The software also provides an optional parameter to disable encryption of the authentication string, on an individual neighbor or peer group basis. By default, the MD5 authentication strings are displayed in encrypted format in the output of the following commands:

- **show running-config** (or **write terminal**)
- **show configuration**
- **show ip bgp config**

When encryption of the authentication string is enabled, the string is encrypted in the CLI regardless of the access level you are using.

If you display the running-config after reloading, the BGP4 commands that specify an authentication string show the string in encrypted form.

In addition, when you save the configuration to the startup-config file, the file contains the new BGP4 command syntax and encrypted passwords or strings.

NOTE: Foundry recommends that you save a copy of the startup-config file for each Layer 3 Switch you plan to upgrade.

Encryption Example

The following commands configure a BGP4 neighbor and a peer group, and specify MD5 authentication strings (passwords) for authenticating packets exchanged with the neighbor or peer group.

```
FESX424 Router(config-bgp-router)# local-as 2
FESX424 Router(config-bgp-router)# neighbor xyz peer-group
FESX424 Router(config-bgp-router)# neighbor xyz password abc
FESX424 Router(config-bgp-router)# neighbor 10.10.200.102 peer-group xyz
FESX424 Router(config-bgp-router)# neighbor 10.10.200.102 password test
```

Here is how the commands appear when you display the BGP4 configuration commands:

```
FESX424 Router(config-bgp-router)# show ip bgp config
Current BGP configuration:
router bgp
 local-as 2
 neighbor xyz peer-group
 neighbor xyz password 1 $!2d
 neighbor 10.10.200.102 peer-group xyz
 neighbor 10.10.200.102 remote-as 1
 neighbor 10.10.200.102 password 1 $on-o
```

Notice that the software has converted the commands that specify an authentication string into the new syntax (described below), and has encrypted display of the authentication strings.

Command Syntax

Since the default behavior does not affect the BGP4 configuration itself but does encrypt display of the authentication string, the CLI does not list the encryption options.

Syntax: [no] neighbor <ip-addr> | <peer-group-name> password [0 | 1] <string>

The <ip-addr> | <peer-group-name> parameter indicates whether you are configuring an individual neighbor or a peer group. If you specify a neighbor's IP address, you are configuring that individual neighbor. If you specify a peer group name, you are configuring a peer group.

The **password** <string> parameter specifies an MD5 authentication string for securing sessions between the Layer 3 Switch and the neighbor. You can enter a string up to 80 characters long. The string can contain any alphanumeric characters, but the first character cannot be a number. If the password contains a number, do not enter a space following the number.

The **0 | 1** parameter is the encryption option, which you can omit (the default) or which can be one of the following.

- **0** – Disables encryption for the authentication string you specify with the command. The password or string is shown as clear text in the output of commands that display neighbor or peer group configuration information.
- **1** – Assumes that the authentication string you enter is the encrypted form, and decrypts the value before using it.

NOTE: If you want the software to assume that the value you enter is the clear-text form, and to encrypt display of that form, do not enter 0 or 1. Instead, omit the encryption option and allow the software to use the default behavior.

If you specify encryption option 1, the software assumes that you are entering the encrypted form of the password or authentication string. In this case, the software decrypts the password or string you enter before using the value for authentication. If you accidentally enter option 1 followed by the clear-text version of the password or string, authentication will fail because the value used by the software will not match the value you intended to use.

Displaying the Authentication String

If you want to display the authentication string, enter the following commands:

```
FESX424 Router(config)# enable password-display
FESX424 Router(config)# show ip bgp neighbors
```

The **enable password-display** command enables display of the authentication string, but only in the output of the **show ip bgp neighbors** command. Display of the string is still encrypted in the startup-config file and running-config. Enter the command at the global CONFIG level of the CLI.

NOTE: The command also displays SNMP community strings in clear text, in the output of the **show snmp server** command.

Adding a BGP4 Peer Group

A ***peer group*** is a set of BGP4 neighbors that share common parameters. Peer groups provide the following benefits:

- Simplified neighbor configuration – You can configure a set of neighbor parameters and then apply them to multiple neighbors. You do not need to individually configure the common parameters individually on each neighbor.
- Flash memory conservation – Using peer groups instead of individually configuring all the parameters for each neighbor requires fewer configuration commands in the startup-config file.

You can perform the following tasks on a peer-group basis.

- Reset neighbor sessions
- Perform soft-outbound resets (the Layer 3 Switch updates outgoing route information to neighbors but does not entirely reset the sessions with those neighbors)
- Clear BGP message statistics

- Clear error buffers

Peer Group Parameters

You can set all neighbor parameters in a peer group. When you add a neighbor to the peer group, the neighbor receives all the parameter settings you set in the group, except parameter values you have explicitly configured for the neighbor. If you do not set a neighbor parameter in the peer group and the parameter also is not set for the individual neighbor, the neighbor uses the default value.

Configuration Rules

The following rules apply to peer group configuration:

- You must configure a peer group before you can add neighbors to the peer group.
- If you remove a parameter from a peer group, the value for that parameter is reset to the default for all the neighbors within the peer group, unless you have explicitly set that parameter on individual neighbors. In this case, the value you set on the individual neighbors applies to those neighbors, while the default value applies to neighbors for which you have not explicitly set the value.

  NOTE: If you enter a command to remove the remote AS parameter from a peer group, the software checks to ensure that the peer group does not contain any neighbors. If the peer group does contain neighbors, the software does not allow you to remove the remote AS. The software prevents removing the remote AS in this case so that the neighbors in the peer group that are using the remote AS do not lose connectivity to the Layer 3 Switch.

- Once you add a neighbor to a peer group, you cannot configure the following outbound parameters (the parameters governing outbound traffic) for the neighbor.
  - Default-information-originate
  - Next-hop-self
  - Outbound route map
  - Outbound filter list
  - Outbound distribute list
  - Outbound prefix list
  - Remote AS, if configured for the peer group
  - Remove private AS
  - Route reflector client
  - Send community
  - Timers
  - Update source

  If you want to change an outbound parameter for an individual neighbor, you must first remove the neighbor from the peer group. In this case, you cannot re-add the neighbor to the same peer group, but you can add the neighbor to a different peer group. All the neighbors within a peer group must have the same values for the outbound parameters. To change an outbound parameter to the same value for all neighbors within a peer group, you can change the parameter on a peer-group basis. In this case, you do not need to remove the neighbors and change the parameter individually for each neighbor.
- If you add an outbound parameter to a peer group, that parameter is automatically applied to all neighbors within the peer group.
- When you add a neighbor to a peer group, the software removes any outbound parameters for that neighbor from the running configuration (running-config). As a result, when you save the configuration to the startup-config file, the file does not contain any outbound parameters for the individual neighbors you have placed in a peer group. The only outbound parameters the startup-config file contains for neighbors within a peer group are the parameters associated with the peer group itself. However, the running-config and the startup-config

file can contain individual parameters listed in the previous section as well as the settings for those parameters within a peer group.

You can override neighbor parameters that do not affect outbound policy on an individual neighbor basis.

- If you do not specify a parameter for an individual neighbor, the neighbor uses the value in the peer group.
- If you set the parameter for the individual neighbor, that value overrides the value you set in the peer group.
- If you add a parameter to a peer group that already contains neighbors, the parameter value is applied to neighbors that do not already have the parameter explicitly set. If a neighbor has the parameter explicitly set, the explicitly set value overrides the value you set for the peer group.
- If you remove the setting for a parameter from a peer group, the value for that parameter changes to the default value for all the neighbors in the peer group that do not have that parameter individually set.

Configuring a Peer Group

To configure a BGP4 peer group, enter commands such as the following at the BGP configuration level:

```
FESX424 Router(config-bgp-router)# neighbor PeerGroup1 peer-group
FESX424 Router(config-bgp-router)# neighbor PeerGroup1 description "EastCoast
Neighbors"
FESX424 Router(config-bgp-router)# neighbor PeerGroup1 remote-as 100
FESX424 Router(config-bgp-router)# neighbor PeerGroup1 distribute-list out 1
```

The commands in this example configure a peer group called “PeerGroup1” and set the following parameters for the peer group:

- A description, “EastCoast Neighbors”
- A remote AS number, 100
- A distribute list for outbound traffic

The software applies these parameters to each neighbor you add to the peer group. You can override the description parameter for individual neighbors. If you set the description parameter for an individual neighbor, the description overrides the description configured for the peer group. However, you cannot override the remote AS and distribute list parameters for individual neighbors. Since these parameters control outbound traffic, the parameters must have the same values for all neighbors within the peer group.

Syntax: neighbor <peer-group-name> peer-group

The <peer-group-name> parameter specifies the name of the group and can be up to 80 characters long. The name can contain special characters and internal blanks. If you use internal blanks, you must use quotation marks around the name. For example, the command **neighbor “My Three Peers” peer-group** is valid, but the command **neighbor My Three Peers peer-group** is not valid.

Syntax: [no] neighbor <ip-addr> | <peer-group-name>
[advertisement-interval <num>]
[default-originate [route-map <map-name>]]
[description <string>]
[distribute-list in | out <num,num,...> | <acl-num> in | out]
[ebgp-multihop [<num>]]
[filter-list in | out <num,num,...> | <acl-num> in | out | weight]
[maximum-prefix <num> [<threshold>] [teardown]]
[next-hop-self]
[password [0 | 1] <string>]
[prefix-list <string> in | out]
[remote-as <as-number>]
[remove-private-as]
[route-map in | out <map-name>]
[route-reflector-client]
[send-community]
[soft-reconfiguration inbound]
[shutdown]
[timers keep-alive <num> hold-time <num>]
[update-source loopback <num>]
[weight <num>]

Syntax: The <ip-addr> | <peer-group-name> parameter indicates whether you are configuring a peer group or an individual neighbor. You can specify a peer group name or IP address with the **neighbor** command. If you specify a peer group name, you are configuring a peer group. If you specify a neighbor's IP address, you are configuring that individual neighbor. Use the <ip-addr> parameter if you are configuring an individual neighbor instead of a peer group. See "Adding BGP4 Neighbors" on page 21-12.

The remaining parameters are the same ones supported for individual neighbors. See "Adding BGP4 Neighbors" on page 21-12.

Applying a Peer Group to a Neighbor

After you configure a peer group, you can add neighbors to the group. When you add a neighbor to a peer group, you are applying all the neighbor attributes specified in the peer group to the neighbor.

To add neighbors to a peer group, enter commands such as the following:

```
FESX424 Router(config-bgp-router)# neighbor 192.168.1.12 peer-group PeerGroup1
FESX424 Router(config-bgp-router)# neighbor 192.168.2.45 peer-group PeerGroup1
FESX424 Router(config-bgp-router)# neighbor 192.168.3.69 peer-group PeerGroup1
```

The commands in this example add three neighbors to the peer group "PeerGroup1". As members of the peer group, the neighbors automatically receive the neighbor parameter values configured for the peer group. You also can override the parameters (except parameters that govern outbound traffic) on an individual neighbor basis. For neighbor parameters not specified for the peer group, the neighbors use the default values.

Syntax: neighbor <ip-addr> peer-group <peer-group-name>

The <ip-addr> parameter specifies the IP address of the neighbor.

The <peer-group-name> parameter specifies the peer group name.

NOTE: You must add the peer group before you can add neighbors to it.

Administratively Shutting Down a Session with a BGP4 Neighbor

You can prevent the Layer 3 Switch from starting a BGP4 session with a neighbor by administratively shutting down the neighbor. This option is very useful for situations in which you want to configure parameters for a neighbor but are not ready to use the neighbor. You can shut the neighbor down as soon as you have added it the Layer 3 Switch, configure the neighbor parameters, then allow the Layer 3 Switch to re-establish a session with the neighbor by removing the shutdown option from the neighbor.

When you apply the new option to shut down a neighbor, the option takes place immediately and remains in effect until you remove the option. If you save the configuration to the startup-config file, the shutdown option remains in effect even after a software reload.

NOTE: The software also contains an option to end the session with a BGP4 neighbor and thus clear the routes learned from the neighbor. Unlike this clear option, the option for shutting down the neighbor can be saved in the startup-config file and thus can prevent the Layer 3 Switch from establishing a BGP4 session with the neighbor even after reloading the software.

NOTE: If you notice that a particular BGP4 neighbor never establishes a session with the Foundry Layer 3 Switch, check the Layer 3 Switch's running-config and startup-config files to see whether the configuration contains a command that is shutting down the neighbor. The neighbor may have been shut down previously by an administrator.

To shut down a BGP4 neighbor, enter commands such as the following:

```
FESX424 Router(config)# router bgp
FESX424 Router(config-bgp-router)# neighbor 209.157.22.26 shutdown
FESX424 Router(config-bgp-router)# write memory
```

Syntax: [no] neighbor <ip-addr> shutdown

The <ip-addr> parameter specifies the IP address of the neighbor.

Optional Configuration Tasks

The following sections describe how to perform optional BGP4 configuration tasks.

Changing the Keep Alive Time and Hold Time

The Keep Alive Time specifies how frequently the router will send KEEPALIVE messages to its BGP4 neighbors. The Hold Time specifies how long the router will wait for a KEEPALIVE or UPDATE message from a neighbor before concluding that the neighbor is dead. When the router concludes that a BGP4 neighbor is dead, the router ends the BGP4 session and closes the TCP connection to the neighbor.

The default Keep Alive time is 60 seconds. The default Hold Time is 180 seconds. To change the timers, use either of the following methods.

NOTE: Generally, you should set the Hold Time to three times the value of the Keep Alive Time.

NOTE: You can override the global Keep Alive Time and Hold Time on individual neighbors. See "Adding BGP4 Neighbors" on page 21-12.

To change the Keep Alive Time to 30 and Hold Time to 90, enter the following command:

```
FESX424 Router(config-bgp-router)# timers keep-alive 30 hold-time 90
```

Syntax: timers keep-alive <num> hold-time <num>

For each keyword, <num> indicates the number of seconds. The Keep Alive Time can be 0 – 65535. The Hold Time can be 0 or 3 – 65535 (1 and 2 are not allowed). If you set the Hold Time to 0, the router waits indefinitely for messages from a neighbor without concluding that the neighbor is dead.

Changing the BGP4 Next-Hop Update Timer

By default, the Layer 3 Switch updates its BGP4 next-hop tables and affected BGP4 routes five seconds after IGP route changes. You can change the update timer to a value from 1 – 30 seconds.

To change the BGP4 update timer value, enter a command such as the following at the BGP configuration level of the CLI:

```
FESX424 Router(config-bgp-router)# update-time 15
```

This command changes the update timer to 15 seconds.

Syntax: [no] update-time <secs>

The <secs> parameter specifies the number of seconds and can be from 1 – 30. The default is 5.

Enabling Fast External Fallover

BGP4 routers rely on KEEPALIVE and UPDATE messages from neighbors to signify that the neighbors are alive. For BGP4 neighbors that are two or more hops away, such messages are the only indication that the BGP4 protocol has concerning the alive state of the neighbors. As a result, if a neighbor dies, the router will wait until the Hold Time expires before concluding that the neighbor is dead and closing its BGP4 session and TCP connection with the neighbor.

The router waits for the Hold Time to expire before ending the connection to a directly-attached BGP4 neighbor that dies.

For directly attached neighbors, the router immediately senses loss of a connection to the neighbor from a change of state of the port or interface that connects the router to its neighbor. For directly attached EBGP neighbors, the router can use this information to immediately close the BGP4 session and TCP connection to locally attached neighbors that die.

NOTE: The fast external fallover feature applies only to directly attached EBGP neighbors. The feature does not apply to IBGP neighbors.

If you want to enable the router to immediately close the BGP4 session and TCP connection to locally attached neighbors that die, use either of the following methods.

To enable fast external fallover, enter the following command:

```
FESX424 Router(config-bgp-router)# fast-external-fallover
```

To disable fast external fallover again, enter the following command:

```
FESX424 Router(config-bgp-router)# no fast-external-fallover
```

Syntax: [no] fast-external-fallover

Changing the Maximum Number of Paths for BGP4 Load Sharing

Load sharing enables the Layer 3 Switch to balance traffic to a route across multiple equal-cost paths of the same type (EBGP or IBGP) for the route.

To configure the Layer 3 Switch to perform BGP4 load sharing:

- Enable IP load sharing if it is disabled.
- Set the maximum number of paths. The default maximum number of BGP4 load sharing paths is 1, which means no BGP4 load sharing takes place by default.

  NOTE: The maximum number of BGP4 load sharing paths cannot be greater than the maximum number of IP load sharing paths.

How Load Sharing Affects Route Selection

During evaluation of multiple paths to select the best path to a given destination for installment in the IP route table, the last comparison the Layer 3 Switch performs is a comparison of the internal paths.

- When IP load sharing is disabled, the Layer 3 Switch prefers the path to the router with the lower router ID.
- When IP load sharing and BGP4 load sharing are enabled, the Layer 3 Switch balances the traffic across the multiple paths instead of choosing just one path based on router ID.

See “How BGP4 Selects a Path for a Route” on page 21-4 for a description of the BGP4 algorithm.

When you enable IP load sharing, the Layer 3 Switch can load balance BGP4 or OSPF routes across up to four equal paths by default. You can change the number of IP load sharing paths to a value from 2 – 6.

How Load Sharing Works

Load sharing is performed in round-robin fashion and is based on the destination IP address only. The first time the router receives a packet destined for a specific IP address, the router uses a round-robin algorithm to select the path that was not used for the last newly learned destination IP address. Once the router associates a path with a particular destination IP address, the router will always use that path as long as the router contains the destination IP address in its cache.

NOTE: The Layer 3 Switch does not perform source routing. The router is concerned only with the paths to the next-hop routers, not the entire paths to the destination hosts.

A BGP4 destination can be learned from multiple BGP4 neighbors, leading to multiple BGP4 paths to reach the same destination. Each of the paths may be reachable through multiple IGP paths (multiple OSPF or RIP paths). In this case, the software installs all the multiple equal-cost paths in the BGP4 route table, up to the maximum number of BGP4 equal-cost paths allowed. The IP load sharing feature then distributes traffic across the equal-cost paths to the destination.

If an IGP path used by a BGP4 next-hop route path installed in the IP route table changes, then the BGP4 paths and IP paths are adjusted accordingly. For example, if one of the OSPF paths to reach the BGP4 next hop goes down, the software removes this path from the BGP4 route table and the IP route table. Similarly, if an additional OSPF path becomes available to reach the BGP4 next-hop router for a particular destination, the software adds the additional path to the BGP4 route table and the IP route table.

Changing the Maximum Number of Shared BGP4 Paths

When IP load sharing is enabled, BGP4 can balance traffic to a specific destination across up to four equal paths. You can set the maximum number of paths to a value from 1 – 4. The default is 1.

NOTE: The maximum number of BGP4 load sharing paths cannot be greater than the maximum number of IP load sharing paths. To increase the maximum number of IP load sharing paths, use the **ip load sharing** <num> command at the global CONFIG level of the CLI.

To change the maximum number of shared paths, enter commands such as the following:

```
FESX424 Router(config)# router bgp
FESX424 Router(config-bgp-router)# maximum-paths 4
FESX424 Router(config-bgp-router)# write memory
```

Syntax: [no] maximum-paths <num>

The <num> parameter specifies the maximum number of paths across which the Layer 3 Switch can balance traffic to a given BGP4 destination. You can change the maximum number of paths to a value from 2 – 4. The default is 1.

Customizing BGP4 Load Sharing

By default, when BGP4 load sharing is enabled, both IBGP and EBGP paths are eligible for load sharing, while paths from different neighboring ASs are not eligible. You can change load sharing to apply only to IBGP or EBGP paths, or to support load sharing among paths from different neighboring ASs.

To enable load sharing of IBGP paths only, enter the following command at the BGP configuration level of the CLI:

```
FESX424 Router(config-bgp-router)# multipath ibgp
```

To enable load sharing of EBGP paths only, enter the following command at the BGP configuration level of the CLI:

```
FESX424 Router(config-bgp-router)# multipath ebgp
```

To enable load sharing of paths from different neighboring ASs, enter the following command at the BGP configuration level of the CLI:

```
FESX424 Router(config-bgp-router)# multipath multi-as
```

Syntax: [no] multipath ebgp | ibgp | multi-as

The **ebgp | ibgp | multi-as** parameter specifies the change you are making to load sharing:

- **ebgp** – Load sharing applies only to EBGP paths. Load sharing is disabled for IBGP paths.
- **ibgp** – Load sharing applies only to IBGP paths. Load sharing is disabled for EBGP paths.
- **multi-as** – Load sharing is enabled for paths from different ASs.

By default, load sharing applies to EBGP and IBGP paths, and does not apply to paths from different neighboring ASs.

Specifying a List of Networks to Advertise

By default, the router sends BGP4 routes only for the networks you identify using the **network** command or that are redistributed into BGP4 from RIP or OSPF. You can specify up to 600 networks.

To specify a network to be advertised, use either of the following methods.

NOTE: The exact route must exist in the IP route table before the Layer 3 Switch can create a local BGP route.

To configure the Layer 3 Switch to advertise network 209.157.22.0/24, enter the following command:

```
FESX424 Router(config-bgp-router)# network 209.157.22.0 255.255.255.0
```

Syntax: network <ip-addr> <ip-mask> [nlri multicast | unicast | multicast unicast]
[route-map <map-name>] | [weight <num>] | [backdoor]

The <ip-addr> is the network number and the <ip-mask> specifies the network mask.

The **nlri multicast | unicast | multicast unicast** parameter specifies whether the neighbor is a multicast neighbor or a unicast neighbor. For MBGP, you must specify **multicast**. Optionally, you also can specify **unicast** if you want the Layer 3 Switch to exchange unicast (BGP4) routes as well as multicast routes with the neighbor. The default is **unicast** only.

The **route-map** <map-name> parameter specifies the name of the route map you want to use to set or change BGP4 attributes for the network you are advertising. The route map must already be configured.

The **weight** <num> parameter specifies a weight to be added to routes to this network.

The **backdoor** parameter changes the administrative distance of the route to this network from the EBGP administrative distance (20 by default) to the Local BGP weight (200 by default), thus tagging the route as a backdoor route. Use this parameter when you want the router to prefer IGP routes such as RIP or OSPF routes over the EBGP route for the network.

Specifying a Route Map Name when Configuring BGP4 Network Information

You can specify a route map as one of the parameters when you configure a BGP4 network to be advertised. The Layer 3 Switch can use the route map to set or change BGP4 attributes when creating a local BGP4 route.

To configure network information and use a route map to set or change BGP4 attributes, use the following CLI method.

NOTE: You must configure the route map before you can specify the route map name in a BGP4 network configuration.

To configure a route map, and use it to set or change route attributes for a network you define for BGP4 to advertise, enter commands such as the following:

```
FESX424 Router(config)# route-map set_net permit 1
FESX424 Router(config-routemap set_net)# set community no-export
FESX424 Router(config-routemap set_net)# exit
FESX424 Router(config)# router bgp
FESX424 Router(config-bgp-router)# network 100.100.1.0/24 route-map set_net
```

The first two commands in this example create a route map named “set_net” that sets the community attribute for routes that use the route map to “NO_EXPORT”. The next two commands change the CLI to the BGP4 configuration level. The last command configures a network for advertising from BGP4, and associates the “set_net” route map with the network. When BGP4 originates the 100.100.1.0/24 network, BGP4 also sets the community attribute for the network to “NO_EXPORT”.

Syntax: network <ip-addr> <ip-mask> [route-map <map-name>] | [weight <num>] | [backdoor]

The **route-map** <map-name> parameter specifies the name of the route map you want to use to set or change BGP4 attributes for the network you are advertising. The route map must already be configured.

For information about the other parameters, see “Defining Route Maps” on page 21-48.

Changing the Default Local Preference

When the router uses the BGP4 algorithm to select a route to send to the IP route table, one of the parameters the algorithm uses is the local preference. Local preference is an attribute that indicates a degree of preference for a route relative to other routes. BGP4 neighbors can send the local preference value as an attribute of a route in an UPDATE message.

Local preference applies only to routes within the local AS. BGP4 routers can exchange local preference information with neighbors who also are in the local AS, but BGP4 routers do not exchange local preference information with neighbors in remote ASs.

The default local preference is 100. For routes learned from EBGP neighbors, the default local preference is assigned to learned routes. For routes learned from IBGP neighbors, the local preference value is not changed for the route.

When the BGP4 algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen.

NOTE: To set the local preference for individual routes, use route maps. See “Defining Route Maps” on page 21-48. See “How BGP4 Selects a Path for a Route” on page 21-4 for information about the BGP4 algorithm.

To change the default local preference to 200, enter the following command:

```
FESX424 Router(config-bgp-router)# default-local-preference 200
```

Syntax: default-local-preference <num>

The <num> parameter indicates the preference and can be a value from 0 – 4294967295.

Using the IP Default Route as a Valid Next Hop for a BGP4 Route

By default, the Layer 3 Switch does not use a default route to resolve a BGP4 next-hop route. If the IP route lookup for the BGP4 next hop does not result in a valid IGP route (including static or direct routes), the BGP4 next hop is considered to be unreachable and the BGP4 route is not used.

In some cases, such as when the Layer 3 Switch is acting as an edge router, you might want to allow the device to use the default route as a valid next hop. To do so, enter the following command at the BGP4 configuration level of the CLI:

```
FESX424 Router(config-bgp-router)# next-hop-enable-default
```

Syntax: [no] next-hop-enable-default

Advertising the Default Route

By default, the Layer 3 Switch does not originate and advertise a default route using BGP4. A BGP4 default route is the IP address 0.0.0.0 and the route prefix 0 or network mask 0.0.0.0. For example, 0.0.0.0/0 is a default route. You can enable the router to advertise a default BGP4 route using either of the following methods.

NOTE: The Foundry Layer 3 Switch checks for the existence of an IGP route for 0.0.0.0/0 in the IP route table before creating a local BGP route for 0.0.0.0/0.

To enable the router to originate and advertise a default BGP4 route, enter the following command:

```
FESX424 Router(config-bgp-router)# default-information-originate
```

Syntax: [no] default-information-originate

Changing the Default MED (Metric) Used for Route Redistribution

The Foundry Layer 3 Switch can redistribute directly connected routes, static IP routes, RIP routes, and OSPF routes into BGP4. The MED (metric) is a global parameter that specifies the cost that will be applied to all routes by default when they are redistributed into BGP4. When routes are selected, lower metric values are preferred over higher metric values. The default BGP4 MED value is 0 and can be assigned a value from 0 – 4294967295.

NOTE: RIP and OSPF also have default metric parameters. The parameters are set independently for each protocol and have different ranges.

To change the default metric to 40, enter the following command:

```
FastIron SuperX Router(config-bgp-router)# default-metric 40
```

Syntax: default-metric <num>

The <num> indicates the metric and can be a value from 0 – 4294967295.

Enabling Next-Hop Recursion

For each BGP4 route a Layer 3 Switch learns, the Layer 3 Switch performs a route lookup to obtain the IP address of the route's next hop. A BGP4 route becomes eligible for installation into the IP route table only if the following conditions are true:

- The lookup succeeds in obtaining a valid next-hop IP address for the route.
- The path to the next-hop IP address is an Interior Gateway Protocol (IGP) path or a static route path.

By default, the software performs only one lookup for a BGP route's next-hop IP address. If the next-hop lookup does not result in a valid next-hop IP address or the path to the next-hop IP address is a BGP path, the software considers the BGP route's destination to be unreachable. The route is not eligible to be installed in the IP route table.

It is possible for the BGP route table to contain a route whose next-hop IP address is not reachable through an IGP route, even though a hop farther away can be reached by the Layer 3 Switch through an IGP route. This can occur when the IGPs do not learn a complete set of IGP routes, resulting in the Layer 3 Switch learning about an internal route through IBGP instead of through an IGP. In this case, the IP route table does not contain a route that can be used to reach the BGP route's destination.

To enable the Layer 3 Switch to find the IGP route to a BGP route's next-hop gateway, enable recursive next-hop lookups. When you enable recursive next-hop lookup, if the first lookup for a BGP route results in an IBGP path originated within the same Autonomous System (AS), rather than an IGP path or static route path, the Layer 3 Switch performs a lookup on the next-hop gateway's next-hop IP address. If this second lookup results in an IGP path, the software considers the BGP route to be valid and thus eligible for installation in the IP route table. Otherwise, the Layer 3 Switch performs a lookup on the next-hop IP address of the next-hop gateway's next hop, and so on, until one of the lookups results in an IGP route.

NOTE: The software does not support using the default route to resolve a BGP4 route's next hop. Instead, you must configure a static route or use an IGP to learn the route to the EBGP multihop peer.

Previous software releases support use of the default route to resolve routes learned from EBGP multihop neighbors. However, even in this case Foundry recommends that you use a static route for the EBGP multihop neighbor instead. In general, we recommend that you do not use the default route as the next hop for BGP4 routes, especially when there are two or more BGP4 neighbors. Using the default route can cause loops.

Example When Recursive Route Lookups Are Disabled

Here is an example of the results of an unsuccessful next-hop lookup for a BGP route. In this case, next-hop recursive lookups are disabled. The example is for the BGP route to network 240.0.0.0/24.

```
FastIron SuperX Router# show ip bgp route
Total number of BGP Routes: 5
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
       Prefix             Next Hop        Metric     LocPrf      Weight Status
1      0.0.0.0/0           10.1.0.2        0          100         0      BI
         AS_PATH: 65001 4355 701 80
2      102.0.0.0/24         10.0.0.1       1          100         0      BI
         AS_PATH: 65001 4355 1
3      104.0.0.0/24         10.1.0.2       0          100         0      BI
         AS_PATH: 65001 4355 701 1 189
4      240.0.0.0/24         102.0.0.1      1          100         0      I
         AS_PATH: 65001 4355 3356 7170 1455
5      250.0.0.0/24         209.157.24.1   1          100         0      I
         AS_PATH: 65001 4355 701
```

In this example, the Layer 3 Switch cannot reach 240.0.0.0/24, because the next-hop IP address for the route is an IBGP route instead of an IGP route, and thus is considered unreachable by the Layer 3 Switch. Here is the IP route table entry for the BGP route's next-hop gateway (102.0.0.1/24):

```
FastIron SuperX Router# show ip route 102.0.0.1
Total number of IP routes: 37
      Network Address   NetMask          Gateway          Port    Cost   Type
      102.0.0.0         255.255.255.0    10.0.0.1         1/1     1      B
```

The route to the next-hop gateway is a BGP route, not an IGP route, and thus cannot be used to reach 240.0.0.0/24. In this case, the Layer 3 Switch tries to use the default route, if present, to reach the sub-net that contains the BGP route's next-hop gateway.

```
FastIron SuperX Router# show ip route 240.0.0.0/24
Total number of IP routes: 37
      Network Address   NetMask          Gateway          Port    Cost   Type
      0.0.0.0           0.0.0.0          10.0.0.202       1/1     1      S
```

Example When Recursive Route Lookups Are Enabled

When recursive next-hop lookups are enabled, the Layer 3 Switch recursively looks up the next-hop gateways along the route until the Layer 3 Switch finds an IGP route to the BGP route's destination. Here is an example.

```
FastIron SuperX Router# show ip bgp route
Total number of BGP Routes: 5
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
       Prefix         Next Hop        Metric     LocPrf      Weight Status
1      0.0.0.0/0       10.1.0.2       0          100         0      BI
         AS_PATH: 65001 4355 701 80
2      102.0.0.0/24    10.0.0.1       1          100         0      BI
         AS_PATH: 65001 4355 1
3      104.0.0.0/24    10.1.0.2       0          100         0      BI
         AS_PATH: 65001 4355 701 1 189
4      240.0.0.0/24    102.0.0.1      1          100         0      BI
         AS_PATH: 65001 4355 3356 7170 1455
5      250.0.0.0/24    209.157.24.1   1          100         0       I
         AS_PATH: 65001 4355 701
```

The first lookup results in an IBGP route, to network 102.0.0.0/24:

```
FastIron SuperX Router# show ip route 102.0.0.1
Total number of IP routes: 38
      Network Address   NetMask          Gateway          Port    Cost    Type
      102.0.0.0         255.255.255.0    10.0.0.1         1/1     1       B
         AS_PATH: 65001 4355 1
```

Since the route to 102.0.0.1/24 is not an IGP route, the Layer 3 Switch cannot reach the next hop through IP, and thus cannot use the BGP route. In this case, since recursive next-hop lookups are enabled, the Layer 3 Switch next performs a lookup for 102.0.0.1's next-hop gateway, 10.0.0.1:

```
FastIron SuperX Router# show ip bgp route 102.0.0.0
Number of BGP Routes matching display condition : 1
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
       Prefix         Next Hop        Metric     LocPrf      Weight Status
1      102.0.0.0/24    10.0.0.1       1          100         0      BI
         AS_PATH: 65001 4355 1
```

The next-hop IP address for 102.0.0.1 is not an IGP route, which means the BGP route's destination still cannot be reached through IP. The recursive next-hop lookup feature performs a lookup on 10.0.0.1's next-hop gateway:

```
FastIron SuperX Router# show ip route 10.0.0.1
Total number of IP routes: 38
      Network Address   NetMask          Gateway          Port    Cost    Type
      10.0.0.0          255.255.255.0    0.0.0.0          1/1     1       D
        AS_PATH: 65001 4355 1
```

This lookup results in an IGP route. In fact, this route is a directly-connected route. As a result, the BGP route's destination is now reachable through IGP, which means the BGP route is eligible for installation in the IP route table. Here is the BGP route in the IP route table:

```
FastIron SuperX Router# show ip route 240.0.0.0/24
Total number of IP routes: 38
      Network Address    NetMask          Gateway          Port     Cost    Type
      240.0.0.0          255.255.255.0    10.0.0.1         1/1      1       B
         AS_PATH: 65001 4355 1
```

This Layer 3 Switch can use this route because the Layer 3 Switch has an IP route to the next-hop gateway. Without recursive next-hop lookups, this route would not be in the IP route table.

Enabling Recursive Next-Hop Lookups

The recursive next-hop lookups feature is disabled by default. To enable recursive next-hop lookups, enter the following command at the BGP configuration level of the CLI:

```
FastIron SuperX Router(config-bgp-router)# next-hop-recursion
```

Syntax: [no] next-hop-recursion

Changing Administrative Distances

BGP4 routers can learn about networks from various protocols, including the EBGP portion of BGP4 and IGPs such as OSPF and RIP. Consequently, the routes to a network may differ depending on the protocol from which the routes were learned.

To select one route over another based on the source of the route information, the Layer 3 Switch can use the administrative distances assigned to the sources. The administrative distance is a protocol-independent metric that IP routers use to compare routes from different sources.

The Layer 3 Switch re-advertises a learned best BGP4 route to the Layer 3 Switch's neighbors even when the software does not also select that route for installation in the IP route table. The best BGP4 routes is the BGP4 path that the software selects based on comparison of the paths' BGP4 route parameters. See "How BGP4 Selects a Path for a Route" on page 21-4.

When selecting a route from among different sources (BGP4, OSPF, RIP, static routes, and so on), the software compares the routes on the basis of each route's administrative distance. If the administrative distance of the paths is lower than the administrative distance of paths from other sources (such as static IP routes, RIP, or OSPF), the BGP4 paths are installed in the IP route table.

NOTE: The software will replace a statically configured default route with a learned default route if the learned route's administrative distance is lower than the statically configured default route's distance. However, the default administrative distance for static routes is changed to 1, so only directly-connected routes are preferred over static routes when the default administrative distances for the routes are used.

Here are the default administrative distances on the Foundry Layer 3 Switch:

- Directly connected – 0 (this value is not configurable)
- Static – 1 (applies to all static routes, including default routes)
- EBGP – 20
- OSPF – 110
- RIP – 120
- IBGP – 200
- Local BGP – 200
- Unknown – 255 (the router will not use this route)

Lower administrative distances are preferred over higher distances. For example, if the router receives routes for the same network from OSPF and from RIP, the router will prefer the OSPF route by default. The administrative distances are configured in different places in the software. The Layer 3 Switch re-advertises a learned best BGP4 route to neighbors by default, regardless of whether the route's administrative distance is lower than other routes from different route sources to the same destination.

- To change the EBGP, IBGP, and Local BGP default administrative distances, see the instructions in this section.
- To change the default administrative distance for OSPF, see "Modify Administrative Distance" on page 20-34.
- To change the default administrative distance for RIP, see "Changing the Administrative Distance" on page 17-5.
- To change the default administrative distance for static routes, see "Configuring Static Routes" on page 16-32.

You can change the default EBGP, IBGP, and Local BGP administrative distances using either of the following methods.

To change the default administrative distances for EBGP, IBGP, and Local BGP, enter a command such as the following:

```
FESX424 Router(config-bgp-router)# distance 180 160 40
```

Syntax: distance <external-distance> <internal-distance> <local-distance>

The <external-distance> sets the EBGP distance and can be a value from 1 – 255.

The <internal-distance> sets the IBGP distance and can be a value from 1 – 255.

The <local-distance> sets the Local BGP distance and can be a value from 1 – 255.

Requiring the First AS to be the Neighbor's AS

By default, the Foundry device does not require the first AS listed in the AS_SEQUENCE field of an AS path Update from an EBGP neighbor to be the AS that the neighbor who sent the Update is in. You can enable the Foundry device for this requirement.

When you enable the Foundry device to require the AS that an EBGP neighbor is in to be the same as the first AS in the AS_SEQUENCE field of an Update from the neighbor, the Foundry device accepts the Update only if the ASs match. If the ASs do not match, the Foundry device sends a Notification message to the neighbor and closes the session. The requirement applies to all Updates received from EBGP neighbors.

To enable this feature, enter the following command at the BGP configuration level of the CLI:

```
FESX424 Router(config-bgp-router)# enforce-first-as
```

Syntax: [no] enforce-first-as

Disabling or Re-Enabling Comparison of the AS-Path Length

AS-Path comparison is Step 5 in the algorithm BGP4 uses to select the next path for a route. Comparison of the AS-Path length is enabled by default. To disable it, enter the following command at the BGP configuration level of the CLI:

```
FESX424 Router(config-bgp-router)# as-path-ignore
```

This command disables comparison of the AS-Path lengths of otherwise equal paths. When you disable AS-Path length comparison, the BGP4 algorithm shown in "How BGP4 Selects a Path for a Route" on page 21-4 skips from Step 4 to Step 6.

Syntax: [no] as-path-ignore

Enabling or Disabling Comparison of the Router IDs

Router ID comparison is Step 10 in the algorithm BGP4 uses to select the next path for a route.

NOTE: Comparison of router IDs is applicable only when BGP4 load sharing is disabled.

When router ID comparison is enabled, the path comparison algorithm compares the router IDs of the neighbors that sent the otherwise equal paths.

- If BGP4 load sharing is disabled (maximum-paths 1), the Layer 3 Switch selects the path that came from the neighbor with the lower router ID.
- If BGP4 load sharing is enabled, the Layer 3 Switch load shares among the remaining paths. In this case, the router ID is not used to select a path.

NOTE: Router ID comparison is disabled by default2. In previous releases, router ID comparison is enabled by default and cannot be disabled.

To enable router ID comparison, enter the following command at the BGP configuration level of the CLI:

```
FESX424 Router(config-bgp-router)# compare-routerid
```

Syntax: [no] compare-routerid

For more information, see “How BGP4 Selects a Path for a Route” on page 21-4.

Configuring the Layer 3 Switch To Always Compare Multi-Exit Discriminators (MEDs)

A Multi-Exit Discriminator (MED) is a value that the BGP4 algorithm uses when comparing multiple paths received from different BGP4 neighbors in the same AS for the same route. In BGP4, a route’s MED is equivalent to its “metric”.

- BGP4 compares the MEDs of two otherwise equivalent paths ***if and only if*** the routes were learned from the same neighboring AS. This behavior is called ***deterministic MED***. 0Deterministic MED is always enabled and cannot be disabled.

  In addition, you can enable the Layer 3 Switch to always compare the MEDs, regardless of the AS information in the paths. To enable this comparison, enter the **always-compare-med** command at the BGP4 configuration level of the CLI. This option is disabled by default.

- The Layer 3 Switch compares the MEDs based on one or more of the following conditions. By default, the Layer 3 Switch compares the MEDs of paths ***only if*** the first AS in the paths is the same. (The Layer 3 Switch skips over the AS-CONFED-SEQUENCE if present.)

You can enable the Layer 3 Switch to always compare the MEDs, regardless of the AS information in the paths. For example, if the router receives UPDATES for the same route from neighbors in three ASs, the router would compare the MEDs of all the paths together, rather than comparing the MEDs for the paths in each AS individually.

NOTE: By default, value 0 (most favorable) is used in MED comparison when the MED attribute is not present. The default MED comparison results in the Layer 3 Switch favoring the route paths that are missing their MEDs. You can use the **med-missing-as-worst** command to make the Layer 3 Switch regard a BGP route with a missing MED attribute as the least favorable route, when comparing the MEDs of the routes.

NOTE: MED comparison is not performed for internal routes originated within the local AS or confederation.

To configure the router to always compare MEDs, enter the following command:

```
FESX424 Router(config-bgp-router)# always-compare-med
```

Syntax: [no] always-compare-med

Treating Missing MEDs as the Worst MEDs

By default, the Layer 3 Switch favors a lower MED over a higher MED during MED comparison. Since the Layer 3 Switch assigns the value 0 to a route path's MED if the MED value is missing, the default MED comparison results in the Layer 3 Switch favoring the route paths that are missing their MEDs.

To change this behavior so that the Layer 3 Switch favors a route that has a MED over a route that is missing its MED, enter the following command at the BGP4 configuration level of the CLI:

```
FESX424 Router(config-bgp-router)# med-missing-as-worst
```

Syntax: [no] med-missing-as-worst

NOTE: This command affects route selection only when route paths are selected based on MED comparison. It is still possible for a route path that is missing its MED to be selected based on other criteria. For example, a route path with no MED can be selected if its weight is larger than the weights of the other route paths.

Configuring Route Reflection Parameters

Normally, all the BGP routers within an AS are fully meshed. Each of the routers has an IBGP session with each of the other BGP routers in the AS. Each IBGP router thus has a route for each of its IBGP neighbors. For large ASs containing many IBGP routers, the IBGP route information in each of the fully-meshed IBGP routers can introduce too much administrative overhead.

To avoid this problem, you can hierarchically organize your IGP routers into clusters.

- A ***cluster*** is a group of IGP routers organized into route reflectors and route reflector clients. You configure the cluster by assigning a cluster ID on the route reflector and identifying the IGP neighbors that are members of that cluster. All the configuration for route reflection takes place on the route reflectors. The clients are unaware that they are members of a route reflection cluster. All members of the cluster must be in the same AS. The cluster ID can be any number from 1 – 4294967295. The default is the router ID, expressed as a 32-bit number.

  NOTE: If the cluster contains more than one route reflector, you need to configure the same cluster ID on all the route reflectors in the cluster. The cluster ID helps route reflectors avoid loops within the cluster.

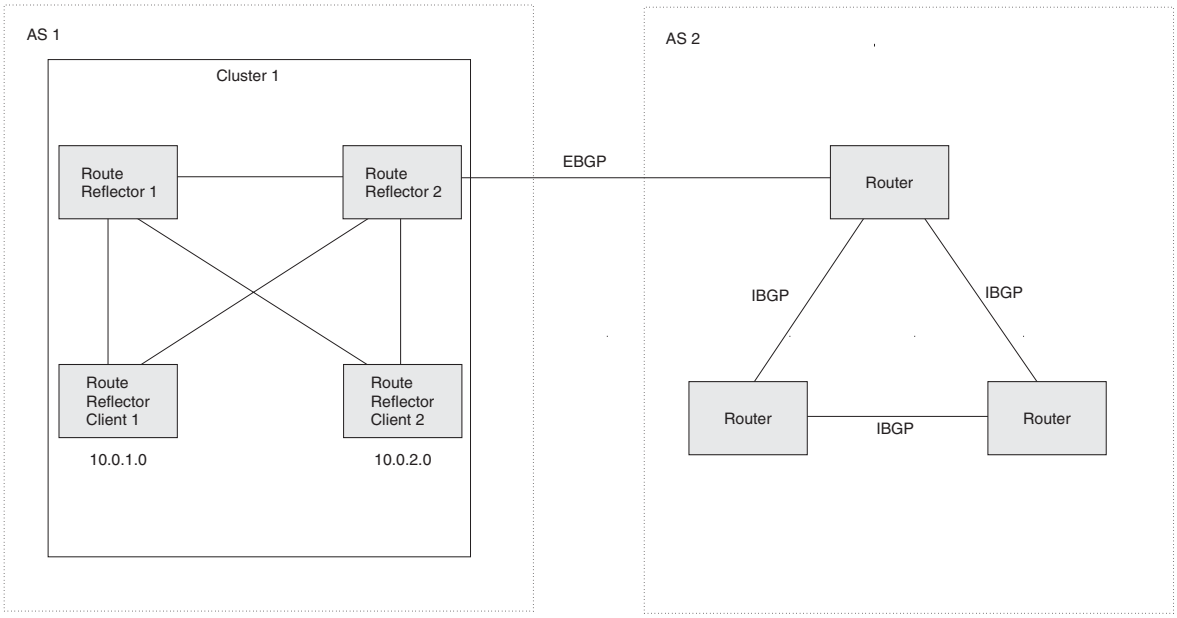
- A ***route reflector*** is an IGP router configured to send BGP route information to all the clients (other BGP4 routers) within the cluster. Route reflection is enabled on all Foundry BGP4 routers by default but does not take effect unless you add route reflector clients to the router.
- A ***route reflector client*** is an IGP router identified as a member of a cluster. You identify a router as a route reflector client on the router that is the route reflector, not on the client. The client itself requires no additional configuration. In fact, the client does not know that it is a route reflector client. The client just knows that it receives updates from its neighbors and does not know whether one or more of those neighbors are route reflectors.

NOTE: Route reflection applies only among IBGP routers within the same AS. You cannot configure a cluster that spans multiple ASs.

Figure 21.2 shows an example of a route reflector configuration. In this example, two Layer 3 Switches are configured as route reflectors for the same cluster. The route reflectors provide redundancy in case one of the reflectors becomes unavailable. Without redundancy, if a route reflector becomes unavailable, its clients are cut off from BGP4 updates.

AS1 contains a cluster with two route reflectors and two clients. The route reflectors are fully meshed with other BGP4 routers, but the clients are not fully meshed. They rely on the route reflectors to propagate BGP4 route updates.

Figure 21.2 Example route reflector configuration



Support for RFC 2796

Route reflection on Foundry devices is based on RFC 2796. This updated RFC helps eliminate routing loops that are possible in some implementations of the older specification, RFC 1966.

NOTE: The configuration procedure for route reflection is the same regardless of whether your software release is using RFC 1966 or RFC 2796. However, the operation of the feature is different as explained below.

RFC 2796 provides more details than RFC 1966 regarding the use of the route reflection attributes, ORIGINATOR_ID and CLUSTER_LIST, to help prevent loops.

- ORIGINATOR_ID – Specifies the router ID of the BGP4 router that originated the route. The route reflector inserts this attribute when reflecting a route to an IBGP neighbor. If a BGP4 router receives an advertisement that contains its own router ID as the ORIGINATOR_ID, the router discards the advertisement and does not forward it.
- CLUSTER_LIST – A list of the route reflection clusters through which the advertisement has passed. A cluster contains a route reflector and its clients. When a route reflector reflects a route, the route reflector adds its cluster ID to the front of the CLUSTER_LIST. If a route reflector receives a route that has its own cluster ID, the router discards the advertisement and does not forward it.

The Foundry device handles the attributes as follows:

- The Layer 3 Switch adds the attributes only if it is a route reflector, and only when advertising IBGP route information to other IBGP neighbors. The attributes are not used when communicating with EBGP neighbors.
- A Layer 3 Switch configured as a route reflector sets the ORIGINATOR_ID attribute to the router ID of the router that originated the route. Moreover, the route reflector sets the attribute only if this is the first time the route is being reflected (sent by a route reflector). In previous software releases, the route reflector set the attribute to the router ID of the route reflector itself. When a Layer 3 Switch receives a route that already has the ORIGINATOR_ID attribute set, the Layer 3 Switch does not change the value of the attribute.
- If a Layer 3 Switch receives a route whose ORIGINATOR_ID attribute has the value of the Layer 3 Switch's own router ID, the Layer 3 Switch discards the route and does not advertise it. By discarding the route, the Layer 3 Switch prevents a routing loop. The Layer 3 Switch did not discard the route in previous software releases.
- The first time a route is reflected by a Layer 3 Switch configured as a route reflector, the route reflector adds

the CLUSTER_LIST attribute to the route. Other route reflectors who receive the route from an IBGP neighbor add their cluster IDs to the front of the route's CLUSTER_LIST. If the route reflector does not have a cluster ID configured, the Layer 3 Switch adds its router ID to the front of the CLUSTER_LIST.

- If Layer 3 Switch configured as a route reflector receives a route whose CLUSTER_LIST contains the route reflector's own cluster ID, the route reflector discards the route and does not forward it.

Configuration Procedures

To configure a Foundry Layer 3 Switch to be a BGP4 route reflector, use either of the following methods.

NOTE: All configuration for route reflection takes place on the route reflectors, not on the clients.

Enter the following commands to configure a Foundry Layer 3 Switch as route reflector 1 in Figure 21.2 on page 21-33. To configure route reflector 2, enter the same commands on the Layer 3 Switch that will be route reflector 2. The clients require no configuration for route reflection.

```
FESX424 Router(config-bgp-router)# cluster-id 1
FESX424 Router(config-bgp-router)# neighbor 10.0.1.0 route-reflector-client
FESX424 Router(config-bgp-router)# neighbor 10.0.2.0 route-reflector-client
```

Syntax: [no] cluster-id <num> | <ip-addr>

The <num> | <ip-addr> parameter specifies the cluster ID and can be a number from 1 – 4294967295 or an IP address. The default is the router ID. You can configure one cluster ID on the router. All route-reflector clients for the router are members of the cluster.

NOTE: If the cluster contains more than one route reflector, you need to configure the same cluster ID on all the route reflectors in the cluster. The cluster ID helps route reflectors avoid loops within the cluster.

To add an IBGP neighbor to the cluster, enter the following command:

Syntax: neighbor <ip-addr> route-reflector-client

For more information about the **neighbor** command, see "Adding BGP4 Neighbors" on page 21-12.

By default, the clients of a route reflector are not required to be fully meshed; the routes from a client are reflected to other clients. However, if the clients are fully meshed, route reflection is not required between clients.

If you need to disable route reflection between clients, enter the following command. When the feature is disabled, route reflection does not occur between clients but reflection does still occur between clients and non-clients.

```
FESX424 Router(config-bgp-router)# no client-to-client-reflection
```

Enter the following command to re-enable the feature:

```
FESX424 Router(config-bgp-router)# client-to-client-reflection
```

Syntax: [no] client-to-client-reflection

Configuring Confederations

A ***confederation*** is a BGP4 Autonomous System (AS) that has been subdivided into multiple, smaller ASs. Subdividing an AS into smaller ASs simplifies administration and reduces BGP-related traffic, thus reducing the complexity of the Interior Border Gateway Protocol (IBGP) mesh among the BGP routers in the AS.

The Foundry implementation of this feature is based on RFC 3065.

Normally, all BGP routers within an AS must be fully meshed, so that each BGP router has interfaces to all the other BGP routers within the AS. This is feasible in smaller ASs but becomes unmanageable in ASs containing many BGP routers.

When you configure BGP routers into a confederation, all the routers within a sub-AS (a subdivision of the AS) use IBGP and must be fully meshed. However, routers use EBGP to communicate between different sub-ASs.

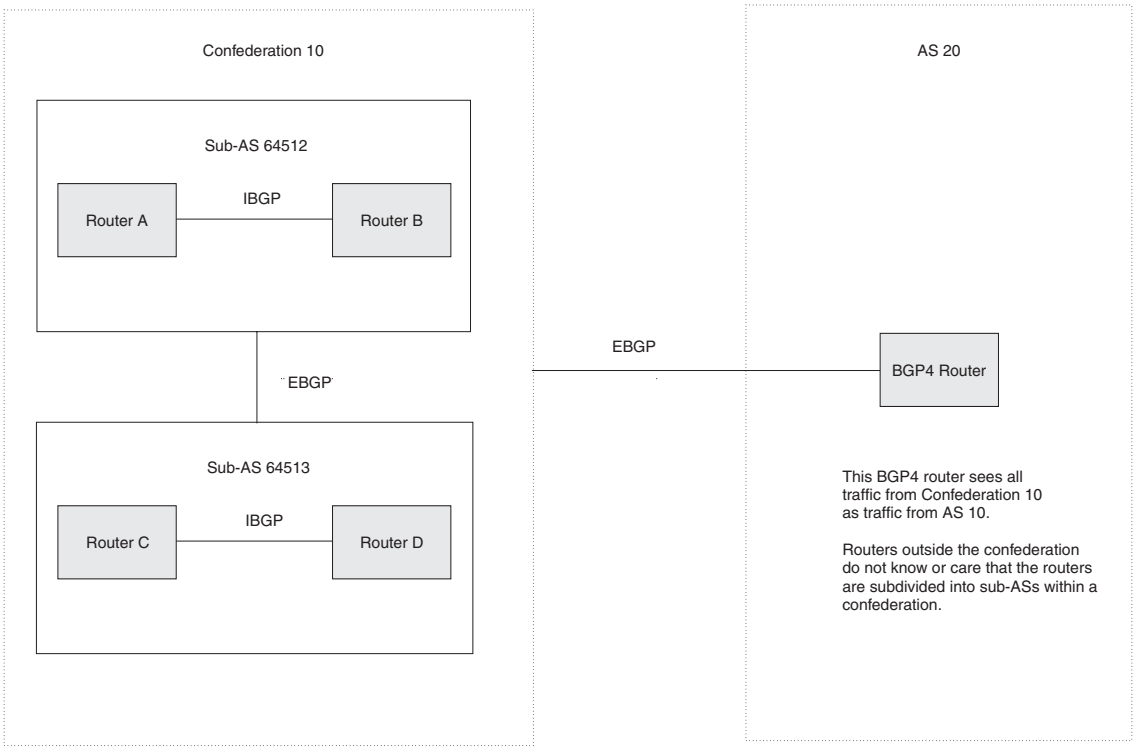
NOTE: Another method for reducing the complexity of an IBGP mesh is to use route reflection. However, if you want to run different Interior Gateway Protocols (IGPs) within an AS, configure a confederation. You can run a separate IGP within each sub-AS.

To configure a confederation, configure groups of BGP routers into sub-ASs. A sub-AS is simply an AS. The term "sub-AS" distinguishes ASs within a confederation from ASs that are not in a confederation. For the viewpoint of remote ASs, the confederation ID is the AS ID. Remote ASs do not know that the AS represents multiple sub-ASs with unique AS IDs.

NOTE: You can use any valid AS numbers for the sub-ASs. If your AS is connected to the Internet, Foundry recommends that you use numbers from within the private AS range (64512 – 65535). These are private ASs numbers and BGP4 routers do not propagate these AS numbers to the Internet.

Figure 21.3 shows an example of a BGP4 confederation.

Figure 21.3 Example BGP4 confederation



In this example, four routers are configured into two sub-ASs, each containing two of the routers. The sub-ASs are members of confederation 10. Routers within a sub-AS must be fully meshed and communicate using IBGP. In this example, routers A and B use IBGP to communicate. Routers C and D also use IBGP. However, the sub-ASs communicate with one another using EBGP. For example, router A communicates with router C using EBGP. The routers in the confederation communicate with other ASs using EBGP.

Routers in other ASs are unaware that routers A – D are configured in a confederation. In fact, when routers in confederation 10 send traffic to routers in other ASs, the confederation ID is the same as the AS number for the routers in the confederation. Thus, routers in other ASs see traffic from AS 10 and are unaware that the routers in AS 10 are subdivided into sub-ASs within a confederation.

Configuring a BGP Confederation

Perform the following configuration tasks on each BGP router within the confederation:

- Configure the local AS number. The local AS number indicates membership in a sub-AS. All BGP routers with the same local AS number are members of the same sub-AS. BGP routers use the local AS number when communicating with other BGP routers within the confederation.
- Configure the confederation ID. The confederation ID is the AS number by which BGP routers outside the confederation know the confederation. Thus, a BGP router outside the confederation is not aware and does not care that your BGP routers are in multiple sub-ASs. BGP routers use the confederation ID when communicating with routers outside the confederation. The confederation ID must be different from the sub-AS numbers.
- Configure the list of the sub-AS numbers that are members of the confederation. All the routers within the same sub-AS use IBGP to exchange router information. Routers in different sub-ASs within the confederation use EBGP to exchange router information.

To configure four Layer 3 Switches to be a member of confederation 10 (as shown in Figure 21.3), consisting of two sub-ASs (64512 and 64513), enter commands such as the following.

Commands for Router A

```
FESX424 RouterA(config)# router bgp
FESX424 RouterA(config-bgp-router)# local-as 64512
FESX424 RouterA(config-bgp-router)# confederation identifier 10
FESX424 RouterA(config-bgp-router)# confederation peers 64512 64513
FESX424 RouterA(config-bgp-router)# write memory
```

Syntax: local-as <num>

The <num> parameter with the **local-as** command indicates the AS number for the BGP routers within the sub-AS. You can specify a number from 1 – 65535. Foundry recommends that you use a number within the range of well-known private ASs, 64512 – 65535.

Syntax: confederation identifier <num>

The <num> parameter with the **confederation identifier** command indicates the confederation number. The confederation ID is the AS number by which BGP routers outside the confederation know the confederation. Thus, a BGP router outside the confederation is not aware and does not care that your BGP routers are in multiple sub-ASs. BGP routers use the confederation ID when communicating with routers outside the confederation. The confederation ID must be different from the sub-AS numbers. You can specify a number from 1 – 65535.

Syntax: confederation peers <num> [<num> …]

The <num> parameter with the **confederation peers** command indicates the sub-AS numbers for the sub-ASs in the confederation. You must specify all the sub-ASs contained in the confederation. All the routers within the same sub-AS use IBGP to exchange router information. Routers in different sub-ASs within the confederation use EBGP to exchange router information. You can specify a number from 1 – 65535.

Commands for Router B

```
FESX424 RouterB(config)# router bgp
FESX424 RouterB(config-bgp-router)# local-as 64512
FESX424 RouterB(config-bgp-router)# confederation identifier 10
FESX424 RouterB(config-bgp-router)# confederation peers 64512 64513
FESX424 RouterB(config-bgp-router)# write memory
```

Commands for Router C

```
FESX424 RouterC(config)# router bgp
FESX424 RouterC(config-bgp-router)# local-as 64513
FESX424 RouterC(config-bgp-router)# confederation identifier 10
FESX424 RouterC(config-bgp-router)# confederation peers 64512 64513
FESX424 RouterC(config-bgp-router)# write memory
```

Commands for Router D

```
FESX424 RouterD(config)# router bgp
FESX424 RouterD(config-bgp-router)# local-as 64513
FESX424 RouterD(config-bgp-router)# confederation identifier 10
FESX424 RouterD(config-bgp-router)# confederation peers 64512 64513
```



```
FESX424 RouterD(config-bgp-router)# write memory
```

Aggregating Routes Advertised to BGP4 Neighbors

By default, the Layer 3 Switch advertises individual routes for all the networks. The aggregation feature allows you to configure the Layer 3 Switch to aggregate routes in a range of networks into a single CIDR number. For example, without aggregation, the Layer 3 Switch will individually advertise routes for networks 207.95.1.0, 207.95.2.0, and 207.95.3.0. You can configure the Layer 3 Switch to instead send a single, aggregate route for the networks. The aggregate route would be advertised as 207.95.0.0.

NOTE: To summarize CIDR networks, you must use the aggregation feature. The auto summary feature does not summarize networks that use CIDR numbers instead of class A, B, or C numbers.

To aggregate routes for 209.157.22.0, 209.157.23.0, and 209.157.24.0, enter the following command:

```
FESX424 Router(config-bgp-router)# aggregate-address 209.157.0.0 255.255.0.0
```

Syntax: aggregate-address <ip-addr> <ip-mask> [as-set] [nlri multicast | unicast | multicast unicast] [summary-only] [suppress-map <map-name>] [advertise-map <map-name>] [attribute-map <map-name>]

The <ip-addr> and <ip-mask> parameters specify the aggregate value for the networks. Specify 0 for the host portion and for the network portion that differs among the networks in the aggregate. For example, to aggregate 10.0.1.0, 10.0.2.0, and 10.0.3.0, enter the IP address 10.0.0.0 and the network mask 255.255.0.0.

The **as-set** parameter causes the router to aggregate AS-path information for all the routes in the aggregate address into a single AS-path.

The **nlri multicast** | **unicast** | **multicast unicast** parameter specifies whether the neighbor is a multicast neighbor or a unicast neighbor. For MBGP, you must specify **multicast**. Optionally, you also can specify **unicast** if you want the Layer 3 Switch to exchange unicast (BGP4) routes as well as multicast routes with the neighbor. The default is **unicast** only.

The **summary-only** parameter prevents the router from advertising more specific routes contained within the aggregate route.

The **suppress-map** <map-name> parameter prevents the more specific routes contained in the specified route map from being advertised.

The **advertise-map** <map-name> parameter configures the router to advertise the more specific routes in the specified route map.

The **attribute-map** <map-name> parameter configures the router to set attributes for the aggregate routes based on the specified route map.

NOTE: For the **suppress-map**, **advertise-map**, and **attribute-map** parameters, the route map must already be defined. See “Defining Route Maps” on page 21-48 for information on defining a route map.

Modifying Redistribution Parameters

By default, the router does not redistribute route information between BGP4 and the IP IGPs (RIP and OSPF). You can configure the router to redistribute OSPF routes, RIP routes, directly connected routes, or static routes into BGP4 by using the following methods.

To enable redistribution of all OSPF routes and directly attached routes into BGP4, enter the following commands.

```
FESX424 Router(config)# router bgp
FESX424 Router(config-bgp-router)# redistribute ospf
```

```
FESX424 Router(config-bgp-router)# redistribute connected
FESX424 Router(config-bgp-router)# write memory
```

Syntax: [no] redistribute connected | ospf | rip | static

The **connected** parameter indicates that you are redistributing routes to directly attached devices into BGP.

The **ospf** parameter indicates that you are redistributing OSPF routes into BGP4.

NOTE: Entering **redistribute ospf** simply redistributes internal OSPF routes. If you want to redistribute external OSPF routes also, you must use the **redistribute ospf match external...** command. See "Redistributing OSPF External Routes" on page 21-39.

The **rip** parameter indicates that you are redistributing RIP routes into BGP4.

The **static** parameter indicates that you are redistributing static routes into BGP.

See the following sections for details on redistributing specific routes using the CLI:

- "Redistributing Connected Routes" on page 21-38
- "Redistributing RIP Routes" on page 21-38
- "Redistributing OSPF External Routes" on page 21-39
- "Redistributing Static Routes" on page 21-39

Redistributing Connected Routes

To configure BGP4 to redistribute directly connected routes, enter the following command:

```
FESX424 Router(config-bgp-router)# redistribute connected
```

Syntax: redistribute connected [metric <num>] [route-map <map-name>]

The **connected** parameter indicates that you are redistributing routes to directly attached devices into BGP4.

The **metric** <num> parameter changes the metric. You can specify a value from 0 – 4294967295. The default is 0.

The **route-map** <map-name> parameter specifies a route map to be consulted before adding the RIP route to the BGP4 route table.

NOTE: The route map you specify must already be configured on the router. See "Defining Route Maps" on page 21-48 for information about defining route maps.

Redistributing RIP Routes

To configure BGP4 to redistribute RIP routes and add a metric of 10 to the redistributed routes, enter the following command:

```
FESX424 Router(config-bgp-router)# redistribute rip metric 10
```

Syntax: redistribute rip [metric <num>] [route-map <map-name>]

The **rip** parameter indicates that you are redistributing RIP routes into BGP4.

The **metric** <num> parameter changes the metric. You can specify a value from 0 – 4294967295. The default is 0.

The **route-map** <map-name> parameter specifies a route map to be consulted before adding the RIP route to the BGP4 route table.

NOTE: The route map you specify must already be configured on the router. See "Defining Route Maps" on page 21-48 for information about defining route maps.

Redistributing OSPF External Routes

To configure the Layer 3 Switch to redistribute OSPF external type 1 routes, enter the following command:

```
FESX424 Router(config-bgp-router)# redistribute ospf match external1
```

Syntax: redistribute ospf [match internal | external1 | external2] [metric <num>] [route-map <map-name>]

The **ospf** parameter indicates that you are redistributing OSPF routes into BGP4.

The **match internal | external1 | external2** parameter applies only to OSPF. This parameter specifies the types of OSPF routes to be redistributed into BGP4. The default is internal.

NOTE: If you do not enter a value for the **match** parameter, (for example, you enter r**edistribute ospf** only) then only internal OSPF routes will be redistributed.

The **metric** <num> parameter changes the metric. You can specify a value from 0 – 4294967295. The default is 0.

The **route-map** <map-name> parameter specifies a route map to be consulted before adding the OSPF route to the BGP4 route table.

NOTE: The route map you specify must already be configured on the router. See “Defining Route Maps” on page 21-48 for information about defining route maps.

NOTE: If you use both the **redistribute ospf route-map** <map-name> command and the **redistribute ospf match internal | external1 | external2** command, the software uses only the route map for filtering.

Redistributing Static Routes

To configure the Layer 3 Switch to redistribute static routes, enter the following command:

```
FESX424 Router(config-bgp-router)# redistribute static
```

Syntax: redistribute static [metric <num>] [route-map <map-name>]

The **static** parameter indicates that you are redistributing static routes into BGP4.

The **metric** <num> parameter changes the metric. You can specify a value from 0 – 4294967295. The default is 0.

The **route-map** <map-name> parameter specifies a route map to be consulted before adding the static route to the BGP4 route table.

NOTE: The route map you specify must already be configured on the router. See “Defining Route Maps” on page 21-48 for information about defining route maps.

Disabling or Re-Enabling Re-Advertisement of All Learned BGP4 Routes to All BGP4 Neighbors

By default, the Layer 3 Switch re-advertises all learned best BGP4 routes to BGP4 neighbors, unless the routes are discarded or blocked by route maps or other filters.

If you want to prevent the Layer 3 Switch from re-advertising a learned best BGP4 route unless that route also is installed in the IP route table, use the following CLI method.

To disable re-advertisement of BGP4 routes to BGP4 neighbors except for routes that the software also installs in the route table, enter the following command:

```
FESX424 Router(config-bgp-router)# no readvertise
```

Syntax: [no] readvertise

To re-enable re-advertisement, enter the following command:

```
FESX424 Router(config-bgp-router)# readvertise
```

Redistributing IBGP Routes into RIP and OSPF

By default, the Layer 3 Switch does not redistribute IBGP routes from BGP4 into RIP or OSPF. This behavior helps eliminate routing loops. However, if your network can benefit from redistributing the IBGP routes from BGP4 into OSPF or RIP, you can enable the Layer 3 Switch to redistribute the routes. To do so, use the following CLI method.

To enable the Layer 3 Switch to redistribute BGP4 routes into OSPF and RIP, enter the following command:

```
FESX424 Router(config-bgp-router)# bgp-redistribute-internal
```

Syntax: [no] bgp-redistribute-internal

To disable redistribution of IBGP routes into RIP and OSPF, enter the following command:

```
FESX424 Router(config-bgp-router)# no bgp-redistribute-internal
```

Filtering

This section describes the following:

Filtering Specific IP Addresses

You can configure the router to explicitly permit or deny specific IP addresses received in updates from BGP4 neighbors by defining IP address filters. The router permits all IP addresses by default. You can define up to 100 IP address filters for BGP4.

- If you want permit to remain the default behavior, define individual filters to deny specific IP addresses.
- If you want to change the default behavior to deny, define individual filters to permit specific IP addresses.

NOTE: Once you define a filter, the default action for addresses that do not match a filter is “deny”. To change the default action to “permit”, configure the last filter as “permit any any”.

Address filters can be referred to by a BGP neighbor's distribute list number as well as by match statements in a route map.

NOTE: If the filter is referred to by a route map’s match statement, the filter is applied in the order in which the filter is listed in the match statement.

NOTE: You also can filter on IP addresses by using IP ACLs.

To define an IP address filter to deny routes to 209.157.0.0, enter the following command:

```
FESX424 Router(config-bgp-router)# address-filter 1 deny 209.157.0.0 255.255.0.0
```

Syntax: address-filter <num> permit | deny <ip-addr> <wildcard> <mask> <wildcard>

The <num> parameter is the filter number.

The **permit | deny** parameter indicates the action the Layer 3 Switch takes if the filter match is true.

- If you specify **permit**, the Layer 3 Switch permits the route into the BGP4 table if the filter match is true.
- If you specify **deny**, the Layer 3 Switch denies the route from entering the BGP4 table if the filter match is true.

NOTE: Once you define a filter, the default action for addresses that do not match a filter is “deny”. To change the default action to “permit”, configure the last filter as “permit any any”.

The <ip-addr> parameter specifies the IP address. If you want the filter to match on all addresses, enter **any**.

The <wildcard> parameter specifies the portion of the IP address to match against. The <wildcard> is a four-part value in dotted-decimal notation (IP address format) consisting of ones and zeros. Zeros in the mask mean the packet’s source address must match the <source-ip>. Ones mean any value matches. For example, the <ip-addr> and <wildcard> values 209.157.22.26 0.0.0.255 mean that all hosts in the Class C sub-net 209.157.22.x match the policy.

If you prefer to specify the wildcard (mask value) in Classless Interdomain Routing (CIDR) format, you can enter a forward slash after the IP address, then enter the number of significant bits in the mask. For example, you can enter the CIDR equivalent of “209.157.22.26 0.0.0.255” as “209.157.22.26/24”. The CLI automatically converts the CIDR number into the appropriate mask (where zeros instead of ones are the significant bits) and changes the non-significant portion of the IP address into zeros. For example, if you specify 209.157.22.26/24 or 209.157.22.26 0.0.0.255, then save the changes to the startup-config file, the value appears as 209.157.22.0/24 (if you have enabled display of sub-net lengths) or 209.157.22.0 0.0.0.255 in the startup-config file.

If you enable the software to display IP sub-net masks in CIDR format, the mask is saved in the file in “/<mask-bits>” format. To enable the software to display the CIDR masks, enter the **ip show-subnet-length** command at the global CONFIG level of the CLI. You can use the CIDR format to configure the filter regardless of whether the software is configured to display the masks in CIDR format.

The <mask> parameter specifies the network mask. If you want the filter to match on all destination addresses, enter **any**. The wildcard works the same as described above.

Filtering AS-Paths

You can filter updates received from BGP4 neighbors based on the contents of the AS-path list accompanying the updates. For example, if you want to deny routes that have the AS 4.3.2.1 in the AS-path from entering the BGP4 route table, you can define a filter to deny such routes.

The Layer 3 Switch provides the following methods for filtering on AS-path information:

- AS-path filters
- AS-path ACLs

NOTE: The Layer 3 Switch cannot actively support AS-path filters and AS-path ACLs at the same time. Use one method or the other but do not mix methods.

NOTE: Once you define a filter or ACL, the default action for updates that do not match a filter is “deny”. To change the default action to “permit”, configure the last filter or ACL as “permit any any”.

AS-path filters or AS-path ACLs can be referred to by a BGP neighbor's filter list number as well as by match statements in a route map.

Defining an AS-Path Filter

To define AS-path filter 4 to permit AS 2500, enter the following command:

```
FESX424 Router(config-bgp-router)# as-path-filter 4 permit 2500
```

Syntax: as-path-filter <num> permit | deny <as-path>

The <num> parameter identifies the filter's position in the AS-path filter list and can be from 1 – 100. Thus, the AS-path filter list can contain up to 100 filters. The Foundry Layer 3 Switch applies the filters in numerical order, beginning with the lowest-numbered filter. When a filter match is true, the Layer 3 Switch stops and does not continue applying filters from the list.

NOTE: If the filter is referred to by a route map's match statement, the filter is applied in the order in which the filter is listed in the match statement.

The **permit | deny** parameter indicates the action the router takes if the filter match is true.

- If you specify **permit**, the router permits the route into the BGP4 table if the filter match is true.
- If you specify **deny**, the router denies the route from entering the BGP4 table if the filter match is true.

The <as-path> parameter indicates the AS-path information. You can enter an exact AS-path string if you want to filter for a specific value. You also can use regular expressions in the filter string.

Defining an AS-Path ACL

To configure an AS-path list that uses ACL 1, enter a command such as the following:

```
FESX424 Router(config)# ip as-path access-list 1 permit 100
FESX424 Router(config)# router bgp
FESX424 Router(config-bgp-router)# neighbor 10.10.10.1 filter-list 1 in
```

The **ip as-path** command configures an AS-path ACL that permits routes containing AS number 100 in their AS paths. The **neighbor** command then applies the AS-path ACL to advertisements and updates received from neighbor 10.10.10.1. In this example, the only routes the Layer 3 Switch permits from neighbor 10.10.10.1 are those whose AS-paths contain AS-path number 100.

Syntax: ip as-path access-list <string> [seq <seq-value>] deny | permit <regular-expression>

The <string> parameter specifies the ACL name. (If you enter a number, the CLI interprets the number as a text string.)

The **seq** <seq-value> parameter is optional and specifies the AS-path list's sequence number. You can configure up to 199 entries in an AS-path list. If you do not specify a sequence number, the software numbers them in increments of 5, beginning with number 5. The software interprets the entries in an AS-path list in numerical order, beginning with the lowest sequence number.

The **deny | permit** parameter specifies the action the software takes if a route's AS-path list matches a match statement in this ACL. To configure the AS-path match statements in a route map, use the **match as-path** command. See "Matching Based on AS-Path ACL" on page 21-51.

The <regular-expression> parameter specifies the AS path information you want to permit or deny to routes that match any of the match statements within the ACL. You can enter a specific AS number or use a regular expression. For the regular expression syntax, see "Using Regular Expressions" on page 21-42.

The **neighbor** command uses the **filter-list** parameter to apply the AS-path ACL to the neighbor. See "Adding BGP4 Neighbors" on page 21-12.

Using Regular Expressions

You use a regular expression for the <as-path> parameter to specify a single character or multiple characters as a filter pattern. If the AS-path matches the pattern specified in the regular expression, the filter evaluation is true; otherwise, the evaluation is false.

In addition, you can include special characters that influence the way the software matches the AS-path against the filter value.

To filter on a specific single-character value, enter the character for the <as-path> parameter. For example, to filter on AS-paths that contain the letter "z", enter the following command:


```
FESX424 Router(config-bgp-router)# as-path-filter 1 permit z
```

To filter on a string of multiple characters, enter the characters in brackets. For example, to filter on AS-paths that contain “x”, “y”, or “z”, enter the following command:

```
FESX424 Router(config-bgp-router)# as-path-filter 1 permit [xyz]
```

Special Characters

When you enter as single-character expression or a list of characters, you also can use the following special characters. Table 21.3 on page 21-43 lists the special characters. The description for each special character includes an example. Notice that you place some special characters in front of the characters they control but you place other special characters after the characters they control. In each case, the examples show where to place the special character.

Table 21.3: BGP4 Special Characters for Regular Expressions

Character	Operation
.	The period matches on any single character, including a blank space. For example, the following regular expression matches for “aa”, “ab”, “ac”, and so on, but not just “a”. a.
*	The asterisk matches on zero or more sequences of a pattern. For example, the following regular expression matches on an AS-path that contains the string “1111” followed by any value: 1111*
+	The plus sign matches on one or more sequences of a pattern. For example, the following regular expression matches on an AS-path that contains a sequence of “g”s, such as “deg”, “degg”, “deggg”, and so on: deg+
?	The question mark matches on zero occurrences or one occurrence of a pattern. For example, the following regular expression matches on an AS-path that contains “dg” or “deg”: de?g
^	A caret (when not used within brackets) matches on the beginning of an input string. For example, the following regular expression matches on an AS-path that begins with “3”: ^3
$	A dollar sign matches on the end of an input string. For example, the following regular expression matches on an AS-path that ends with “deg”: deg$

Table 21.3: BGP4 Special Characters for Regular Expressions (Continued)

Character	Operation
_	An underscore matches on one or more of the following: • **,** (comma) • **{** (left curly brace) • **}** (right curly brace) • **(** (left parenthesis) • **)** (right parenthesis) • The beginning of the input string • The end of the input string • A blank space For example, the following regular expression matches on “100” but not on “1002”, “2100”, and so on. _100_
[ ]	Square brackets enclose a range of single-character patterns. For example, the following regular expression matches on an AS-path that contains “1”, “2”, “3”, “4”, or “5”: [1-5] You can use the following expression symbols within the brackets. These symbols are allowed only inside the brackets. • **^ –** The caret matches on any characters *except* the ones in the brackets. For example, the following regular expression matches on an AS-path that does *not* contain “1”, “2”, “3”, “4”, or “5”: [^1-5] • **-** The hyphen separates the beginning and ending of a range of characters. A match occurs if any of the characters within the range is present. See the example above.
|	A vertical bar (sometimes called a pipe or a “logical or”) separates two alternative values or sets of values. The AS-path can match one or the other value. For example, the following regular expression matches on an AS-path that contains either “abc” or “defg”: (abc)|(defg) **Note**: The parentheses group multiple characters to be treated as one value. See the following row for more information about parentheses.
( )	Parentheses allow you to create complex expressions. For example, the following complex expression matches on “abc”, “abcabc”, or “abcabcabcdefg”, but not on “abcdefgdefg”: ((abc)+)|((defg)?)

If you want to filter for a special character instead of using the special character as described in Table 21.3 on page 21-43, enter “****” (backslash) in front of the character. For example, to filter on AS-path strings containing an asterisk, enter the asterisk portion of the regular expression as “*****”.

```
FESX424 Router(config-bgp-router)# as-path-filter 2 deny \*
```


To use the backslash as a string character, enter two slashes. For example, to filter on AS-path strings containing a backslash, enter the backslash portion of the regular expression as "****".

```
FESX424 Router(config-bgp-router)# as-path-filter 2 deny \\
```

Filtering Communities

You can filter routes received from BGP4 neighbors based on community names. Use either of the following methods to do so.

A community is an optional attribute that identifies the route as a member of a user-defined class of routes. Community names are arbitrary values made of two five-digit integers joined by a colon. You determine what the name means when you create the community name as one of a route's attributes. Each string in the community name can be a number from 0 – 65535.

This format allows you to easily classify community names. For example, a common convention used in community naming is to configure the first string as the local AS and the second string as the unique community within that AS. Using this convention, communities 1:10, 1:20, and 1:30 can be easily identified as member communities of AS 1.

The Layer 3 Switch provides the following methods for filtering on community information:

- Community filters
- Community list ACLs

NOTE: The Layer 3 Switch cannot actively support community filters and community list ACLs at the same time. Use one method or the other but do not mix methods.

NOTE: Once you define a filter or ACL, the default action for communities that do not match a filter or ACL is "deny". To change the default action to "permit", configure the last filter or ACL entry as "permit any any".

Community filters or ACLs can be referred to by match statements in a route map.

Defining a Community Filter

To define filter 3 to permit routes that have the NO_ADVERTISE community, enter the following command:

```
FESX424 Router(config-bgp-router)# community-filter 3 permit no-advertise
```

Syntax: community-filter <num> permit | deny <num>:<num> | internet | local-as | no-advertise | no-export

The <num> parameter identifies the filter's position in the community filter list and can be from 1 – 100. Thus, the community filter list can contain up to 100 filters. The router applies the filters in numerical order, beginning with the lowest-numbered filter. When a filter match is true, the router stops and does not continue applying filters from the list.

NOTE: If the filter is referred to by a route map's match statement, the filter is applied in the order in which the filter is listed in the match statement.

The **permit | deny** parameter indicates the action the router takes if the filter match is true.

- If you specify **permit**, the router permits the route into the BGP4 table if the filter match is true.
- If you specify **deny**, the router denies the route from entering the BGP4 table if the filter match is true.

The <num>:<num> parameter indicates a specific community number to filter. Use this parameter to filter for a private (administrator-defined) community. You can enter up to 20 community numbers with the same command.

If you want to filter for the well-known communities "LOCAL_AS", "NO_EXPORT" or "NO_ADVERTISE", use the corresponding keyword (described below).

The **internet** keyword checks for routes that do not have the community attribute. Routes without a specific community are considered by default to be members of the largest community, the Internet.

The **local-as** keyword checks for routes with the well-known community "LOCAL_AS". This community applies only to confederations. The Layer 3 Switch advertises the route only within the sub-AS. For information about confederations, see "Configuring Confederations" on page 21-34.

The **no-advertise** keyword filters for routes with the well-known community "NO_ADVERTISE". A route in this community should not be advertised to any BGP4 neighbors.

The **no-export** keyword filters for routes with the well-known community "NO_EXPORT". A route in this community should not be advertised to any BGP4 neighbors outside the local AS. If the router is a member of a confederation, the Layer 3 Switch advertises the route only within the confederation. For information about confederations, see "Configuring Confederations" on page 21-34.

Defining a Community ACL

To configure community ACL 1, enter a command such as the following:

```
FESX424 Router(config)# ip community-list 1 permit 123:2
```

This command configures a community ACL that permits routes that contain community 123:2.

NOTE: See "Matching Based on Community ACL" on page 21-51 for information about how to use a community list as a match condition in a route map.

Syntax: ip community-list standard <string> [seq <seq-value>] deny | permit <community-num>

Syntax: ip community-list extended <string> [seq <seq-value>] deny | permit <community-num> | <regular-expression>

The <string> parameter specifies the ACL name. (If you enter a number, the CLI interprets the number as a text string.)

The **standard** or **extended** parameter specifies whether you are configuring a standard community ACL or an extended one. A standard community ACL does not support regular expressions whereas an extended one does. This is the only difference between standard and extended IP community lists.

The **seq** <seq-value> parameter is optional and specifies the community list's sequence number. You can configure up to 199 entries in a community list. If you do not specify a sequence number, the software numbers them in increments of 5, beginning with number 5. The software interprets the entries in a community list in numerical order, beginning with the lowest sequence number.

The **deny** | **permit** parameter specifies the action the software takes if a route's community list matches a match statement in this ACL. To configure the community-list match statements in a route map, use the **match community** command. See "Matching Based on Community ACL" on page 21-51.

The <community-num> parameter specifies the community type or community number. This parameter can have the following values:

- <num>**:**<num> – A specific community number
- **internet** – The Internet community
- **no-export** – The community of sub-ASs within a confederation. Routes with this community can be exported to other sub-ASs within the same confederation but cannot be exported outside the confederation to other ASs or otherwise sent to EBGP neighbors.
- **local-as** – The local sub-AS within the confederation. Routes with this community can be advertised only within the local subAS.
- **no-advertise** – Routes with this community cannot be advertised to any other BGP4 routers at all.

The <regular-expression> parameter specifies a regular expression for matching on community names. For information about regular expression syntax, see "Using Regular Expressions" on page 21-42. You can specify a regular expression only in an extended community ACL.

Defining IP Prefix Lists

An IP prefix list specifies a list of networks. When you apply an IP prefix list to a neighbor, the Layer 3 Switch sends or receives only a route whose destination is in the IP prefix list. You can configure up to 100 prefix lists. The software interprets the prefix lists in order, beginning with the lowest sequence number.

To configure an IP prefix list and apply it to a neighbor, enter commands such as the following:

```
FESX424 Router(config)# ip prefix-list Routesfor20 permit 20.20.0.0/24
FESX424 Router(config)# router bgp
FESX424 Router(config-bgp-router)# neighbor 10.10.10.1 prefix-list Routesfor20 out
```

These commands configure an IP prefix list named Routesfor20, which permits routes to network 20.20.0.0/24. The **neighbor** command configures the Layer 3 Switch to use IP prefix list Routesfor20 to determine which routes to send to neighbor 10.10.10.1. The Layer 3 Switch sends routes that go to 20.20.x.x to neighbor 10.10.10.1 because the IP prefix list explicitly permits these routes to be sent to the neighbor.

Syntax: ip prefix-list <name> [seq <seq-value>] [description <string>] deny | permit <network-addr>/<mask-bits> [ge <ge-value>] [le <le-value>]

The <name> parameter specifies the prefix list name. You use this name when applying the prefix list to a neighbor.

The **description** <string> parameter is a text string describing the prefix list.

The **seq** <seq-value> parameter is optional and specifies the IP prefix list's sequence number. You can configure up to 100 prefix list entries. If you do not specify a sequence number, the software numbers them in increments of 5, beginning with prefix list entry 5. The software interprets the prefix list entries in numerical order, beginning with the lowest sequence number.

The **deny | permit** parameter specifies the action the software takes if a neighbor's route is in this prefix list.

The prefix-list matches only on this network unless you use the **ge** <ge-value> or **le** <le-value> parameters. (See below.)

The <network-addr>/<mask-bits> parameter specifies the network number and the number of bits in the network mask.

You can specify a range of prefix length for prefixes that are more specific than <network-addr>/<mask-bits>.

- If you specify only **ge** <ge-value>, then the mask-length range is from <ge-value> to 32.
- If you specify only **le** <le-value>, then the mask-length range is from length to <le-value>.

The <ge-value> or <le-value> you specify must meet the following condition:

length < ge-value <= le-value <= 32

If you do not specify **ge** <ge-value> or **le** <le-value>, the prefix list matches only on the exact network prefix you specify with the <network-addr>/<mask-bits> parameter.

For the syntax of the **neighbor** command shown in the example above, see "Adding BGP4 Neighbors" on page 21-12.

Defining Neighbor Distribute Lists

A neighbor distribute list is a list of BGP4 address filters or ACLs that filter the traffic to or from a neighbor. To configure a neighbor distribute list, use either of the following methods.

To configure a distribute list that uses ACL 1, enter a command such as the following:

```
FESX424 Router(config-bgp-router)# neighbor 10.10.10.1 distribute-list 1 in
```

This command configures the Layer 3 Switch to use ACL 1 to select the routes that the Layer 3 Switch will accept from neighbor 10.10.10.1.

Syntax: neighbor <ip-addr> distribute-list <name-or-num> in | out

The <ip-addr> parameter specifies the neighbor.

The <name-or-num> parameter specifies the name or number of a standard, extended, or named ACL.

The **in** | **out** parameter specifies whether the distribute list applies to inbound or outbound routes:

- **in** – controls the routes the Layer 3 Switch will accept from the neighbor.
- **out** – controls the routes sent to the neighbor.

NOTE: The command syntax shown above is new. However, the **neighbor** <ip-addr> **distribute-list in** | **out** <num> command (where the direction is specified before the filter number) is the same as in earlier software releases. Use the new syntax when you are using an IP ACL with the distribute list. Use the old syntax when you are using a BGP4 address filter with the distribute list.

Defining Route Maps

A ***route map*** is a named set of match conditions and parameter settings that the router can use to modify route attributes and to control redistribution of the routes into other protocols. A route map consists of a sequence of up to 50 ***instances***. If you think of a route map as a table, an instance is a row in that table. The router evaluates a route according to a route map's instances in ascending numerical order. The route is first compared against instance 1, then against instance 2, and so on. As soon as a match is found, the router stops evaluating the route against the route map instances.

Route maps can contain ***match*** statements and ***set*** statements. Each route map contains a "permit" or "deny" action for routes that match the match statements.

- If the route map contains a permit action, a route that matches a match statement is permitted; otherwise, the route is denied.
- If the route map contains a deny action, a route that matches a match statement is denied.
- If a route does not match any match statements in the route map, the route is denied. This is the default action. To change the default action, configure the last match statement in the last instance of the route map to "permit any any".
- If there is no match statement, the software considers the route to be a match.
- For route maps that contain address filters, AS-path filters, or community filters, if the action specified by a filter conflicts with the action specified by the route map, the route map's action takes precedence over the individual filter's action.

If the route map contains set statements, routes that are permitted by the route map's match statements are modified according to the set statements.

Match statements compare the route against one or more of the following:

- The route's BGP4 MED (metric)
- A sequence of AS-path filters
- A sequence of community filters
- A sequence of address filters
- The IP address of the next hop router
- The route's tag
- For OSPF routes only, the route's type (internal, external type-1, or external type-2)
- An AS-path ACL
- A community ACL
- An IP prefix list
- An IP ACL

For routes that match all of the match statements, the route map's set statements can perform one or more of the following modifications to the route's attributes:

- Prepend AS numbers to the front of the route's AS-path. By adding AS numbers to the AS-path, you can cause the route to be less preferred when compared to other routes on the basis of the length of the AS-path.
- Add a user-defined tag to the route or add an automatically calculated tag to the route.
- Set the community value.
- Set the local preference.
- Set the MED (metric).
- Set the IP address of the next hop router.
- Set the origin to IGP or INCOMPLETE.
- Set the weight.

For example, when you configure parameters for redistributing routes into RIP, one of the optional parameters is a route map. If you specify a route map as one of the redistribution parameters, the router will match the route against the match statements in the route map. If a match is found and if the route map contains set statements, the router will set attributes in the route according to the set statements.

To create a route map, you define instances of the map. Each instance is identified by a sequence number. A route map can contain up to 50 instances.

To define a route map, use the procedures in the following sections.

Entering the Route Map Into the Software

To add instance 1 of a route map named "GET_ONE" with a permit action, enter the following command.

```
FESX424 Router(config)# route-map GET_ONE permit 1
FESX424 Router(config-routemap GET_ONE)#
```

Syntax: [no] route-map <map-name> permit | deny <num>

As shown in this example, the command prompt changes to the Route Map level. You can enter the match and set statements at this level. See "Specifying the Match Conditions" on page 21-50 and "Setting Parameters in the Routes" on page 21-52.

The <map-name> is a string of characters that names the map. Map names can be up to 32 characters in length.

The **permit | deny** parameter specifies the action the router will take if a route matches a match statement.

- If you specify **deny**, the Layer 3 Switch does not advertise or learn the route.
- If you specify **permit**, the Layer 3 Switch applies the match and set statements associated with this route map instance.

The <num> parameter specifies the instance of the route map you are defining. Each route map can have up to 50 instances.

To delete a route map, enter a command such as the following. When you delete a route map, all the permit and deny entries in the route map are deleted.

```
FESX424 Router(config)# no route-map Map1
```

This command deletes a route map named "Map1". All entries in the route map are deleted.

To delete a specific instance of a route map without deleting the rest of the route map, enter a command such as the following:

```
FESX424 Router(config)# no route-map Map1 permit 10
```

This command deletes the specified instance from the route map but leaves the other instances of the route map intact.

Specifying the Match Conditions

Use the following command to define the match conditions for instance 1 of the route map GET_ONE. This instance compares the route updates against BGP4 address filter 11.

```
FESX424 Router(config-routemap GET_ONE)# match address-filters 11
```

Syntax: match
[as-path <num>] |
[address-filters | as-path-filters | community-filters <num,num,...>] |
[community <num>] |
[community <acl> exact-match] |
[ip address <acl> | prefix-list <string>] |
[ip route-source <acl> | prefix <name>]
[metric <num>] |
[next-hop <address-filter-list>] |
[nlri multicast | unicast | multicast unicast] |
[route-type internal | external-type1 | external-type2] |
[tag <tag-value>]

The **as-path** <num> parameter specifies an AS-path ACL. You can specify up to five AS-path ACLs. To configure an AS-path ACL, use the **ip as-path access-list** command. See “Defining an AS-Path ACL” on page 21-42.

The **address-filters | as-path-filters | community-filters** <num,num,...> parameter specifies a filter or list of filters to be matched for each route. The router treats the first match as the best match. If a route does not match any filter in the list, then the router considers the match condition to have failed. To configure these types of filters, use commands at the BGP configuration level.

- To configure an address filter, see “Filtering Specific IP Addresses” on page 21-40.
- To configure an AS-path filter or AS-path ACL, see “Filtering AS-Paths” on page 21-41.
- To configure a community filter or community ACL, see “Filtering Communities” on page 21-45.

You can enter up to six community names on the same command line.

NOTE: The filters must already be configured.

The **community** <num> parameter specifies a community ACL.

NOTE: The ACL must already be configured.

The **community** <acl> **exact-match** parameter matches a route if (and only if) the route's community attributes field contains the same community numbers specified in the match statement.

The **ip address | next-hop** <acl-num> | prefix-list <string> parameter specifies an ACL or IP prefix list. Use this parameter to match based on the destination network or next-hop gateway. To configure an IP ACL for use with this command, use the **ip access-list** command. See “Rule-Based IP Access Control Lists (ACLs)” on page 12-1. To configure an IP prefix list, use the **ip prefix-list** command. See “Defining IP Prefix Lists” on page 21-47.

The **ip route-source** <acl> | **prefix** <name> parameter matches based on the source of a route (the IP address of the neighbor from which the Foundry device learned the route).

The **metric** <num> parameter compares the route’s MED (metric) to the specified value.

The **next-hop** <address-filter-list> parameter compares the IP address of the route’s next hop to the specified IP address filters. The filters must already be configured.

The **nlri multicast | unicast | multicast unicast** parameter specifies whether you want the route map to match on multicast routes, unicast routes, or both route types.

NOTE: By default, route maps apply to both unicast and multicast traffic.

The **route-type internal | external-type1 | external-type2** parameter applies only to OSPF routes. This parameter compares the route's type to the specified value.

The **tag** <tag-value> parameter compares the route's tag to the specified value.

Match Examples Using ACLs

The following sections show some detailed examples of how to configure route maps that include match statements that match on ACLs.

Matching Based on AS-Path ACL

To construct a route map that matches based on AS-path ACL 1, enter the following commands:

```
FESX424 Router(config)# route-map PathMap permit 1
FESX424 Router(config-routemap PathMap)# match as-path 1
```

Syntax: match as-path <num>

The <num> parameter specifies an AS-path ACL and can be a number from 1 – 199. You can specify up to five AS-path ACLs. To configure an AS-path ACL, use the **ip as-path access-list** command. See "Defining an AS-Path ACL" on page 21-42.

Matching Based on Community ACL

To construct a route map that matches based on community ACL 1, enter the following commands:

```
FESX424 Router(config)# ip community-list 1 permit 123:2
FESX424 Router(config)# route-map CommMap permit 1
FESX424 Router(config-routemap CommMap)# match community 1
```

Syntax: match community <string>

The <string> parameter specifies a community list ACL. To configure a community list ACL, use the **ip community-list** command. See "Defining a Community ACL" on page 21-46.

Matching Based on Destination Network

To construct match statements for a route map that match based on destination network, use the following method. You can use the results of an IP ACL or an IP prefix list as the match condition.

```
FESX424 Router(config)# route-map NetMap permit 1
FESX424 Router(config-routemap NetMap)# match ip address 1
```

Syntax: match ip address <name-or-num>

Syntax: match ip address prefix-list <name>

The <name-or-num> parameter with the first command specifies an IP ACL and can be a number from 1 – 199 or the ACL name if it is a named ACL. To configure an IP ACL, use the **ip access-list** or **access-list** command. See "Rule-Based IP Access Control Lists (ACLs)" on page 12-1.

The <name> parameter with the second command specifies an IP prefix list name. To configure an IP prefix list, see "Defining IP Prefix Lists" on page 21-47.

Matching Based on Next-Hop Router

To construct match statements for a route map that match based on the IP address of the next-hop router, use either of the following methods. You can use the results of an IP ACL or an IP prefix list as the match condition.

To construct a route map that matches based on the next-hop router, enter commands such as the following:

```
FESX424 Router(config)# route-map HopMap permit 1
FESX424 Router(config-routemap HopMap)# match ip next-hop 2
```

Syntax: match ip next-hop <num>

Syntax: match ip next-hop prefix-list <name>

The <num> parameter with the first command specifies an IP ACL and can be a number from 1 – 199 or the ACL name if it is a named ACL. To configure an IP ACL, use the **ip access-list** or **access-list** command. See "Rule-Based IP Access Control Lists (ACLs)" on page 12-1.

The <name> parameter with the second command specifies an IP prefix list name. To configure an IP prefix list, see “Defining IP Prefix Lists” on page 21-47.

Matching Based on the Route Source

To match a BGP4 route based on its source, use the **match ip route-source** statement. Here is an example:

```
FESX424 Router(config)# access-list 10 permit 192.168.6.0 0.0.0.255
FESX424 Router(config)# route-map bgp1 permit 1
FESX424 Router(config-routemap bgp1)# match ip route-source 10
```

The first command configures an IP ACL that matches on routes received from 192.168.6.0/24. The remaining commands configure a route map that matches on all BGP4 routes advertised by the BGP4 neighbors whose addresses match addresses in the IP prefix list. You can add a set statement to change a route attribute in the routes that match. You also can use the route map as input for other commands, such as the **neighbor** and **network** commands and some show commands.

Syntax: match ip route-source <acl> | prefix <name>

The <acl> | prefix <name> parameter specifies the name or ID of an IP ACL, or an IP prefix list.

Matching On Routes Containing a Specific Set of Communities

Foundry software enables you to match routes based on the presence of a community name or number in a route, 0and to match when a route contains exactly the set of communities you specify. To match based on a set of communities, configure a community ACL that lists the communities, then compare routes against the ACL.

Here is an example.

```
FESX424 Router(config)# ip community-list standard std_1 permit 12:34 no-export
FESX424 Router(config)# route-map bgp2 permit 1
FESX424 Router(config-routemap bgp2)# match community std_1 exact-match
```

The first command configures a community ACL that contains community number 12:34 and community name no-export. The remaining commands configure a route map that matches the community attributes field in BGP4 routes against the set of communities in the ACL. A route matches the route map only if the route contains all the communities in the ACL and no other communities.

Syntax: match community <acl> exact-match

The <acl> parameter specifies the name of a community list ACL. You can specify up to five ACLs. Separate the ACL names or IDs with spaces.

Here is another example.

```
FESX424 Router(config)# ip community-list standard std_2 permit 23:45 56:78
FESX424 Router(config)# route-map bgp3 permit 1
FESX424 Router(config-routemap bgp3)# match community std_1 std_2 exact-match
```

These commands configure an additional community ACL, std_2, that contains community numbers 23:45 and 57:68. Route map bgp3 compares each BGP4 route against the sets of communities in ACLs std_1 and std_2. A BGP4 route that contains ***either but not both*** sets of communities matches the route map. For example, a route containing communities 23:45 and 57:68 matches. However, a route containing communities 23:45, 57:68 and 12:34, or communities 23:45, 57:68, 12:34, and no-export does not match. To match, the route’s communities must be the same as those in exactly one of the community ACLs used by the match community statement.

Setting Parameters in the Routes

Use the following command to define a set statement that prepends an AS number to the AS path on each route that matches the corresponding match statement.

```
FESX424 Router(config-routemap GET_ONE)# set as-path prepend 65535
```

Syntax: set
[as-path [prepend <as-num,as-num,...>]] |
[automatic-tag] |

[comm-list <acl> delete] |
[community <num>:<num> | <num> | internet | local-as | no-advertise | no-export] |
[dampening [<half-life> <reuse> <suppress> <max-suppress-time>]]
[[default] interface null0 |
[ip [default] next hop <ip-addr>]
[ip next-hop peer-address] |
[local-preference <num>] |
[metric [+ | -]<num> | none] |
[metric-type type-1 | type-2] |
[metric-type internal] |
[next-hop <ip-addr>] |
[nlri multicast | unicast | multicast unicast] |
[origin igp | incomplete] |
[tag <tag-value>] |
[weight <num>]

The **as-path prepend** <num,num,...> parameter adds the specified AS numbers to the front of the AS-path list for the route.

The **automatic-tag** parameter calculates and sets an automatic tag value for the route.

NOTE: This parameter applies only to routes redistributed into OSPF.

The **comm-list** parameter deletes a community from a BGP4 route's community attributes field.

The **community** parameter sets the community attribute for the route to the number or well-known type you specify.

The **dampening** [<half-life> <reuse> <suppress> <max-suppress-time>] parameter sets route dampening parameters for the route. The <half-life> parameter specifies the number of minutes after which the route's penalty becomes half its value. The <reuse> parameter specifies how low a route's penalty must become before the route becomes eligible for use again after being suppressed. The <suppress> parameter specifies how high a route's penalty can become before the Layer 3 Switch suppresses the route. The <max-suppress-time> parameter specifies the maximum number of minutes that a route can be suppressed regardless of how unstable it is. For information and examples, see "Configuring Route Flap Dampening" on page 21-58.

The **[default] interface null0** parameter redirects the traffic to the specified interface. You can send the traffic to the null0 interface, which is the same as dropping the traffic. You can specify more than one interface, in which case the Layer 3 Switch uses the first available port. If the first port is unavailable, the Layer 3 Switch sends the traffic to the next port in the list. If you specify **default**, the route map redirects the traffic to the specified interface only if the Layer 3 Switch does not already have explicit routing information for the traffic. This option is used in Policy-Based Routing (PBR).

The **ip [default] next hop** <ip-addr> parameter sets the next-hop IP address for traffic that matches a match statement in the route map. If you specify **default**, the route map sets the next-hop gateway only if the Layer 3 Switch does not already have explicit routing information for the traffic. This option is used in Policy-Based Routing (PBR).

The **ip next-hop peer-address** parameter sets the BGP4 next hop for a route to the specified neighbor address.

The **local-preference** <num> parameter sets the local preference for the route. You can set the preference to a value from 0 – 4294967295.

The **metric** [**+** | **-**]<num> | none parameter sets the MED (metric) value for the route. The default MED value is 0. You can set the preference to a value from 0 – 4294967295.

- **set metric** <num> – Sets the route's metric to the number you specify.
- **set metric +**<num> – Increases route's metric by the number you specify.
- **set metric -**<num> – Decreases route's metric by the number you specify.
- **set metric none** – Removes the metric from the route (removes the MED attribute from the BGP4 route).

The **metric-type type-1 | type-2** parameter changes the metric type of a route redistributed into OSPF.

The **metric-type internal** parameter sets the route's MED to the same value as the IGP metric of the BGP4 next-hop route. The parameter does this when advertising a BGP4 route to an EBGP neighbor.

The **next-hop** <ip-addr> parameter sets the IP address of the route's next hop router.

The **nlri multicast | unicast | multicast unicast** parameter redistributes routes into the multicast Routing Information Base (RIB) instead of the unicast RIB.

NOTE: Setting the NLRI type to multicast applies only when you are using the route map to redistribute directly-connected routes. Otherwise, the set option is ignored.

The **origin igp | incomplete** parameter sets the route's origin to IGP or INCOMPLETE.

The **tag** <tag-value> parameter sets the route's tag. You can specify a tag value from 0 – 4294967295.

NOTE: This parameter applies only to routes redistributed into OSPF.

NOTE: You also can set the tag value using a table map. The table map changes the value only when the Layer 3 Switch places the route in the IP route table instead of changing the value in the BGP route table. See "Using a Table Map To Set the Tag Value" on page 21-55.

The **weight** <num> parameter sets the weight for the route. You can specify a weight value from 0 – 4294967295.

Setting a BP4 Route's MED to the same Value as the IGP Metric of the Next-Hop Route

To set a route's MED to the same value as the IGP metric of the BGP4 next-hop route, when advertising the route to a neighbor, enter commands such as the following:

```
FESX424 Router(config)# access-list 1 permit 192.168.9.0 0.0.0.255
FESX424 Router(config)# route-map bgp4 permit 1
FESX424 Router(config-routemap bgp4)# match ip address 1
FESX424 Router(config-routemap bgp4)# set metric-type internal
```

The first command configures an ACL that matches on routes with destination network 192.168.9.0. The remaining commands configure a route map that matches on the destination network in ACL 1, then sets the metric type for those routes to the same value as the IGP metric of the BGP4 next-hop route.

Syntax: set metric-type internal

Setting the Next Hop of a BGP4 Route

To set the next hop address of a BGP4 route to a neighbor address, enter commands such as the following:

```
FESX424 Router(config)# route-map bgp5 permit 1
FESX424 Router(config-routemap bgp5)# match ip address 1
FESX424 Router(config-routemap bgp5)# set ip next-hop peer-address
```

These commands configure a route map that matches on routes whose destination network is specified in ACL 1, and sets the next hop in the routes to the neighbor address (inbound filtering) or the local IP address of the BGP4 session (outbound filtering).

Syntax: set ip next-hop peer-address

The value that the software substitutes for **peer-address** depends on whether the route map is used for inbound filtering or outbound filtering:

- When you use the **set ip next-hop peer-address** command in an inbound route map filter, **peer-address** substitutes for the neighbor's IP address.
- When you use the **set ip next-hop peer-address** command in an outbound route map filter, **peer-address** substitutes for the local IP address of the BGP4 session.

NOTE: You can use this command for a peer group configuration.

Deleting a Community from a BGP4 Route

To delete a community from a BGP4 route's community attributes field, enter commands such as the following:

```
FESX424 Router(config)# ip community-list standard std_3 permit 12:99 12:86
FESX424 Router(config)# route-map bgp6 permit 1
FESX424 Router(config-routemap bgp6)# match ip address 1
FESX424 Router(config-routemap bgp6)# set comm-list std_3 delete
```

The first command configures a community ACL containing community numbers 12:99 and 12:86. The remaining commands configure a route map that matches on routes whose destination network is specified in ACL 1, and deletes communities 12:99 and 12:86 from those routes. The route does not need to contain all the specified communities in order for them to be deleted. For example, if a route contains communities 12:86, 33:44, and 66:77, community 12:86 is deleted.

Syntax: set comm-list <acl> delete

The <acl> parameter specifies the name of a community list ACL.

Using a Table Map To Set the Tag Value

Route maps that contain set statements change values in routes when the routes are accepted by the route map. For inbound route maps (route maps that filter routes received from neighbors), this means that the routes are changed before they enter the BGP4 route table.

For tag values, if you do not want the value to change until a route enters the IP route table, you can use a table map to change the value. A table map is a route map that you have associated with the IP routing table. The Layer 3 Switch applies the set statements for tag values in the table map to routes before adding them to the route table.

To configure a table map, you configure the route map, then identify it as a table map. The table map does not require separate configuration. You create it simply by calling an existing route map a table map. You can have one table map.

NOTE: Use table maps only for setting the tag value. Do not use table maps to set other attributes. To set other route attributes, use route maps or filters.

To create a route map and identify it as a table map, enter commands such as following. These commands create a route map that uses an address filter. For routes that match the address filter, the route map changes the tag value to 100. This route map is then identified as a table map. As a result, the route map is applied only to routes that the Layer 3 Switch places in the IP route table. The route map is not applied to all routes. This example assumes that address filter 11 has already been configured.

```
FESX424 Router(config)# route-map TAG_IP permit 1
FESX424 Router(config-routemap TAG_IP)# match address-filters 11
FESX424 Router(config-routemap TAG_IP)# set tag 100
FESX424 Router(config-routemap TAG_IP)# router bgp
FESX424 Router(config-bgp-router)# table-map TAG_IP
```

Configuring Cooperative BGP4 Route Filtering

By default, the Layer 3 Switch performs all filtering of incoming routes locally, on the Layer 3 Switch itself. You can use cooperative BGP4 route filtering to cause the filtering to be performed by a neighbor before it sends the routes to the Layer 3 Switch. Cooperative filtering conserves resources by eliminating unnecessary route updates and filter processing. For example, the Layer 3 Switch can send a deny filter to its neighbor, which the neighbor uses to filter out updates before sending them to the Layer 3 Switch. The neighbor saves the resources it would otherwise use to generate the route updates, and the Layer 3 Switch saves the resources it would use to filter out the routes.

When you enable cooperative filtering, the Layer 3 Switch advertises this capability in its Open message to the neighbor when initiating the neighbor session. The Open message also indicates whether the Layer 3 Switch is

configured to send filters, receive filters or both, and the types of filters it can send or receive. The Layer 3 Switch sends the filters as Outbound Route Filters (ORFs) in Route Refresh messages.

To configure cooperative filtering, perform the following tasks on the Layer 3 Switch and on its BGP4 neighbor:

- Configure the filter.

  NOTE: The current release supports cooperative filtering only for filters configured using IP prefix lists.

- Apply the filter as in ***inbound*** filter to the neighbor.
- Enable the cooperative route filtering feature on the Layer 3 Switch. You can enable the Layer 3 Switch to send ORFs to the neighbor, to receive ORFs from the neighbor, or both. The neighbor uses the ORFs you send as outbound filters when it sends routes to the Layer 3 Switch. Likewise, the Layer 3 Switch uses the ORFs it receives from the neighbor as outbound filters when sending routes to the neighbor.
- Reset the BGP4 neighbor session to send and receive ORFs.
- Perform these steps on the other device.

NOTE: If the Layer 3 Switch has inbound filters, the filters are still processed even if equivalent filters have been sent as ORFs to the neighbor.

Enabling Cooperative Filtering

To configure cooperative filtering, enter commands such as the following:

```
FESX424 Router(config)# ip prefix-list Routesfrom1234 deny 20.20.0.0/24
FESX424 Router(config)# ip prefix-list Routesfrom1234 permit 0.0.0.0/0 le 32
FESX424 Router(config)# router bgp
FESX424 Router(config-bgp-router)# neighbor 1.2.3.4 prefix-list Routesfrom1234 in
FESX424 Router(config-bgp-router)# neighbor 1.2.3.4 capability orf prefixlist send
```

The first two commands configure statements for the IP prefix list Routesfrom1234. The first command configures a statement that denies routes to 20.20.20./24. The second command configures a statement that permits all other routes. (Once you configure an IP prefix list statement, all routes not explicitly permitted by statements in the prefix list are denied.)

The next two commands change the CLI to the BGP4 configuration level, then apply the IP prefix list to neighbor 1.2.3.4. The last command enables the Layer 3 Switch to send the IP prefix list as an ORF to neighbor 1.2.3.4. When the Layer 3 Switch sends the IP prefix list to the neighbor, the neighbor filters out the 20.20.0.x routes from its updates to the Layer 3 Switch. (This assumes that the neighbor also is configured for cooperative filtering.)

Syntax: [no] neighbor <ip-addr> | <peer-group-name> capability orf prefixlist [send | receive]

The <ip-addr> | <peer-group-name> parameter specifies the IP address of a neighbor or the name of a peer group of neighbors.

The **send** | **receive** parameter specifies the support you are enabling:

- **send** – The Layer 3 Switch sends the IP prefix lists to the neighbor.
- **receive** – The Layer 3 Switch accepts filters from the neighbor.

If you do not specify the capability, both capabilities are enabled.

The **prefixlist** parameter specifies the type of filter you want to send to the neighbor.

NOTE: The current release supports cooperative filtering only for filters configured using IP prefix lists.

Sending and Receiving ORFs

Cooperative filtering affects neighbor sessions that start after the filtering is enabled, but do not affect sessions that are already established.

To activate cooperative filtering, reset the session with the neighbor. This is required because the cooperative filtering information is exchanged in Open messages during the start of a session.

To place a prefix-list change into effect after activating cooperative filtering, perform a soft reset of the neighbor session. A soft reset does not end the current session, but sends the prefix list to the neighbor in the next route refresh message.

NOTE: Make sure cooperative filtering is enabled on the Layer 3 Switch and on the neighbor before you send the filters.

To reset a neighbor session and send ORFs to the neighbor, enter a command such as the following:

```
FESX424 Router# clear ip bgp neighbor 1.2.3.4
```

This command resets the BGP4 session with neighbor 1.2.3.4 and sends the ORFs to the neighbor. If the neighbor sends ORFs to the Layer 3 Switch, the Layer 3 Switch accepts them if the send capability is enabled.

To perform a soft reset of a neighbor session and send ORFs to the neighbor, enter a command such as the following:

```
FESX424 Router# clear ip bgp neighbor 1.2.3.4 soft in prefix-list
```

Syntax: clear ip bgp neighbor <ip-addr> [soft in prefix-filter]

If you use the **soft in prefix-filter** parameter, the Layer 3 Switch sends the updated IP prefix list to the neighbor as part of its route refresh message to the neighbor.

NOTE: If the Layer 3 Switch or the neighbor is not configured for cooperative filtering, the command sends a normal route refresh message.

Displaying Cooperative Filtering Information

You can display the following cooperative filtering information:

- The cooperative filtering configuration on the Layer 3 Switch.
- The ORFs received from neighbors.

To display the cooperative filtering configuration on the Layer 3 Switch, enter a command such as the following. The line shown in bold type shows the cooperative filtering status.

```
FESX424 Router# show ip bgp neighbor 10.10.10.1
1   IP Address: 10.10.10.1, AS: 65200 (IBGP), RouterID: 10.10.10.1
    State: ESTABLISHED, Time: 0h0m7s, KeepAliveTime: 60, HoldTime: 180
       RefreshCapability: Received
       CooperativeFilteringCapability: Received
    Messages:    Open    Update  KeepAlive Notification Refresh-Req
       Sent    : 1       0       1         0            1
       Received: 1       0       1         0            1
    Last Update Time: NLRI       Withdraw          NLRI       Withdraw
                  Tx: ---        ---           Rx: ---        ---
    Last Connection Reset Reason:Unknown
    Notification Sent:     Unspecified
    Notification Received: Unspecified
    TCP Connection state: ESTABLISHED
       Byte Sent:   110, Received: 110
       Local host:  10.10.10.2, Local  Port: 8138
       Remote host: 10.10.10.1, Remote Port: 179
       ISentSeq:        460  SendNext:        571  TotUnAck:          0
       TotSent:         111  ReTrans:           0  UnAckSeq:        571
       IRcvSeq:        7349  RcvNext:        7460  SendWnd:       16384
       TotalRcv:        111  DupliRcv:          0  RcvWnd:        16384
       SendQue:           0  RcvQue:            0  CngstWnd:       5325
```

Syntax: show ip bgp neighbor <ip-addr>

To display the ORFs received from a neighbor, enter a command such as the following:

```
FESX424 Router# show ip bgp neighbor 10.10.10.1 received prefix-filter
ip prefix-list 10.10.10.1: 4 entries
     seq 5 permit 10.10.0.0/16 ge 18 le 28
     seq 10 permit 20.20.10.0/24
     seq 15 permit 30.0.0.0/8 le 32
     seq 20 permit 40.10.0.0/16 ge 18
```

Syntax: show ip bgp neighbor <ip-addr> received prefix-filter

Configuring Route Flap Dampening

A “route flap” is the change in a route’s state, from up to down or down to up. When a route’s state changes, the state change causes changes in the route tables of the routers that support the route. Frequent changes in a route’s state can cause Internet instability and add processing overhead to the routers that support the route.

Route flap dampening is a mechanism that reduces the impact of route flap by changing a BGP4 router’s response to route state changes. When route flap dampening is configured, the Layer 3 Switch suppresses unstable routes until the route’s state changes reduce enough to meet an acceptable degree of stability. The Foundry implementation of route flap dampening is based on RFC 2439.

Route flap dampening is disabled by default. You can enable the feature globally or on an individual route basis using route maps.

NOTE: The Layer 3 Switch applies route flap dampening only to routes learned from EBGP neighbors.

The route flap dampening mechanism is based on penalties. When a route exceeds a configured penalty value, the Layer 3 Switch stops using that route and also stops advertising it to other routers. The mechanism also

allows a route's penalties to reduce over time if the route's stability improves. The route flap dampening mechanism uses the following parameters:

- Suppression threshold – Specifies the penalty value at which the Layer 3 Switch stops using the route. Each time a route becomes unreachable or is withdrawn by a BGP4 UPDATE from a neighbor, the route receives a penalty of 1000. By default, when a route has a penalty value greater than 2000, the Layer 3 Switch stops using the route. Thus, by default, if a route goes down more than twice, the Layer 3 Switch stops using the route. You can set the suppression threshold to a value from 1 – 20000. The default is 2000.
- Half-life – Once a route has been assigned a penalty, the penalty decreases exponentially and decreases by half after the half-life period. The default half-life period is 15 minutes. The software reduces route penalties every five seconds. For example, if a route has a penalty of 2000 and does not receive any more penalties (it does not go down again) during the half-life, the penalty is reduced to 1000 after the half-life expires. You can configure the half-life to be from 1 – 45 minutes. The default is 15 minutes.
- Reuse threshold – Specifies the minimum penalty a route can have and still be suppressed by the Layer 3 Switch. If the route's penalty falls below this value, the Layer 3 Switch un-suppresses the route and can use it again. The software evaluates the dampened routes every ten seconds and un-suppresses the routes that have penalties below the reuse threshold. You can set the reuse threshold to a value from 1 – 20000. The default is 750.
- Maximum suppression time – Specifies the maximum number of minutes a route can be suppressed regardless of how unstable the route has been before this time. You can set the parameter to a value from 1 – 20000 minutes. The default is four times the half-life. When the half-life value is set to its default (15 minutes), the maximum suppression time defaults to 60 minutes.

You can configure route flap dampening globally or for individual routes using route maps. If you configure route flap dampening parameters globally and also use route maps, the settings in the route maps override the global values.

Globally Configuring Route Flap Dampening

To enable route flap dampening using the default values, enter the following command:

```
FESX424 Router(config-bgp-router)# dampening
```

Syntax: dampening [<half-life> <reuse> <suppress> <max-suppress-time>]

The <half-life> parameter specifies the number of minutes after which the route's penalty becomes half its value. The route penalty allows routes that have remained stable for a while despite earlier instability to eventually become eligible for use again. The decay rate of the penalty is proportional to the value of the penalty. After the half-life expires, the penalty decays to half its value. Thus, a dampened route that is no longer unstable can eventually become eligible for use again. You can configure the half-life to be from 1 - 45 minutes. The default is 15 minutes.

The <reuse> parameter specifies how low a route's penalty must become before the route becomes eligible for use again after being suppressed. You can set the reuse threshold to a value from 1 – 20000. The default is 750 (0.75, or three-fourths, of the penalty assessed for a one "flap").

The <suppress> parameter specifies how high a route's penalty can become before the Layer 3 Switch suppresses the route. You can set the suppression threshold to a value from 1 – 20000. The default is 2000 (two "flaps").

The <max-suppress-time> parameter specifies the maximum number of minutes that a route can be suppressed regardless of how unstable it is. You can set the maximum suppression time to a value from 1 – 20000 minutes. The default is four times the half-life setting. Thus, if you use the default half-life of 15 minutes, the maximum suppression time is 60 minutes.

The following example shows how to change the dampening parameters.

```
FESX424 Router(config-bgp-router)# dampening 20 200 2500 40
```

This command changes the half-life to 20 minutes, the reuse threshold to 200, the suppression threshold to 2500, and the maximum number of minutes a route can be dampened to 40.

NOTE: To change any of the parameters, you must specify all the parameters with the command. If you want to leave some parameters unchanged, enter their default values.

Using a Route Map To Configure Route Flap Dampening for Specific Routes

Route maps enable you to fine tune route flap dampening parameters for individual routes. To configure route flap dampening parameters using route maps, configure BGP4 address filters for each route you want to set the dampening parameters for, then configure route map entries that set the dampening parameters for those routes. The following sections show examples.

To configure address filters and a route map for dampening specific routes, enter commands such as the following:

```
FESX424 Router(config)# router bgp
FESX424 Router(config-bgp-router)# address-filter 9 permit 209.157.22.0
255.255.255.0 255.255.255.0 255.255.255.0
FESX424 Router(config-bgp-router)# address-filter 10 permit 209.157.23.0
255.255.255.0 255.255.255.0 255.255.255.0
FESX424 Router(config-bgp-router)# exit
FESX424 Router(config)# route-map DAMPENING_MAP permit 9
FESX424 Router(config-routemap DAMPENING_MAP)# match address-filters 9
FESX424 Router(config-routemap DAMPENING_MAP)# set dampening 10 200 2500 40
FESX424 Router(config-routemap DAMPENING_MAP)# exit
FESX424 Router(config)# route-map DAMPENING_MAP permit 10
FESX424 Router(config-routemap DAMPENING_MAP)# match address-filters 10
FESX424 Router(config-routemap DAMPENING_MAP)# set dampening 20 200 2500 60
FESX424 Router(config-routemap DAMPENING_MAP)# router bgp
FESX424 Router(config-bgp-router)# dampening route-map DAMPENING_MAP
```

The **address-filter** commands in this example configure two BGP4 address filters, for networks 209.157.22.0 and 209.157.23.0. The first route-map command creates an entry in a route map called “DAMPENING_MAP”. Within this entry of the route map, the **match** command matches based on address filter 9, and the **set** command sets the dampening parameters for the route that matches. Thus, for BGP4 routes to 209.157.22.0, the Layer 3 Switch uses the route map to set the dampening parameters. These parameters override the globally configured dampening parameters.

The commands for the second entry in the route map (instance 10 in this example) perform the same functions for route 209.157.23.0. Notice that the dampening parameters are different for each route.

Using a Route Map To Configure Route Flap Dampening for a Specific Neighbor

You can use a route map to configure route flap dampening for a specific neighbor by performing the following tasks:

- Configure an empty route map with no match or set statements. This route map does not specify particular routes for dampening but does allow you to enable dampening globally when you refer to this route map from within the BGP configuration level.
- Configure another route map that explicitly enables dampening. Use a set statement within the route map to enable dampening. When you associate this route map with a specific neighbor, the route map enables dampening for all routes associated with the neighbor. You also can use match statements within the route map to selectively perform dampening on some routes from the neighbor.

  NOTE: You still need to configure the first route map to enable dampening globally. The second route map does not enable dampening by itself; it just applies dampening to a neighbor.

- Apply the route map to the neighbor.

To enable route flap dampening for a specific BGP4 neighbor, enter commands such as the following:

```
FESX424 Router(config)# route-map DAMPENING_MAP_ENABLE permit 1
FESX424 Router(config-routemap DAMPENING_MAP_ENABLE)# exit
FESX424 Router(config)# route-map DAMPENING_MAP_NEIGHBOR_A permit 1
FESX424 Router(config-routemap DAMPENING_MAP_NEIGHBOR_A)# set dampening
FESX424 Router(config-routemap DAMPENING_MAP_NEIGHBOR_A)# exit
FESX424 Router(config)# router bgp
FESX424 Router(config-bgp-router)# dampening route-map DAMPENING_MAP_ENABLE
FESX424 Router(config-bgp-router)# neighbor 10.10.10.1 route-map in
DAMPENING_MAP_NEIGHBOR_A
```

In this example, the first command globally enables route flap dampening. This route map does not contain any match or set statements. At the BGP configuration level, the **dampening route-map** command refers to the DAMPENING_MAP_ENABLE route map created by the first command, thus enabling dampening globally.

The third and fourth commands configure a second route map that explicitly enables dampening. Notice that the route map does not contain a match statement. The route map implicitly applies to all routes. Since the route map will be applied to a neighbor at the BGP configuration level, the route map will apply to all routes associated with the neighbor.

Although the second route map enables dampening, the first route map is still required. The second route map enables dampening for the neighbors to which the route map is applied. However, unless dampening is already enabled globally by the first route map, the second route map has no effect.

The last two commands apply the route maps. The **dampening route-map** command applies the first route map, which enables dampening globally. The **neighbor** command applies the second route map to neighbor 10.10.10.1. Since the second route map does not contain match statements for specific routes, the route map enables dampening for all routes received from the neighbor.

Removing Route Dampening from a Route

You can un-suppress routes by removing route flap dampening from the routes. The Layer 3 Switch allows you to un-suppress all routes at once or un-suppress individual routes.

To un-suppress all the suppressed routes, enter the following command at the Privileged EXEC level of the CLI:

```
FESX424 Router# clear ip bgp damping
```

Syntax: clear ip bgp damping [<ip-addr> <ip-mask>]

The <ip-addr> parameter specifies a particular network.

The <ip-mask> parameter specifies the network's mask.

To un-suppress a specific route, enter a command such as the following:

```
FESX424 Router# clear ip bgp damping 209.157.22.0 255.255.255.0
```

This command un-suppresses only the route(s) for network 209.157.22.0/24.

Removing Route Dampening from a Neighbor's Routes Suppressed Due to Aggregation

You can selectively unsuppress more-specific routes that have been suppressed due to aggregation, and allow the routes to be advertised to a specific neighbor or peer group.

Here is an example.

```
FESX424 Router(config-bgp-router)# aggregate-address 209.1.0.0 255.255.0.0
summary-only
FESX424 Router(config-bgp-router)# show ip bgp route 209.1.0.0/16 longer
Number of BGP Routes matching display condition : 2
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      209.1.0.0/16       0.0.0.0                    101        32768  BAL
         AS_PATH:
2      209.1.44.0/24      10.2.0.1        1          101        32768  BLS
```

The **aggregate-address** command configures an aggregate address. The **summary-only** parameter prevents the Layer 3 Switch from advertising more specific routes contained within the aggregate route. The **show ip bgp route** command shows that the more specific routes aggregated into 209.1.0.0/16 have been suppressed. In this case, the route to 209.1.44.0/24 has been suppressed. The following command indicates that the route is not being advertised to the Layer 3 Switch's BGP4 neighbors.

```
FESX424 Router(config-bgp-router)# show ip bgp route 209.1.44.0/24
Number of BGP Routes matching display condition : 1
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      209.1.44.0/24      10.2.0.1        1          101        32768  BLS
         AS_PATH:
       Route is not advertised to any peers
```

If you want to override the **summary-only** parameter and allow a specific route to be advertised to a neighbor, enter commands such as the following:

```
FESX424 Router(config)# ip prefix-list Unsuppress1 permit 209.1.44.0/24
FESX424 Router(config)# route-map RouteMap1 permit 1
FESX424 Router(config-routemap RouteMap1)# match prefix-list Unsuppress1
FESX424 Router(config-routemap RouteMap1)# exit
FESX424 Router(config)# router bgp
FESX424 Router(config-bgp-router)# neighbor 10.1.0.2 unsuppress-map RouteMap1
FESX424 Router(config-bgp-router)# clear ip bgp neighbor 10.1.0.2 soft-out
```

The **ip prefix-list** command configures an IP prefix list for network 209.1.44.0/24, which is the route you want to unsuppress. The next two commands configure a route map that uses the prefix list as input. The **neighbor** command enables the Layer 3 Switch to advertise the routes specified in the route map to neighbor 10.1.0.2. The **clear** command performs a soft reset of the session with the neighbor so that the Layer 3 Switch can advertise the unsuppressed route.

Syntax: [no] neighbor <ip-addr> | <peer-group-name> unsuppress-map <map-name>

The following command verifies that the route has been unsuppressed.

```
FESX424 Router(config-bgp-router)# show ip bgp route 209.1.44.0/24
Number of BGP Routes matching display condition : 1
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      209.1.44.0/24      10.2.0.1        1          101        32768  BLS
         AS_PATH:
       Route is advertised to 1 peers:
        10.1.0.2(4)
```

Displaying and Clearing Route Flap Dampening Statistics

The software provides many options for displaying and clearing route flap statistics. To display the statistics, use either of the following methods.

Displaying Route Flap Dampening Statistics

To display route dampening statistics or all the dampened routes, enter the following command at any level of the CLI:

```
FESX424 Router# show ip bgp flap-statistics

Total number of flapping routes: 414
    Status Code  >:best d:damped h:history *:valid
    Network            From            Flaps Since    Reuse    Path
h>  192.50.206.0/23    166.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 701
h>  203.255.192.0/20   166.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 7018
h>  203.252.165.0/24   166.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 7018
h>  192.50.208.0/23    166.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 701
h>  133.33.0.0/16      166.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 701
*>  204.17.220.0/24    166.90.213.77   1     0 :1 :4  0 :0 :0  65001 4355 701 62
```

Syntax: show ip bgp flap-statistics [regular-expression <regular-expression> | <address> <mask> [longer-prefixes] | neighbor <ip-addr>]

The **regular-expression** <regular-expression> parameter is a regular expression. The regular expressions are the same ones supported for BGP4 AS-path filters. See “Using Regular Expressions” on page 21-42.

The <address> <mask> parameter specifies a particular route. If you also use the optional **longer-prefixes** parameter, then all statistics for routes that match the specified route or have a longer prefix than the specified route are displayed. For example, if you specify **209.157.0.0 longer**, then all routes with the prefix 209.157. or that have a longer prefix (such as 209.157.22.) are displayed.

The **neighbor** <ip-addr> parameter displays route flap dampening statistics only for routes learned from the specified neighbor. You also can display route flap statistics for routes learned from a neighbor by entering the following command: **show ip bgp neighbor** <ip-addr> **flap-statistics**.

This display shows the following information.

Table 21.4: Route Flap Dampening Statistics

This Field...	Displays...
Total number of flapping routes	The total number of routes in the Layer 3 Switch’s BGP4 route table that have changed state and thus have been marked as flapping routes.

Table 21.4: Route Flap Dampening Statistics

This Field...	Displays...
Status code	Indicates the dampening status of the route, which can be one of the following: • > – This is the best route among those in the BGP4 route table to the route's destination. • d – This route is currently dampened, and thus unusable. • h – The route has a history of flapping and is unreachable now. • * – The route has a history of flapping but is currently usable.
Network	The destination network of the route.
From	The neighbor that sent the route to the Layer 3 Switch.
Flaps	The number of flaps (state changes) the route has experienced.
Since	The amount of time since the first flap of this route.
Reuse	The amount of time remaining until this route will be un-suppressed and thus be usable again.
Path	Shows the AS-path information for the route.

You also can display all the dampened routes by entering the following command:
show ip bgp dampened-paths.

Clearing Route Flap Dampening Statistics

To clear route flap dampening statistics, use the following CLI method.

NOTE: Clearing the dampening statistics for a route does not change the dampening status of the route.

To clear all the route dampening statistics, enter the following command at any level of the CLI:

```
FESX424 Router# clear ip bgp flap-statistics
```

Syntax: clear ip bgp flap-statistics [regular-expression <regular-expression> | <address> <mask> | neighbor <ip-addr>]

The parameters are the same as those for the **show ip bgp flap-statistics** command (except the **longer-prefixes** option is not supported). See "Displaying Route Flap Dampening Statistics" on page 21-63.

NOTE: The **clear ip bgp damping** command not only clears statistics but also un-suppresses the routes. See "Displaying Route Flap Dampening Statistics" on page 21-63.

Generating Traps for BGP

You can enable and disable SNMP traps for BGP. BGP traps are enabled by default.

To enable BGP traps after they have been disabled, enter the following command:

```
FESX424 Router(config)# snmp-server enable traps bgp
```

Syntax: [no] snmp-server enable traps bgp

Use the **no** form of the command to disable BGP traps.

Displaying BGP4 Information

You can display the following configuration information and statistics for the BGP4 protocol on the router:

- Summary BGP4 configuration information for the router
- Active BGP4 configuration information (the BGP4 information in the running-config)
- CPU utilization statistics
- Neighbor information
- Peer-group information
- Information about the paths from which BGP4 selects routes
- Summary BGP4 route information
- The router's BGP4 route table
- Route flap dampening statistics
- Active route maps (the route map configuration information in the running-config)

Displaying Summary BGP4 Information

You can display the local AS number, the maximum number of routes and neighbors supported, and some BGP4 statistics.

To view summary BGP4 information for the router, enter the following command at any CLI prompt:

```
FESX424 Router# show ip bgp summary
  BGP4 Summary
  Router ID: 101.0.0.1   Local AS Number : 4
  Confederation Identifier : not configured
  Confederation Peers: 4 5
  Maximum Number of Paths Supported for Load Sharing : 1
  Number of Neighbors Configured : 11
  Number of Routes Installed : 2
  Number of Routes Advertising to All Neighbors : 8
  Number of Attribute Entries Installed : 6
  Neighbor Address  AS#   State   Time     Rt:Accepted Filtered Sent   ToSend
  1.2.3.4           200   ADMDN   0h44m56s    0        0        0      2
  10.0.0.2          5     ADMDN   0h44m56s    0        0        0      0
  10.1.0.2          5     ESTAB   0h44m56s    1        11       0      0
  10.2.0.2          5     ESTAB   0h44m55s    1        0        0      0
  10.3.0.2          5     ADMDN   0h25m28s    0        0        0      0
  10.4.0.2          5     ADMDN   0h25m31s    0        0        0      0
  10.5.0.2          5     CONN    0h 0m 8s    0        0        0      0
  10.7.0.2          5     ADMDN   0h44m56s    0        0        0      0
  100.0.0.1         4     ADMDN   0h44m56s    0        0        0      2
  102.0.0.1         4     ADMDN   0h44m56s    0        0        0      2
  150.150.150.150   0     ADMDN   0h44m56s    0        0        0      2
```

This display shows the following information.

Table 21.5: BGP4 Summary Information

This Field...	Displays...
Router ID	The Layer 3 Switch's router ID.
Local AS Number	The BGP4 AS number the router is in.
Confederation Identifier	The AS number of the confederation the Layer 3 Switch is in.
Confederation Peers	The numbers of the local ASs contained in the confederation. This list matches the confederation peer list you configure on the Layer 3 Switch.
Maximum Number of Paths Supported for Load Sharing	The maximum number of route paths across which the device can balance traffic to the same destination. The feature is enabled by default but the default number of paths is 1. You can increase the number from 2 – 4 paths. See "Changing the Maximum Number of Paths for BGP4 Load Sharing" on page 21-22.
Number of Neighbors Configured	The number of BGP4 neighbors configured on this Layer 3 Switch.
Number of Routes Installed	The number of BGP4 routes in the router's BGP4 route table. To display the BGP4 route table, see "Displaying the BGP4 Route Table" on page 21-88.
Number of Routes Advertising to All Neighbors	The total of the RtSent and RtToSend columns for all neighbors.
Number of Attribute Entries Installed	The number of BGP4 route-attribute entries in the router's route-attributes table. To display the route-attribute table, see "Displaying BGP4 Route-Attribute Entries" on page 21-96.
Neighbor Address	The IP addresses of this router's BGP4 neighbors.
AS#	The AS number.

Table 21.5: BGP4 Summary Information (Continued)

This Field...	Displays...

State	The state of this router's neighbor session with each neighbor. The states are from this router's perspective of the session, not the neighbor's perspective. The state values are based on the BGP4 state machine values described in RFC 1771 and can be one of the following for each router:
	• IDLE – The BGP4 process is waiting to be started. Usually, enabling BGP4 or establishing a neighbor session starts the BGP4 process.
	  • A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes.
	• ADMND – The neighbor has been administratively shut down. See "Administratively Shutting Down a Session with a BGP4 Neighbor" on page 21-20.
	  • A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes.
	• CONNECT – BGP4 is waiting for the connection process for the TCP neighbor session to be completed.
	• ACTIVE – BGP4 is waiting for a TCP connection from the neighbor.
	Note: If the state frequently changes between CONNECT and ACTIVE, there may be a problem with the TCP connection.
	• OPEN SENT – BGP4 is waiting for an Open message from the neighbor.
	• OPEN CONFIRM – BGP4 has received an OPEN message from the neighbor and is now waiting for either a KEEPALIVE or NOTIFICATION message. If the router receives a KEEPALIVE message from the neighbor, the state changes to Established. If the message is a NOTIFICATION, the state changes to Idle.
	• ESTABLISHED – BGP4 is ready to exchange UPDATE packets with the neighbor.
	  • If there is more BGP data in the TCP receiver queue, a plus sign (+) is also displayed.
	Note: If you display information for the neighbor using the **show ip bgp neighbor** <ip-addr> command, the TCP receiver queue value will be greater than 0.

Time	The time that has passed since the state last changed.
Accepted	The number of routes received from the neighbor that this router installed in the BGP4 route table. Usually, this number is lower than the RoutesRcvd number. The difference indicates that this router filtered out some of the routes received in the UPDATE messages.

Table 21.5: BGP4 Summary Information (Continued)

This Field...	Displays...
Filtered	The routes or prefixes that have been filtered out. • If soft reconfiguration is enabled, this field shows how many routes were filtered out (not placed in the BGP4 route table) but retained in memory. • If soft reconfiguration is not enabled, this field shows the number of BGP4 routes that have been filtered out.
Sent	The number of BGP4 routes that the Layer 3 Switch has sent to the neighbor.
ToSend	The number of routes the Layer 3 Switch has queued to send to this neighbor.

Displaying the Active BGP4 Configuration

To view the active BGP4 configuration information contained in the running-config without displaying the entire running-config, use the following CLI method.

To display the device's active BGP4 configuration, enter the following command at any level of the CLI:

```
FESX424 Router# show ip bgp config
Current BGP configuration:
router bgp
 address-filter  1 deny  any   any
 as-path-filter  1 permit ^65001$
 local-as 65002
 maximum-paths 4
 neighbor pg1 peer-group
 neighbor pg1 remote-as 65001
 neighbor pg1 description "FESX424 Router group 1"
 neighbor pg1 distribute-list out 1
 neighbor 192.169.100.1 peer-group pg1
 neighbor 192.169.101.1 peer-group pg1
 neighbor 192.169.102.1 peer-group pg1
 neighbor 192.169.201.1 remote-as 65101
 neighbor 192.169.201.1 shutdown
 neighbor 192.169.220.3 remote-as 65432
 network 1.1.1.0 255.255.255.0
 network 2.2.2.0 255.255.255.0
 redistribute connected
```

Syntax: show ip bgp config

Displaying CPU Utilization Statistics

You can display CPU utilization statistics for BGP4 and other IP protocols.

To display CPU utilization statistics for BGP4 for the previous one-second, one-minute, five-minute, and fifteen-minute intervals, enter the following command at any level of the CLI:

```
FESX424 Router# show process cpu
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.01      0.03      0.09      0.22              9
BGP              0.04      0.06      0.08      0.14             13
GVRP             0.00      0.00      0.00      0.00              0
ICMP             0.00      0.00      0.00      0.00              0
IP               0.00      0.00      0.00      0.00              0
OSPF             0.00      0.00      0.00      0.00              0
RIP              0.00      0.00      0.00      0.00              0
STP              0.00      0.00      0.00      0.00              0
VRRP             0.00      0.00      0.00      0.00              0
```

If the software has been running less than 15 minutes (the maximum interval for utilization statistics), the command indicates how long the software has been running. Here is an example:

```
FESX424 Router# show process cpu
The system has only been up for 6 seconds.
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.01      0.00      0.00      0.00              0
BGP              0.00      0.00      0.00      0.00              0
GVRP             0.00      0.00      0.00      0.00              0
ICMP             0.01      0.00      0.00      0.00              1
IP               0.00      0.00      0.00      0.00              0
OSPF             0.00      0.00      0.00      0.00              0
RIP              0.00      0.00      0.00      0.00              0
STP              0.00      0.00      0.00      0.00              0
VRRP             0.00      0.00      0.00      0.00              0
```

To display utilization statistics for a specific number of seconds, enter a command such as the following:

```
FESX424 Router# show process cpu 2
Statistics for last 1 sec and 80 ms
Process Name   Sec(%)   Time(ms)
ARP              0.00          0
BGP              0.00          0
GVRP             0.00          0
ICMP             0.01          1
IP               0.00          0
OSPF             0.00          0
RIP              0.00          0
STP              0.01          0
VRRP             0.00          0
```

When you specify how many seconds' worth of statistics you want to display, the software selects the sample that most closely matches the number of seconds you specified. In this example, statistics are requested for the previous two seconds. The closest sample available is actually for the previous 1 second plus 80 milliseconds.

Syntax: show process cpu [<num>]

The <num> parameter specifies the number of seconds and can be from 1 – 900. If you use this parameter, the command lists the usage statistics only for the specified number of seconds. If you do not use this parameter, the command lists the usage statistics for the previous one-second, one-minute, five-minute, and fifteen-minute intervals.

Displaying Summary Neighbor Information

To display summary neighbor information, enter a command such as the following at any level of the CLI:

```
FESX424 Router(config-bgp-router)# show ip bgp neighbor 192.168.4.211 routes-
summary
1   IP Address: 192.168.4.211
Routes Accepted/Installed:1,  Filtered/Kept:11,  Filtered:11
   Routes Selected as BEST Routes:1
      BEST Routes not Installed in IP Forwarding Table:0
   Unreachable Routes (no IGP Route for NEXTHOP):0
   History Routes:0

NLRIs Received in Update Message:24,  Withdraws:0 (0),  Replacements:1
   NLRIs Discarded due to
      Maximum Prefix Limit:0, AS Loop:0
      Invalid Nexthop:0, Invalid Nexthop Address:0.0.0.0
      Duplicated Originator_ID:0, Cluster_ID:0

Routes Advertised:0,  To be Sent:0,  To be Withdrawn:0
NLRIs Sent in Update Message:0,  Withdraws:0,  Replacements:0

Peer Out of Memory Count for:
   Receiving Update Messages:0, Accepting Routes(NLRI):0
```

Syntax: show ip bgp neighbors [<ip-addr>] | [route-summary]

This display shows the following information.

Table 21.6: BGP4 Route Summary Information for a Neighbor

This Field...	Displays...
IP Address	The IP address of the neighbor
Routes Received	How many routes the Layer 3 Switch has received from the neighbor during the current BGP4 session. • Accepted/Installed – Indicates how many of the received routes the Layer 3 Switch accepted and installed in the BGP4 route table. • Filtered/Kept – Indicates how many routes were filtered out, but were nonetheless retained in memory for use by the soft reconfiguration feature. • Filtered – Indicates how many of the received routes were filtered out.
Routes Selected as BEST Routes	The number of routes that the Layer 3 Switch selected as the best routes to their destinations.
BEST Routes not Installed in IP Forwarding Table	The number of routes received from the neighbor that are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table because the Layer 3 Switch received better routes from other sources (such as OSPF, RIP, or static IP routes).

Table 21.6: BGP4 Route Summary Information for a Neighbor (Continued)

This Field...	Displays...
Unreachable Routes	The number of routes received from the neighbor that are unreachable because the Layer 3 Switch does not have a valid RIP, OSPF, or static route to the next hop.
History Routes	The number of routes that are down but are being retained for route flap dampening purposes.
NLRIs Received in Update Message	The number of routes received in Network Layer Reachability (NLRI) format in UPDATE messages. • Withdraws – The number of withdrawn routes the Layer 3 Switch has received. • Replacements – The number of replacement routes the Layer 3 Switch has received.
NLRIs Discarded due to	Indicates the number of times the Layer 3 Switch discarded an NLRI for the neighbor due to the following reasons: • Maximum Prefix Limit – The Layer 3 Switch's configured maximum prefix amount had been reached. • AS Loop – An AS loop occurred. An AS loop occurs when the BGP4 AS-path attribute contains the local AS number. • Invalid Nexthop – The next hop value was not acceptable. • Duplicated Originator_ID – The originator ID was the same as the local router ID. • Cluster_ID – The cluster list contained the local cluster ID, or contained the local router ID (see above) if the cluster ID is not configured.
Routes Advertised	The number of routes the Layer 3 Switch has advertised to this neighbor. • To be Sent – The number of routes the Layer 3 Switch has queued to send to this neighbor. • To be Withdrawn – The number of NLRIs for withdrawing routes the Layer 3 Switch has queued up to send to this neighbor in UPDATE messages.
NLRIs Sent in Update Message	The number of NLRIs for new routes the Layer 3 Switch has sent to this neighbor in UPDATE messages. • Withdraws – The number of routes the Layer 3 Switch has sent to the neighbor to withdraw. • Replacements – The number of routes the Layer 3 Switch has sent to the neighbor to replace routes the neighbor already has.

Table 21.6: BGP4 Route Summary Information for a Neighbor (Continued)

This Field...	Displays...
Peer Out of Memory Count for	Statistics for the times the Layer 3 Switch has run out of BGP4 memory for the neighbor during the current BGP4 session. • Receiving Update Messages – The number of times UPDATE messages were discarded because there was no memory for attribute entries. • Accepting Routes(NLRI) – The number of NLRIs discarded because there was no memory for NLRI entries. This count is not included in the Receiving Update Messages count. • Attributes – The number of times there was no memory for BGP4 attribute entries. • Outbound Routes(RIB-out) – The number of times there was no memory to place a “best” route into the neighbor's route information base (Adj-RIB-Out) for routes to be advertised.

Displaying BGP4 Neighbor Information

To view BGP4 neighbor information including the values for all the configured parameters, enter the following command.

NOTE: The display shows all the configured parameters for the neighbor. Only the parameters that have values different from their defaults are shown.

```
FESX424 Router(config-bgp-router)# show ip bgp neighbor 10.4.0.2
1   IP Address: 10.4.0.2, AS: 5 (EBGP), RouterID: 100.0.0.1
      Description: neighbor 10.4.0.2
    State: ESTABLISHED, Time: 0h1m0s, KeepAliveTime: 0, HoldTime: 0
      PeerGroup: pg1
      Multihop-EBGP: yes, ttl: 1
      RouteReflectorClient: yes
      SendCommunity: yes
      NextHopSelf: yes
      DefaultOriginate: yes (default sent)
      MaximumPrefixLimit: 90000
      RemovePrivateAs: : yes
      RefreshCapability: Received
    Route Filter Policies:
      Distribute-list: (out) 20
      Filter-list: (in) 30
      Prefix-list: (in) pf1
      Route-map: (in) setnp1  (out) setnp2
    Messages:    Open    Update  KeepAlive Notification Refresh-Req
      Sent    : 1       1       1         0            0
      Received: 1       8       1         0            0
    Last Update Time: NLRI       Withdraw          NLRI       Withdraw
                 Tx: 0h0m59s    ---           Rx: 0h0m59s    ---
    Last Connection Reset Reason:Unknown
      Notification Sent:     Unspecified
      Notification Received: Unspecified
    TCP Connection state: ESTABLISHED
      Local host:  10.4.0.1, Local  Port: 179
      Remote host: 10.4.0.2, Remote Port: 8053
      ISentSeq:   52837276  SendNext:   52837392  TotUnAck:          0
      TotSent:         116  ReTrans:           0  UnAckSeq:   52837392
      IRcvSeq:  2155052043  RcvNext:  2155052536  SendWnd:       16384
      TotalRcv:        493  DupliRcv:          0  RcvWnd:        16384
      SendQue:           0  RcvQue:            0  CngstWnd:       1460
```

This example shows how to display information for a specific neighbor, by specifying the neighbor's IP address with the command. None of the other display options are used; thus, all of the information is displayed for the neighbor. The number in the far left column indicates the neighbor for which information is displayed. When you list information for multiple neighbors, this number makes the display easier to read.

The TCP statistics at the end of the display show status for the TCP session with the neighbor. Most of the fields show information stored in the Layer 3 Switch's Transmission Control Block (TCB) for the TCP session between the Layer 3 Switch and its neighbor. These fields are described in detail in section 3.2 of RFC 793, "Transmission Control Protocol Functional Specification".

Syntax: show ip bgp neighbors [<ip-addr> [advertised-routes [detail [<ip-addr>[/<mask-bits>]]]] | [attribute-entries [detail]] | [flap-statistics] | [last-packet-with-error] | [received prefix-filter] | [received-routes] | [routes [best] | [detail [best] | [not-installed-best] | [unreachable]] | [rib-out-routes [<ip-addr>/<mask-bits> | <ip-addr> <net-mask> | detail]] | [routes-summary]]

The <ip-addr> option lets you narrow the scope of the command to a specific neighbor.

The **advertised-routes** option displays only the routes that the Layer 3 Switch has advertised to the neighbor during the current BGP4 neighbor session.

The **attribute-entries** option shows the attribute-entries associated with routes received from the neighbor.

The **flap-statistics** option shows the route flap statistics for routes received from or sent to the neighbor.

The **last-packet-with-error** option displays the last packet from the neighbor that contained an error. The packet's contents are displayed in decoded (human-readable) format.

The **received prefix-filter** option shows the Outbound Route Filters (ORFs) received from the neighbor. This option applies to cooperative route filtering.

The **received-routes** option lists all the route information received in route updates from the neighbor since the soft reconfiguration feature was enabled. See “Using Soft Reconfiguration” on page 21-100.

The **routes** option lists the routes received in UPDATE messages from the neighbor. You can specify the following additional options:

- **best** – Displays the routes received from the neighbor that the Layer 3 Switch selected as the best routes to their destinations.
- **not-installed-best** – Displays the routes received from the neighbor that are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table because the Layer 3 Switch received better routes from other sources (such as OSPF, RIP, or static IP routes).
- **unreachable** – Displays the routes that are unreachable because the Layer 3 Switch does not have a valid RIP, OSPF, or static route to the next hop.
- **detail** – Displays detailed information for the specified routes. You can refine your information request by also specifying one of the options above (**best**, **not-installed-best**, or **unreachable**).

The **rib-out-routes** option lists the route information base (RIB) for outbound routes. You can display all the routes or specify a network address.

The **routes-summary** option displays a summary of the following information:

- Number of routes received from the neighbor
- Number of routes accepted by this Layer 3 Switch from the neighbor
- Number of routes this Layer 3 Switch filtered out of the UPDATES received from the neighbor and did not accept
- Number of routes advertised to the neighbor
- Number of attribute entries associated with routes received from or advertised to the neighbor.

This display shows the following information.

Table 21.7: BGP4 Neighbor Information

This Field...	Displays...
IP Address	The IP address of the neighbor.
AS	The AS the neighbor is in.

Table 21.7: BGP4 Neighbor Information (Continued)

This Field...	Displays...
EBGP/IBGP	Whether the neighbor session is an IBGP session, an EBGP session, or a confederation EBGP session. • EBGP – The neighbor is in another AS. • EBGP_Confed – The neighbor is a member of another sub-AS in the same confederation. • IBGP – The neighbor is in the same AS.
RouterID	The neighbor's router ID.
Description	The description you gave the neighbor when you configured it on the Layer 3 Switch.

Table 21.7: BGP4 Neighbor Information (Continued)

This Field...	Displays...

State	The state of the router's session with the neighbor. The states are from this router's perspective of the session, not the neighbor's perspective. The state values are based on the BGP4 state machine values described in RFC 1771 and can be one of the following for each router:
	• IDLE – The BGP4 process is waiting to be started. Usually, enabling BGP4 or establishing a neighbor session starts the BGP4 process.
	  • A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes.
	• ADMND – The neighbor has been administratively shut down. See "Administratively Shutting Down a Session with a BGP4 Neighbor" on page 21-20.
	  • A minus sign (-) indicates that the session has gone down and the software is clearing or removing routes.
	• CONNECT – BGP4 is waiting for the connection process for the TCP neighbor session to be completed.
	• ACTIVE – BGP4 is waiting for a TCP connection from the neighbor.
	Note: If the state frequently changes between CONNECT and ACTIVE, there may be a problem with the TCP connection.
	• OPEN SENT – BGP4 is waiting for an Open message from the neighbor.
	• OPEN CONFIRM – BGP4 has received an OPEN message from the neighbor and is now waiting for either a KEEPALIVE or NOTIFICATION message. If the router receives a KEEPALIVE message from the neighbor, the state changes to Established. If the message is a NOTIFICATION, the state changes to Idle.
	• ESTABLISHED – BGP4 is ready to exchange UPDATE messages with the neighbor.
	  • If there is more BGP data in the TCP receiver queue, a plus sign (+) is also displayed.
	Note: If you display information for the neighbor using the **show ip bgp neighbor** <ip-addr> command, the TCP receiver queue value will be greater than 0.

Time	The amount of time this session has been in its current state.
KeepAliveTime	The keep alive time, which specifies how often this router sends keep alive messages to the neighbor. See "Changing the Keep Alive Time and Hold Time" on page 21-21.
HoldTime	The hold time, which specifies how many seconds the router will wait for a KEEPALIVE or UPDATE message from a BGP4 neighbor before deciding that the neighbor is dead. See "Changing the Keep Alive Time and Hold Time" on page 21-21.
PeerGroup	The name of the peer group the neighbor is in, if applicable.

Table 21.7: BGP4 Neighbor Information (Continued)

This Field...	Displays...
Multihop-EBGP	Whether this option is enabled for the neighbor.
RouteReflectorClient	Whether this option is enabled for the neighbor.
SendCommunity	Whether this option is enabled for the neighbor.
NextHopSelf	Whether this option is enabled for the neighbor.
DefaultOriginate	Whether this option is enabled for the neighbor.
MaximumPrefixLimit	Lists the maximum number of prefixes the Layer 3 Switch will accept from this neighbor.
RemovePrivateAs	Whether this option is enabled for the neighbor.
RefreshCapability	Whether this Layer 3 Switch has received confirmation from the neighbor that the neighbor supports the dynamic refresh capability.
CooperativeFilteringCapability	Whether the neighbor is enabled for cooperative route filtering.
Distribute-list	Lists the distribute list parameters, if configured.
Filter-list	Lists the filter list parameters, if configured.
Prefix-list	Lists the prefix list parameters, if configured.
Route-map	Lists the route map parameters, if configured.
Messages Sent	The number of messages this router has sent to the neighbor. The display shows statistics for the following message types: • Open • Update • KeepAlive • Notification • Refresh-Req
Messages Received	The number of messages this router has received from the neighbor. The message types are the same as for the Message Sent field.
Last Update Time	Lists the last time updates were sent and received for the following: • NLRIs • Withdraws

Table 21.7: BGP4 Neighbor Information (Continued)

This Field...	Displays...

Last Connection Reset Reason	The reason the previous session with this neighbor ended. The reason can be one of the following:
	• Reasons described in the BGP specifications:
	  • Message Header Error
	  • Connection Not Synchronized
	  • Bad Message Length
	  • Bad Message Type
	  • OPEN Message Error
	  • Unsupported Version Number
	  • Bad Peer AS Number
	  • Bad BGP Identifier
	  • Unsupported Optional Parameter
	  • Authentication Failure
	  • Unacceptable Hold Time
	  • Unsupported Capability
	  • UPDATE Message Error
	  • Malformed Attribute List
	  • Unrecognized Well-known Attribute
	  • Missing Well-known Attribute
	  • Attribute Flags Error
	  • Attribute Length Error
	  • Invalid ORIGIN Attribute
	  • Invalid NEXT_HOP Attribute
	  • Optional Attribute Error
	  • Invalid Network Field
	  • Malformed AS_PATH
	  • Hold Timer Expired
	  • Finite State Machine Error
	  • Rcv Notification

Table 21.7: BGP4 Neighbor Information (Continued)

This Field...	Displays...

Last Connection Reset Reason (cont.)	• Reasons specific to the Foundry implementation:
	  • Reset All Peer Sessions
	  • User Reset Peer Session
	  • Port State Down
	  • Peer Removed
	  • Peer Shutdown
	  • Peer AS Number Change
	  • Peer AS Confederation Change
	  • TCP Connection KeepAlive Timeout
	  • TCP Connection Closed by Remote
	  • TCP Data Stream Error Detected

Table 21.7: BGP4 Neighbor Information (Continued)

This Field...	Displays...

Notification Sent	If the router receives a NOTIFICATION message from the neighbor, the message contains an error code corresponding to one of the following errors. Some errors have subcodes that clarify the reason for the error. Where applicable, the subcode messages are listed underneath the error code messages.
	• Message Header Error
	  • Connection Not Synchronized
	  • Bad Message Length
	  • Bad Message Type
	  • Unspecified
	• Open Message Error
	  • Unsupported Version
	  • Bad Peer As
	  • Bad BGP Identifier
	  • Unsupported Optional Parameter
	  • Authentication Failure
	  • Unacceptable Hold Time
	  • Unspecified
	• Update Message Error
	  • Malformed Attribute List
	  • Unrecognized Attribute
	  • Missing Attribute
	  • Attribute Flag Error
	  • Attribute Length Error
	  • Invalid Origin Attribute
	  • Invalid NextHop Attribute
	  • Optional Attribute Error
	  • Invalid Network Field
	  • Malformed AS Path
	  • Unspecified
	• Hold Timer Expired
	• Finite State Machine Error
	• Cease
	• Unspecified

Notification Received	See above.

Table 21.7: BGP4 Neighbor Information (Continued)

This Field...	Displays...
TCP Connection state	The state of the connection with the neighbor. The connection can have one of the following states: • LISTEN – Waiting for a connection request. • SYN-SENT – Waiting for a matching connection request after having sent a connection request. • SYN-RECEIVED – Waiting for a confirming connection request acknowledgment after having both received and sent a connection request. • ESTABLISHED – Data can be sent and received over the connection. This is the normal operational state of the connection. • FIN-WAIT-1 – Waiting for a connection termination request from the remote TCP, or an acknowledgment of the connection termination request previously sent. • FIN-WAIT-2 – Waiting for a connection termination request from the remote TCP. • CLOSE-WAIT – Waiting for a connection termination request from the local user. • CLOSING – Waiting for a connection termination request acknowledgment from the remote TCP. • LAST-ACK – Waiting for an acknowledgment of the connection termination request previously sent to the remote TCP (which includes an acknowledgment of its connection termination request). • TIME-WAIT – Waiting for enough time to pass to be sure the remote TCP received the acknowledgment of its connection termination request. • CLOSED – There is no connection state.
Byte Sent	The number of bytes sent.
Byte Received	The number of bytes received.
Local host	The IP address of the Layer 3 Switch.
Local port	The TCP port the Layer 3 Switch is using for the BGP4 TCP session with the neighbor.
Remote host	The IP address of the neighbor.
Remote port	The TCP port the neighbor is using for the BGP4 TCP session with the Layer 3 Switch.
ISentSeq	The initial send sequence number for the session.
SendNext	The next sequence number to be sent.
TotUnAck	The number of sequence numbers sent by the Layer 3 Switch that have not been acknowledged by the neighbor.
TotSent	The number of sequence numbers sent to the neighbor.

Table 21.7: BGP4 Neighbor Information (Continued)

This Field...	Displays...
ReTrans	The number of sequence numbers that the Layer 3 Switch retransmitted because they were not acknowledged.
UnAckSeq	The current acknowledged sequence number.
IRcvSeq	The initial receive sequence number for the session.
RcvNext	The next sequence number expected from the neighbor.
SendWnd	The size of the send window.
TotalRcv	The number of sequence numbers received from the neighbor.
DupliRcv	The number of duplicate sequence numbers received from the neighbor.
RcvWnd	The size of the receive window.
SendQue	The number of sequence numbers in the send queue.
RcvQue	The number of sequence numbers in the receive queue.
CngstWnd	The number of times the window has changed.

Displaying Route Information for a Neighbor

You can display routes based on the following criteria:

- A summary of the routes for a specific neighbor.
- The routes received from the neighbor that the Layer 3 Switch selected as the best routes to their destinations.
- The routes received from the neighbor that are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table because the Layer 3 Switch received better routes from other sources (such as OSPF, RIP, or static IP routes).
- The routes that are unreachable because the Layer 3 Switch does not have a valid RIP, OSPF, or static route to the next hop.
- Routes for a specific network advertised by the Layer 3 Switch to the neighbor.
- The Routing Information Base (RIB) for a specific network advertised to the neighbor. You can display the RIB regardless of whether the Layer 3 Switch has already sent it to the neighbor.

To display route information for a neighbor, use the following CLI methods.

Displaying Summary Route Information

To display summary route information, enter a command such as the following at any level of the CLI:

```
FESX424 Router(config-bgp-router)# show ip bgp neighbor 10.1.0.2 routes-summary
1   IP Address: 10.1.0.2
Routes Accepted/Installed:1,  Filtered/Kept:11,  Filtered:11
   Routes Selected as BEST Routes:1
      BEST Routes not Installed in IP Forwarding Table:0
   Unreachable Routes (no IGP Route for NEXTHOP):0
   History Routes:0

NLRIs Received in Update Message:24,  Withdraws:0 (0),  Replacements:1
   NLRIs Discarded due to
      Maximum Prefix Limit:0, AS Loop:0
      Invalid Nexthop:0, Invalid Nexthop Address:0.0.0.0
      Duplicated Originator_ID:0, Cluster_ID:0

Routes Advertised:0,  To be Sent:0,  To be Withdrawn:0
NLRIs Sent in Update Message:0,  Withdraws:0,  Replacements:0

Peer Out of Memory Count for:
   Receiving Update Messages:0, Accepting Routes(NLRI):0
   Attributes:0, Outbound Routes(RIB-out):0
```

This display shows the following information.

Table 21.8: BGP4 Route Summary Information for a Neighbor

This Field...	Displays...
Routes Received	How many routes the Layer 3 Switch has received from the neighbor during the current BGP4 session. • Accepted/Installed – Indicates how many of the received routes the Layer 3 Switch accepted and installed in the BGP4 route table. • Filtered – Indicates how many of the received routes the Layer 3 Switch did not accept or install because they were denied by filters on the Layer 3 Switch.
Routes Selected as BEST Routes	The number of routes that the Layer 3 Switch selected as the best routes to their destinations.
BEST Routes not Installed in IP Forwarding Table	The number of routes received from the neighbor that are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table because the Layer 3 Switch received better routes from other sources (such as OSPF, RIP, or static IP routes).
Unreachable Routes	The number of routes received from the neighbor that are unreachable because the Layer 3 Switch does not have a valid RIP, OSPF, or static route to the next hop.
History Routes	The number of routes that are down but are being retained for route flap dampening purposes.

Table 21.8: BGP4 Route Summary Information for a Neighbor (Continued)

This Field...	Displays...
NLRIs Received in Update Message	The number of routes received in Network Layer Reachability (NLRI) format in UPDATE messages. • Withdraws – The number of withdrawn routes the Layer 3 Switch has received. • Replacements – The number of replacement routes the Layer 3 Switch has received.
NLRIs Discarded due to	Indicates the number of times the Layer 3 Switch discarded an NLRI for the neighbor due to the following reasons: • Maximum Prefix Limit – The Layer 3 Switch's configured maximum prefix amount had been reached. • AS Loop – An AS loop occurred. An AS loop occurs when the BGP4 AS-path attribute contains the local AS number. • Invalid Nexthop – The next hop value was not acceptable. • Duplicated Originator_ID – The originator ID was the same as the local router ID. • Cluster_ID – The cluster list contained the local cluster ID, or contained the local router ID (see above) if the cluster ID is not configured.
Routes Advertised	The number of routes the Layer 3 Switch has advertised to this neighbor. • To be Sent – The number of routes the Layer 3 Switch has queued to send to this neighbor. • To be Withdrawn – The number of NLRIs for withdrawing routes the Layer 3 Switch has queued up to send to this neighbor in UPDATE messages.
NLRIs Sent in Update Message	The number of NLRIs for new routes the Layer 3 Switch has sent to this neighbor in UPDATE messages. • Withdraws – The number of routes the Layer 3 Switch has sent to the neighbor to withdraw. • Replacements – The number of routes the Layer 3 Switch has sent to the neighbor to replace routes the neighbor already has.

Table 21.8: BGP4 Route Summary Information for a Neighbor (Continued)

This Field...	Displays...
Peer Out of Memory Count for	Statistics for the times the Layer 3 Switch has run out of BGP4 memory for the neighbor during the current BGP4 session. • Receiving Update Messages – The number of times UPDATE messages were discarded because there was no memory for attribute entries. • Accepting Routes(NLRI) – The number of NLRIs discarded because there was no memory for NLRI entries. This count is not included in the Receiving Update Messages count. • Attributes – The number of times there was no memory for BGP4 attribute entries. • Outbound Routes(RIB-out) – The number of times there was no memory to place a “best” route into the neighbor's route information base (Adj-RIB-Out) for routes to be advertised.

Displaying Advertised Routes

To display the routes the Layer 3 Switch has advertised to a specific neighbor for a specific network, enter a command such as the following at any level of the CLI:

```
FESX424 Router# show ip bgp neighbors 192.168.4.211 advertised-routes
       There are 2 routes advertised to neighbor 192.168.4.211
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST I:IBGP L:LOCAL
       Network        Next Hop        Metric     LocPrf     Weight     Status
1      102.0.0.0/24   192.168.2.102   12                    32768      BL
2      200.1.1.0/24   192.168.2.102   0                     32768    BL
```

You also can enter a specific route, as in the following example:

```
FESX424 Router# show ip bgp neighbors 192.168.4.211 advertised 200.1.1.0/24
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST I:IBGP L:LOCAL
       Network        Next Hop        Metric     LocPrf     Weight     Status
1      200.1.1.0/24   192.168.2.102   0                     32768      BL
```

Syntax: show ip bgp neighbor <ip-addr> advertised-routes [<ip-addr>/<prefix>]

For information about the fields in this display, see Table 21.10 on page 21-91. The fields in this display also appear in the **show ip bgp** display.

Displaying the Best Routes

To display the routes received from a specific neighbor that are the “best” routes to their destinations, enter a command such as the following at any level of the CLI:

```
FESX424 Router(config-bgp-router)# show ip bgp neighbor 192.168.4.211 routes best
```

Syntax: show ip bgp neighbor <ip-addr> routes best

For information about the fields in this display, see Table 21.10 on page 21-91. The fields in this display also appear in the **show ip bgp** display.

Displaying the Best Routes that Were Nonetheless Not Installed in the IP Route Table

To display the BGP4 routes received from a specific neighbor that are the "best" routes to their destinations but are not installed in the Layer 3 Switch's IP route table, enter a command such as the following at any level of the CLI:

```
FESX424 Router(config-bgp-router)# show ip bgp neighbor 192.168.4.211 routes not-
installed-best
```

Each of the displayed routes is a valid path to its destination, but the Layer 3 Switch received another path from a different source (such as OSPF, RIP, or a static route) that has a lower administrative distance. The Layer 3 Switch always selects the path with the lowest administrative distance to install in the IP route table.

Syntax: show ip bgp neighbor <ip-addr> routes not-installed-best

For information about the fields in this display, see Table 21.10 on page 21-91. The fields in this display also appear in the **show ip bgp** display.

Displaying the Routes Whose Destinations Are Unreachable

To display BGP4 routes whose destinations are unreachable using any of the BGP4 paths in the BGP4 route table, enter a command such as the following at any level of the CLI:

```
FESX424 Router(config-bgp-router)# show ip bgp neighbor 192.168.4.211 routes
unreachable
```

Syntax: show ip bgp neighbor <ip-addr> routes unreachable

For information about the fields in this display, see Table 21.10 on page 21-91. The fields in this display also appear in the **show ip bgp** display.

Displaying the Adj-RIB-Out for a Neighbor

To display the Layer 3 Switch's current BGP4 Routing Information Base (Adj-RIB-Out) for a specific neighbor and a specific destination network, enter a command such as the following at any level of the CLI:

```
FESX424 Router(config-bgp-router)# show ip bgp neighbor 192.168.4.211 rib-out-
routes 192.168.1.0/24
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST I:IBGP L:LOCAL
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      200.1.1.0/24       0.0.0.0         0          101        32768  BL
```

The Adj-RIB-Out contains the routes that the Layer 3 Switch either has most recently sent to the neighbor or is about to send to the neighbor.

Syntax: show ip bgp neighbor <ip-addr> rib-out-routes [<ip-addr>/<prefix>]

For information about the fields in this display, see Table 21.10 on page 21-91. The fields in this display also appear in the **show ip bgp** display.

Displaying Peer Group Information

You can display configuration information for peer groups.

To display peer-group information, enter a command such as the following at the Privileged EXEC level of the CLI:

```
FESX424 Router# show ip bgp peer-group pg1
1   BGP peer-group is pg
    Description: peer group abc
       SendCommunity: yes
       NextHopSelf: yes
       DefaultOriginate: yes
    Members:
       IP Address: 192.168.10.10, AS: 65111
```


Syntax: show ip bgp peer-group [<peer-group-name>]

Only the parameters that have values different from their defaults are listed.

Displaying Summary Route Information

To display summary statistics for all the routes in the Layer 3 Switch's BGP4 route table, enter a command such as the following at any level of the CLI:

```
FESX424 Router(config-bgp-router)# show ip bgp routes summary
  Total number of BGP routes (NLRIs) Installed     : 20
  Distinct BGP destination networks                : 20
  Filtered BGP routes for soft reconfig            : 100178
  Routes originated by this router                 : 2
  Routes selected as BEST routes                   : 19
  BEST routes not installed in IP forwarding table : 1
  Unreachable routes (no IGP route for NEXTHOP)    : 1
  IBGP routes selected as best routes              : 0
  EBGP routes selected as best routes              : 17
```

Syntax: show ip bgp routes summary

This display shows the following information.

Table 21.9: BGP4 Summary Route Information

This Field...	Displays...
Total number of BGP routes (NLRIs) Installed	The number of BGP4 routes the Layer 3 Switch has installed in the BGP4 route table.
Distinct BGP destination networks	The number of destination networks the installed routes represent. The BGP4 route table can have multiple routes to the same network.
Filtered BGP routes for soft reconfig	The number of route updates received from soft-reconfigured neighbors or peer groups that have been filtered out but retained. For information about soft reconfiguration, see "Using Soft Reconfiguration" on page 21-100.
Routes originated by this router	The number of routes in the BGP4 route table that this Layer 3 Switch originated.
Routes selected as BEST routes	The number of routes in the BGP4 route table that this Layer 3 Switch has selected as the best routes to the destinations.
BEST routes not installed in IP forwarding table	The number of BGP4 routes that are the best BGP4 routes to their destinations but were not installed in the IP route table because the Layer 3 Switch received better routes from other sources (such as OSPF, RIP, or static IP routes).
Unreachable routes (no IGP route for NEXTHOP)	The number of routes in the BGP4 route table whose destinations are unreachable because the next hop is unreachable.
IBGP routes selected as best routes	The number of "best" routes in the BGP4 route table that are IBGP routes.
EBGP routes selected as best routes	The number of "best" routes in the BGP4 route table that are EBGP routes.

Displaying the BGP4 Route Table

BGP4 uses filters you define as well as the algorithm described in "How BGP4 Selects a Path for a Route" on page 21-4 to determine the preferred route to a destination. BGP4 sends only the preferred route to the router's IP table. However, if you want to view all the routes BGP4 knows about, you can display the BGP4 table using either of the following methods.

To view the BGP4 route table, enter the following command:

```
FESX424 Router(config-bgp-router)# show ip bgp routes
Total number of BGP Routes: 97371
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      3.0.0.0/8          192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 701 80
2      4.0.0.0/8          192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 1
3      4.60.212.0/22      192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 701 1 189
4      6.0.0.0/8          192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 3356 7170 1455
5      8.8.1.0/24         192.168.4.106   0           100        0      BE
         AS_PATH: 65001
```

Syntax: show ip bgp routes [[network] <ip-addr>] | <num> | [age <secs>] | [as-path-access-list <num>] | [best] | [cidr-only] | [community <num> | no-export | no-advertise | internet | local-as] | [community-access-list <num>] | [community-list <num> | [detail <option>] | [filter-list <num, num,...>] | [next-hop <ip-addr>] | [no-best] | [not-installed-best] | [prefix-list <string>] | [regular-expression <regular-expression>] | [route-map <map-name>] | [summary] | [unreachable]

The <ip-addr> option displays routes for a specific network. The **network** keyword is optional. You can enter the network address without entering "network" in front of it.

The <num> option specifies the table entry with which you want the display to start. For example, if you want to list entries beginning with table entry 100, specify 100.

The **age** <secs> parameter displays only the routes that have been received or updated more recently than the number of seconds you specify.

The **as-path-access-list** <num> parameter filters the display using the specified AS-path ACL.

The **best** parameter displays the routes received from the neighbor that the Layer 3 Switch selected as the best routes to their destinations.

The **cidr-only** option lists only the routes whose network masks do not match their class network length.

The **community** option lets you display routes for a specific community. You can specify **local-as**, **no-export**, **no-advertise**, **internet**, or a private community number. You can specify the community number as either two five-digit integer values of up to 1– 65535, separated by a colon (for example, 12345:6789) or a single long integer value.

The **community-access-list** <num> parameter filters the display using the specified community ACL.

The **community-list** option lets you display routes that match a specific community filter.

The **detail** option lets you display more details about the routes. You can refine your request by also specifying one of the other display options after the **detail** keyword.

The **filter-list** option displays routes that match a specific address filter list.

The **next-hop** <ip-addr> option displays the routes for a given next-hop IP address.

The **no-best** option displays the routes for which none of the routes to a given prefix were selected as the best route. The IP route table does not contain a BGP4 route for any of the routes listed by the command.

The **not-installed-best** option displays the routes received from the neighbor that are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table because the Layer 3 Switch received better routes from other sources (such as OSPF, RIP, or static IP routes).

The **prefix-list** <string> parameter filters the display using the specified IP prefix list.

The **regular-expression** <regular-expression> option filters the display based on a regular expression. See "Using Regular Expressions" on page 21-42.

The **route-map** <map-name> parameter filters the display using the specified route map. The software displays only the routes that match the match statements in the route map. The software disregards the route map's set statements.

The **summary** option displays summary information for the routes.

The **unreachable** option displays the routes that are unreachable because the Layer 3 Switch does not have a valid RIP, OSPF, or static route to the next hop.

Displaying the Best BGP4 Routes

To display all the BGP4 routes in the Layer 3 Switch's BGP4 route table that are the best routes to their destinations, enter a command such as the following at any level of the CLI:

```
FESX424 Router(config-bgp-router)# show ip bgp routes best
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      3.0.0.0/8          192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 701 80
2      4.0.0.0/8          192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 1
3      4.60.212.0/22      192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 701 1 189
4      6.0.0.0/8          192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 3356 7170 1455
5      9.2.0.0/16         192.168.4.106               100        0      BE
         AS_PATH: 65001 4355 701
```

Syntax: show ip bgp routes best

For information about the fields in this display, see Table 21.10 on page 21-91. The fields in this display also appear in the **show ip bgp** display.

Displaying Those Best BGP4 Routes that Are Nonetheless Not in the IP Route Table

When the Layer 3 Switch has multiple routes to a destination from different sources (such as BGP4, OSPF, RIP, or static routes), the Layer 3 Switch selects the route with the lowest administrative distance as the best route, and installs that route in the IP route table.

To display the BGP4 routes are the “best” routes to their destinations but are not installed in the Layer 3 Switch’s IP route table, enter a command such as the following at any level of the CLI:

```
FESX424 Router(config-bgp-router)# show ip bgp routes not-installed-best
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      192.168.4.0/24     192.168.4.106   0          100        0      bE
         AS_PATH: 65001
```

Each of the displayed routes is a valid path to its destination, but the Layer 3 Switch received another path from a different source (such as OSPF, RIP, or a static route) that has a lower administrative distance. The Layer 3 Switch always selects the path with the lowest administrative distance to install in the IP route table.

Notice that the route status in this example is the new status, “b”. See Table 21.10 on page 21-91 for a description.

Syntax: show ip bgp routes not-installed-best

For information about the fields in this display, see Table 21.10 on page 21-91. The fields in this display also appear in the **show ip bgp** display.

NOTE: To display the routes that the Layer 3 Switch has selected as the best routes and installed in the IP route table, display the IP route table using the **show ip route** command.

Displaying BGP4 Routes Whose Destinations Are Unreachable

To display BGP4 routes whose destinations are unreachable using any of the BGP4 paths in the BGP4 route table, enter a command such as the following at any level of the CLI:

```
FESX424 Router(config-bgp-router)# show ip bgp routes unreachable
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
       Prefix      Next Hop        Metric     LocPrf     Weight Status
1      8.8.8.0/24  192.168.5.1     0          101        0
         AS_PATH: 65001 4355 1
```

Syntax: show ip bgp routes unreachable

For information about the fields in this display, see Table 21.10 on page 21-91. The fields in this display also appear in the **show ip bgp** display.

Displaying Information for a Specific Route

To display BGP4 network information by specifying an IP address within the network, enter a command such as the following at any level of the CLI:

```
FastIron SuperX Router(config-bgp-router)# show ip bgp 9.3.4.0
Number of BGP Routes matching display condition : 1
Status codes: s suppressed, d damped, h history, * valid, > best, i internal
Origin codes: i - IGP, e - EGP, ? - incomplete
    Network            Next Hop        Metric LocPrf Weight Path
*>  9.3.4.0/24         192.168.4.106          100    0      65001 4355 1 1221 ?
       Last update to IP routing table: 0h11m38s, 1 path(s) installed:
         Gateway           Port
         192.168.2.1       2/1
       Route is advertised to 1 peers:
        20.20.20.2(65300)
```

Syntax: show ip bgp [route] <ip-addr>/<prefix> [longer-prefixes] | <ip-addr>

If you use the **route** option, the display for the information is different, as shown in the following example:

```
FastIron SuperX Router(config-bgp-router)# show ip bgp route 9.3.4.0
Number of BGP Routes matching display condition : 1
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      9.3.4.0/24         192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 1 1221
       Last update to IP routing table: 0h12m1s, 1 path(s) installed:
         Gateway           Port
         192.168.2.1       2/1
       Route is advertised to 1 peers:
        20.20.20.2(65300)
```

These displays show the following information.

Table 21.10: BGP4 Network Information

This Field...	Displays...
Number of BGP Routes matching display condition	The number of routes that matched the display parameters you entered. This is the number of routes displayed by the command.
Status codes	A list of the characters the display uses to indicate the route's status. The status code appears in the left column of the display, to the left of each route. The status codes are described in the command's output. **Note**: This field appears only if you *do not* enter the **route** option.
Prefix	The network address and prefix.
Next Hop	The next-hop router for reaching the network from the Layer 3 Switch.
Metric	The value of the route's MED attribute. If the route does not have a metric, this field is blank.

Table 21.10: BGP4 Network Information (Continued)

This Field...	Displays...
LocPrf	The degree of preference for this route relative to other routes in the local AS. When the BGP4 algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 – 4294967295.
Weight	The value that this router associates with routes from a specific neighbor. For example, if the router receives routes to the same destination from two BGP4 neighbors, the router prefers the route from the neighbor with the larger weight.
Path	The route's AS path. **Note**: This field appears only if you *do not* enter the **route** option.
Origin code	A character the display uses to indicate the route's origin. The origin code appears to the right of the AS path (Path field). The origin codes are described in the command's output. **Note**: This field appears only if you *do not* enter the **route** option.

Table 21.10: BGP4 Network Information (Continued)

This Field...	Displays...
Status	The route's status, which can be one or more of the following: • A – AGGREGATE. The route is an aggregate route for multiple networks. • B – BEST. BGP4 has determined that this is the optimal route to the destination. **Note**: If the "b" is shown in lowercase, the software was not able to install the route in the IP route table. • b – NOT-INSTALLED-BEST. The routes received from the neighbor are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table because the Layer 3 Switch received better routes from other sources (such as OSPF, RIP, or static IP routes). • C – CONFED_EBGP. The route was learned from a neighbor in the same confederation and AS, but in a different sub-AS within the confederation. • D – DAMPED. This route has been dampened (by the route dampening feature), and is currently unusable. • H – HISTORY. Route dampening is configured for this route, and the route has a history of flapping and is unreachable now. • I – INTERNAL. The route was learned through BGP4. • L – LOCAL. The route originated on this Layer 3 Switch. • M – MULTIPATH. BGP4 load sharing is enabled and this route was selected as one of the best ones to the destination. The best route among the multiple paths also is marked with "B". **Note**: If the "m" is shown in lowercase, the software was not able to install the route in the IP route table. • S – SUPPRESSED. This route was suppressed during aggregation and thus is not advertised to neighbors. **Note**: This field appears only if you enter the **route** option.

Displaying Route Details

Here is an example of the information displayed when you use the **detail** option. In this example, the information for one route is shown.

```
FESX424 Router# show ip bgp routes detail
Total number of BGP Routes: 2
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
1       Prefix: 10.5.0.0/24,  Status: BME,  Age: 0h28m28s
         NEXT_HOP: 201.1.1.2,  Learned from Peer: 10.1.0.2 (5)
          LOCAL_PREF: 101,  MED: 0,  ORIGIN: igp,  Weight: 10
           AS_PATH: 5
            Adj_RIB_out count: 4,  Admin distance 20
```

These displays show the following information.

Table 21.11: BGP4 Route Information

This Field...	Displays...
Total number of BGP Routes	The number of BGP4 routes.
Status codes	A list of the characters the display uses to indicate the route's status. The status code is appears in the left column of the display, to the left of each route. The status codes are described in the command's output.
Prefix	The network prefix and mask length.
Status	The route's status, which can be one or more of the following: • A – AGGREGATE. The route is an aggregate route for multiple networks. • B – BEST. BGP4 has determined that this is the optimal route to the destination. **Note**: If the "b" is shown in lowercase, the software was not able to install the route in the IP route table. • b – NOT-INSTALLED-BEST. The routes received from the neighbor are the best BGP4 routes to their destinations, but were nonetheless not installed in the IP route table because the Layer 3 Switch received better routes from other sources (such as OSPF, RIP, or static IP routes). • C – CONFED_EBGP. The route was learned from a neighbor in the same confederation and AS, but in a different sub-AS within the confederation. • D – DAMPED. This route has been dampened (by the route dampening feature), and is currently unusable. • H – HISTORY. Route dampening is configured for this route, and the route has a history of flapping and is unreachable now. • I – INTERNAL. The route was learned through BGP4. • L – LOCAL. The route originated on this Layer 3 Switch. • M – MULTIPATH. BGP4 load sharing is enabled and this route was selected as one of the best ones to the destination. The best route among the multiple paths also is marked with "B". **Note**: If the "m" is shown in lowercase, the software was not able to install the route in the IP route table. • S – SUPPRESSED. This route was suppressed during aggregation and thus is not advertised to neighbors.
Age	The last time an update occurred.
Next_Hop	The next-hop router for reaching the network from the Layer 3 Switch.
Learned from Peer	The IP address of the neighbor that sent this route.

Table 21.11: BGP4 Route Information (Continued)

This Field...	Displays...
Local_Pref	The degree of preference for this route relative to other routes in the local AS. When the BGP4 algorithm compares routes on the basis of local preferences, the route with the higher local preference is chosen. The preference can have a value from 0 – 4294967295.
MED	The route's metric. If the route does not have a metric, this field is blank.
Origin	The source of the route information. The origin can be one of the following: • EGP – The routes with this set of attributes came to BGP through EGP. • IGP – The routes with this set of attributes came to BGP through IGP. • INCOMPLETE – The routes came from an origin other than one of the above. For example, they may have been redistributed from OSPF or RIP. When BGP4 compares multiple routes to a destination to select the best route, IGP is preferred over EGP and both are preferred over INCOMPLETE.
Weight	The value that this router associates with routes from a specific neighbor. For example, if the router receives routes to the same destination from two BGP4 neighbors, the router prefers the route from the neighbor with the larger weight.
Atomic	Whether network information in this route has been aggregated *and* this aggregation has resulted in information loss. **Note**: Information loss under these circumstances is a normal part of BGP4 and does not indicate an error.
Aggregation ID	The router that originated this aggregator.
Aggregation AS	The AS in which the network information was aggregated. This value applies only to aggregated routes and is otherwise 0.
Originator	The originator of the route in a route reflector environment.
Cluster List	The route-reflector clusters through which this route has passed.
Learned From	The IP address of the neighbor from which the Layer 3 Switch learned the route.
Admin Distance	The administrative distance of the route.
Adj_RIB_out	The number of neighbors to which the route has been or will be advertised. This is the number of times the route has been selected as the best route and placed in the Adj-RIB-Out (outbound queue) for a BGP4 neighbor.
Communities	The communities the route is in.

Displaying BGP4 Route-Attribute Entries

The route-attribute entries table lists the sets of BGP4 attributes stored in the router's memory. Each set of attributes is unique and can be associated with one or more routes. In fact, the router typically has fewer route attribute entries than routes. To display the route-attribute entries table, use one of the following methods.

To display the IP route table, enter the following command:

```
FESX424 Router# show ip bgp attribute-entries
```

Syntax: show ip bgp attribute-entries

Here is an example of the information displayed by this command. A zero value indicates that the attribute is not set.

```
FESX424 Router# show ip bgp attribute-entries
        Total number of BGP Attribute Entries: 7753
1       Next Hop  :192.168.11.1       Metric   :0                Origin:IGP
        Originator:0.0.0.0            Cluster List:None
         Aggregator:AS Number :0         Router-ID:0.0.0.0          Atomic:FALSE
        Local Pref:100                Communities:Internet
        AS Path   :(65002) 65001 4355 2548 3561 5400 6669 5548
2       Next Hop  :192.168.11.1       Metric   :0                Origin:IGP
        Originator:0.0.0.0            Cluster List:None
         Aggregator:AS Number :0         Router-ID:0.0.0.0          Atomic:FALSE
        Local Pref:100                Communities:Internet
        AS Path   :(65002) 65001 4355 2548
```

This display shows the following information.

Table 21.12: BGP4 Route-Attribute Entries Information

This Field...	Displays...
Total number of BGP Attribute Entries	The number of routes contained in this router's BGP4 route table.
Next Hop	The IP address of the next hop router for routes that have this set of attributes.
Metric	The cost of the routes that have this set of attributes.
Origin	The source of the route information. The origin can be one of the following: • EGP – The routes with this set of attributes came to BGP through EGP. • IGP – The routes with this set of attributes came to BGP through IGP. • INCOMPLETE – The routes came from an origin other than one of the above. For example, they may have been redistributed from OSPF or RIP. When BGP4 compares multiple routes to a destination to select the best route, IGP is preferred over EGP and both are preferred over INCOMPLETE.
Originator	The originator of the route in a route reflector environment.

Table 21.12: BGP4 Route-Attribute Entries Information (Continued)

This Field...	Displays...
Cluster List	The route-reflector clusters through which this set of attributes has passed.
Aggregator	Aggregator information: • `AS Number` shows the AS in which the network information in the attribute set was aggregated. This value applies only to aggregated routes and is otherwise 0. • `Router-ID` shows the router that originated this aggregator.
Atomic	Whether the network information in this set of attributes has been aggregated *and* this aggregation has resulted in information loss. • TRUE – Indicates information loss has occurred • FALSE – Indicates no information loss has occurred **Note**: Information loss under these circumstances is a normal part of BGP4 and does not indicate an error.
Local Pref	The degree of preference for routes that use this set of attributes relative to other routes in the local AS.
Communities	The communities that routes with this set of attributes are in.
AS Path	The ASs through which routes with this set of attributes have passed. The local AS is shown in parentheses.

Displaying the Routes BGP4 Has Placed in the IP Route Table

The IP route table indicates the routes it has received from BGP4 by listing “BGP” as the route type.

To display the IP route table, enter the following command:

```
FESX424 Router# show ip route
```

Syntax: show ip route [<ip-addr> | <num> | bgp | ospf | rip]

Here is an example of the information displayed by this command. Notice that most of the routes in this example have type “B”, indicating that their source is BGP4.

```
FESX424 Router# show ip route

Total number of IP routes: 50834

B:BGP D:Directly-Connected  O:OSPF  R:RIP  S:Static

      Network Address  NetMask          Gateway          Port      Cost    Type
      3.0.0.0          255.0.0.0        192.168.13.2     1/1       0       B
      4.0.0.0          255.0.0.0        192.168.13.2     1/1       0       B
      9.20.0.0         255.255.128.0    192.168.13.2     1/1       0       B
      10.1.0.0         255.255.0.0      0.0.0.0          1/1       1       D
      10.10.11.0       255.255.255.0    0.0.0.0          2/24      1       D
      12.2.97.0        255.255.255.0    192.168.13.2     1/1       0       B
      12.3.63.0        255.255.255.0    192.168.13.2     1/1       0       B
      12.3.123.0       255.255.255.0    192.168.13.2     1/1       0       B
      12.5.252.0       255.255.254.0    192.168.13.2     1/1       0       B
      12.6.42.0        255.255.254.0    192.168.13.2     1/1       0       B
remaining 50824 entries not shown...
```

Displaying Route Flap Dampening Statistics

To display route dampening statistics or all the dampened routes, enter the following command at any level of the CLI:

```
FESX424 Router# show ip bgp flap-statistics

Total number of flapping routes: 414
    Status Code  >:best d:damped h:history *:valid
    Network            From            Flaps Since    Reuse    Path
h>  192.50.206.0/23    166.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 701
h>  203.255.192.0/20   166.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 7018
h>  203.252.165.0/24   166.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 7018
h>  192.50.208.0/23    166.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 701
h>  133.33.0.0/16      166.90.213.77   1     0 :0 :13 0 :0 :0  65001 4355 1 701
*>  204.17.220.0/24    166.90.213.77   1     0 :1 :4  0 :0 :0  65001 4355 701 62
```

Syntax: show ip bgp flap-statistics [regular-expression <regular-expression> | <address> <mask> [longer-prefixes] | neighbor <ip-addr> | filter-list <num>...]

The **regular-expression** <regular-expression> parameter is a regular expression. The regular expressions are the same ones supported for BGP4 AS-path filters. See “Using Regular Expressions” on page 21-42.

The <address> <mask> parameter specifies a particular route. If you also use the optional **longer-prefixes** parameter, then all statistics for routes that match the specified route or have a longer prefix than the specified route are displayed. For example, if you specify **209.157.0.0 longer**, then all routes with the prefix 209.157 or that have a longer prefix (such as 209.157.22) are displayed.

The **neighbor** <ip-addr> parameter displays route flap dampening statistics only for routes learned from the specified neighbor. You also can display route flap statistics for routes learned from a neighbor by entering the following command: **show ip bgp neighbor** <ip-addr> **flap-statistics**.

The **filter-list** <num> parameter specifies one or more filters. Only the routes that have been dampened and that match the specified filter(s) are displayed.

This display shows the following information.

Table 21.13: Route Flap Dampening Statistics

This Field...	Displays...
Total number of flapping routes	The total number of routes in the Layer 3 Switch's BGP4 route table that have changed state and thus have been marked as flapping routes.
Status code	Indicates the dampening status of the route, which can be one of the following: • > – This is the best route among those in the BGP4 route table to the route's destination. • d – This route is currently dampened, and thus unusable. • h – The route has a history of flapping and is unreachable now. • * – The route has a history of flapping but is currently usable.
Network	The destination network of the route.
From	The neighbor that sent the route to the Layer 3 Switch.
Flaps	The number of flaps (state changes) the route has experienced.
Since	The amount of time since the first flap of this route.
Reuse	The amount of time remaining until this route will be un-suppressed and thus be usable again.
Path	Shows the AS-path information for the route.

You also can display all the dampened routes by entering the following command:
show ip bgp dampened-paths.

Displaying the Active Route Map Configuration

To view the device's active route map configuration (contained in the running-config) without displaying the entire running-config, enter the following command at any level of the CLI:

```
FESX424 Router# show route-map
route-map permitnet4 permit 10
 match ip address prefix-list plist1
route-map permitnet1 permit 1
 match ip address prefix-list plist2
route-map setcomm permit 1
 set community 1234:2345 no-export
route-map test111 permit 111
 match address-filters 11
 set community 11:12 no-export
route-map permit1122 permit 12
 match ip address 11
route-map permit1122 permit 13
 match ip address std_22
```

This example shows that the running-config contains six route maps. Notice that the match and set statements within each route map are listed beneath the command for the route map itself. In this simplified example, each route map contains only one match or set statement.

To display the active configuration for a specific route map, enter a command such as the following, which specifies a route map name:

```
FESX424 Router# show route-map setcomm
route-map setcomm permit 1
 set community 1234:2345 no-export
```

This example shows the active configuration for a route map called “setcomm“.

Syntax: show route-map [<map-name>]

Updating Route Information and Resetting a Neighbor Session

The following sections describe ways to update route information with a neighbor, reset the session with a neighbor, and close a session with a neighbor.

Whenever you change a policy (ACL, route map, and so on) that affects the routes that the Layer 3 Switch learns from a BGP4 neighbor or peer group of neighbors, you must enter a command to place the changes into effect. The changes take place automatically, but only affect new route updates. To make changes retroactive for routes received or sent before the changes were made, you need to enter a clear command.

You can update the learned routes using either of the following methods:

- Request the complete BGP4 route table from the neighbor or peer group. You can use this method if the neighbor supports the refresh capability (RFCs 2842 and 2858).
- Clear (reset) the session with the neighbor or peer group. This is the only method you can use if the neighbor does not support the refresh capability.

Each of these methods is effective, but can be disruptive to the network. The first method adds overhead while the Layer 3 Switch learns and filters the neighbor’s or group’s entire route table, while the second method adds more overhead while the devices re-establish their BGP4 sessions.

You also can clear and reset the BGP4 routes that have been installed in the IP route table. See “Clearing and Resetting BGP4 Routes in the IP Route Table” on page 21-106.

Using Soft Reconfiguration

The ***soft reconfiguration*** feature places policy changes into effect without resetting the BGP4 session. Soft reconfiguration does not request the neighbor or group to send its entire BGP4 table, nor does the feature reset the session with the neighbor or group. Instead, the soft reconfiguration feature stores all the route updates received from the neighbor or group. When you request a soft reset of inbound routes, the software performs route selection by comparing the policies against the stored route updates, instead of requesting the neighbor’s BGP4 route table or resetting the session with the neighbor.

When you enable the soft reconfiguration feature, it sends a refresh message to the neighbor or group if the neighbor or group supports dynamic refresh. Otherwise, the feature resets the neighbor session. This step is required to ensure that the soft reconfiguration feature has a complete set of updates to use, and occurs only once, when you enable the feature. The feature accumulates all the route updates from the neighbor, eliminating the need for additional refreshes or resets when you change policies in the future.

To use soft reconfiguration:

- Enable the feature.
- Make the policy changes.
- Apply the changes by requesting a soft reset of the inbound updates from the neighbor or group.

Use the following CLI methods to configure soft configuration, apply policy changes, and display information for the updates that are filtered out by the policies.

Enabling Soft Reconfiguration

To configure a neighbor for soft reconfiguration, enter a command such as the following:

```
FESX424 Router(config-bgp-router)# neighbor 10.10.200.102 soft-reconfiguration
inbound
```

This command enables soft reconfiguration for updates received from 10.10.200.102. The software dynamically refreshes or resets the session with the neighbor, then retains all route updates from the neighbor following the reset.

Syntax: [no] neighbor <ip-addr> | <peer-group-name> soft-reconfiguration inbound

NOTE: The syntax related to soft reconfiguration is shown. For complete command syntax, see "Adding BGP4 Neighbors" on page 21-12.

Placing a Policy Change into Effect

To place policy changes into effect, enter a command such as the following:

```
FESX424 Router(config-bgp-router)# clear ip bgp neighbor 10.10.200.102 soft in
```

This command updates the routes by comparing the route policies against the route updates that the Layer 3 Switch has stored. The command does not request additional updates from the neighbor or otherwise affect the session with the neighbor.

Syntax: clear ip bgp neighbor <ip-addr> | <peer-group-name> soft in

NOTE: If you do not specify "in", the command applies to both inbound and outbound updates.

NOTE: The syntax related to soft reconfiguration is shown. For complete command syntax, see "Dynamically Refreshing Routes" on page 21-103.

Displaying the Filtered Routes Received from the Neighbor or Peer Group

When you enable soft reconfiguration, the Layer 3 Switch saves all updates received from the specified neighbor or peer group. This includes updates that contain routes that are filtered out by the BGP4 route policies in effect on the Layer 3 Switch. To display the routes that have been filtered out, enter the following command at any level of the CLI:

```
FESX424 Router# show ip bgp filtered-routes
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      3.0.0.0/8          192.168.4.106              100        0      EF
         AS_PATH: 65001 4355 701 80
2      4.0.0.0/8          192.168.4.106              100        0      EF
         AS_PATH: 65001 4355 1
3      4.60.212.0/22      192.168.4.106              100        0      EF
         AS_PATH: 65001 4355 701 1 189
```

The routes displayed by the command are the routes that the Layer 3 Switch's BGP4 policies filtered out. The Layer 3 Switch did not place the routes in the BGP4 route table, but did keep the updates. If a policy change causes these routes to be permitted, the Layer 3 Switch does not need to request the route information from the neighbor, but instead uses the information in the updates.

Syntax: show ip bgp filtered-routes [<ip-addr>] | [as-path-access-list <num>] | [detail] | [prefix-list <string>]

The <ip-addr> parameter specifies the IP address of the destination network.

The **as-path-access-list** <num> parameter specifies an AS-path ACL. Only the routes permitted by the AS-path ACL are displayed.

The **detail** parameter displays detailed information for the routes. (The example above shows summary information.) You can specify any of the other options after **detail** to further refine the display request.

The prefix-list <string> parameter specifies an IP prefix list. Only the routes permitted by the prefix list are displayed.

NOTE: The syntax for displaying filtered routes is shown. For complete command syntax, see "Displaying the BGP4 Route Table" on page 21-88.

Displaying All the Routes Received from the Neighbor

To display all the route information received in route updates from a neighbor since you enabled soft reconfiguration, enter a command such as the following at any level of the CLI:

```
FESX424 Router# show ip bgp neighbor 192.168.4.106 received-routes
       There are 97345 received routes from  neighbor 192.168.4.106
Searching for matching routes, use ^C to quit...
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST C:CONFED_EBGP D:DAMPED
       E:EBGP H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED F:FILTERED
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      3.0.0.0/8          192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 701 80
2      4.0.0.0/8          192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 1
3      4.60.212.0/22      192.168.4.106              100        0      BE
         AS_PATH: 65001 4355 701 1 189
4      6.0.0.0/8          192.168.4.106              100        0      BE
```

Syntax: show ip bgp neighbors <ip-addr> received-routes [detail]

The **detail** parameter displays detailed information for the routes. The example above shows summary information.

NOTE: The syntax for displaying received routes is shown. For complete command syntax, see "Displaying BGP4 Neighbor Information" on page 21-73.

NOTE: The **show ip bgp neighbor** <ip-addr> **received-routes** syntax supported in previous software releases is changed to the following syntax: **show ip bgp neighbor** <ip-addr> **routes**.

Dynamically Requesting a Route Refresh from a BGP4 Neighbor

You can easily apply changes to filters that control BGP4 routes received from or advertised to a neighbor, without resetting the BGP4 session between the Layer 3 Switch and the neighbor. For example, if you add, change, or remove a BGP4 address filter that denies specific routes received from a neighbor, you can apply the filter change by requesting a route refresh from the neighbor. If the neighbor also supports dynamic route refreshes, the neighbor resends its Adj-RIB-Out, its table of BGP4 routes. Using the route refresh feature, you do not need to reset the session with the neighbor.

The route refresh feature is based on the following specifications:

- RFC 2842. This RFC specifies the Capability Advertisement, which a BGP4 router uses to dynamically negotiate a capability with a neighbor.
- RFC 2858 for Multi-protocol Extension.

NOTE: The Foundry implementation of dynamic route refresh supports negotiation of IP version 4 unicasts only.

- RFC 2918, which describes the dynamic route refresh capability

The dynamic route refresh capability is enabled by default and cannot be disabled. When the Layer 3 Switch sends a BGP4 OPEN message to a neighbor, the Layer 3 Switch includes a Capability Advertisement to inform the neighbor that the Layer 3 Switch supports dynamic route refresh.

NOTE: The option for dynamically refreshing routes received from a neighbor requires the neighbor to support dynamic route refresh. If the neighbor does not support this feature, the option does not take effect and the software displays an error message. The option for dynamically re-advertising routes to a neighbor does not require the neighbor to support dynamic route refresh.

To use the dynamic refresh feature, use either of the following methods.

Dynamically Refreshing Routes

The following sections describe how to dynamically refresh BGP4 routes to place new or changed filters into effect.

To request a dynamic refresh of all routes from a neighbor, enter a command such as the following:

```
FESX424 Router(config-bgp-router)# clear ip bgp neighbor 192.168.1.170 soft in
```

This command asks the neighbor to send its BGP4 table (Adj-RIB-Out) again. The Layer 3 Switch applies its filters to the incoming routes and adds, modifies, or removes BGP4 routes as necessary.

Syntax: clear ip bgp neighbor all | <ip-addr> | <peer-group-name> | <as-num> [soft-outbound | soft [in | out]]

The **all** | <ip-addr> | <peer-group-name> | <as-num> specifies the neighbor. The <ip-addr> parameter specifies a neighbor by its IP interface with the Layer 3 Switch. The <peer-group-name> specifies all neighbors in a specific peer group. The <as-num> parameter specifies all neighbors within the specified AS. The **all** parameter specifies all neighbors.

The **soft-outbound** parameter updates all outbound routes by applying the new or changed filters, but sends only the existing routes affected by the new or changed filters to the neighbor.

The **soft [in | out]** parameter specifies whether you want to refresh the routes received from the neighbor or sent to the neighbor:

- **soft in** does one of the following:
  - If you enabled soft reconfiguration for the neighbor or peer group, **soft in** updates the routes by comparing the route policies against the route updates that the Layer 3 Switch has stored. Soft reconfiguration does not request additional updates from the neighbor or otherwise affect the session with the neighbor. See “Using Soft Reconfiguration” on page 21-100.
  - If you did not enable soft reconfiguration, **soft in** requests the neighbor’s entire BGP4 route table (Adj-RIB-Out), then applies the filters to add, change, or exclude routes.
  - If a neighbor does not support dynamic refresh, **soft in** resets the neighbor session.
- **soft out** updates all outbound routes, then sends the Layer 3 Switch’s entire BGP4 route table (Adj-RIB-Out) to the neighbor, after changing or excluding the routes affected by the filters.

If you do not specify **in** or **out**, the Layer 3 Switch performs both options.

NOTE: The **soft-outbound** parameter updates all outbound routes by applying the new or changed filters, but sends only the existing routes affected by the new or changed filters to the neighbor. The **soft out** parameter updates all outbound routes, then sends the Layer 3 Switch's entire BGP4 route table (Adj-RIB-Out) to the neighbor, after changing or excluding the routes affected by the filters. Use **soft-outbound** if only the outbound policy is changed.

To dynamically resend all the Layer 3 Switch's BGP4 routes to a neighbor, enter a command such as the following:

```
FESX424 Router(config-bgp-router)# clear ip bgp neighbor 192.168.1.170 soft out
```

This command applies its filters for outgoing routes to the Layer 3 Switch's BGP4 route table (Adj-RIB-Out), changes or excludes routes accordingly, then sends the resulting Adj-RIB-Out to the neighbor.

NOTE: The Foundry Layer 3 Switch does not automatically update outbound routes using a new or changed outbound policy or filter when a session with the neighbor goes up or down. Instead, the Layer 3 Switch applies a new or changed policy or filter when a route is placed in the outbound queue (Adj-RIB-Out).

To place a new or changed outbound policy or filter into effect, you must enter a **clear ip bgp neighbor** command regardless of whether the neighbor session is up or down. You can enter the command without optional parameters or with the **soft out** or **soft-outbound** option. Either way, you must specify a parameter for the neighbor (<ip-addr>, <as-num>, <peer-group-name>, or **all**).

Displaying Dynamic Refresh Information

You can use the **show ip bgp neighbors** command to display information for dynamic refresh requests. For each neighbor, the display lists the number of dynamic refresh requests the Layer 3 Switch has sent to or received from the neighbor and indicates whether the Layer 3 Switch received confirmation from the neighbor that the neighbor supports dynamic route refresh.

The RefreshCapability field indicates whether this Layer 3 Switch has received confirmation from the neighbor that the neighbor supports the dynamic refresh capability. The statistics in the Message Sent and Message Received

rows under Refresh-Req indicate how many dynamic refreshes have been sent to and received from the neighbor. The statistic is cumulative across sessions.

```
FESX424 Router(config-bgp-router)# show ip bgp neighbor 10.4.0.2
1   IP Address: 10.4.0.2, AS: 5 (EBGP), RouterID: 100.0.0.1
      Description: neighbor 10.4.0.2
    State: ESTABLISHED, Time: 0h1m0s, KeepAliveTime: 0, HoldTime: 0
      PeerGroup: pg1
      Mutihop-EBGP: yes, ttl: 1
      RouteReflectorClient: yes
      SendCommunity: yes
      NextHopSelf: yes
      DefaultOriginate: yes (default sent)
      MaximumPrefixLimit: 90000
      RemovePrivateAs: : yes
      RefreshCapability: Received
    Route Filter Policies:
      Distribute-list: (out) 20
      Filter-list: (in) 30
      Prefix-list: (in) pf1
      Route-map: (in) setnp1  (out) setnp2
    Messages:    Open    Update  KeepAlive Notification Refresh-Req
      Sent    : 1       1       1         0            0
      Received: 1       8       1         0            0
    Last Update Time: NLRI       Withdraw          NLRI       Withdraw
                 Tx: 0h0m59s     ---          Rx: 0h0m59s    ---
    Last Connection Reset Reason:Unknown
      Notification Sent:     Unspecified
      Notification Received: Unspecified
    TCP Connection state: ESTABLISHED
      Byte Sent:   115, Received: 492
      Local host:  10.4.0.1, Local  Port: 179
      Remote host: 10.4.0.2, Remote Port: 8053
      ISentSeq:   52837276  SendNext:   52837392  TotUnAck:          0
      TotSent:         116  ReTrans:           0  UnAckSeq:   52837392
      IRcvSeq:  2155052043  RcvNext:  2155052536  SendWnd:       16384
      TotalRcv:        493  DupliRcv:          0  RcvWnd:        16384
      SendQue:           0  RcvQue:            0  CngstWnd:       1460
```

Closing or Resetting a Neighbor Session

You can close a neighbor session or resend route updates to a neighbor.

If you make changes to filters or route maps and the neighbor does not support dynamic route refresh, use these methods to ensure that neighbors contain only the routes you want them to contain.

- If you close a neighbor session, the Layer 3 Switch and the neighbor clear all the routes they learned from each other. When the Layer 3 Switch and neighbor establish a new BGP4 session, they exchange route tables again. Use this method if you want the Layer 3 Switch to relearn routes from the neighbor and resend its own route table to the neighbor.
- If you use the soft-outbound option, the Layer 3 Switch compiles a list of all the routes it would normally send to the neighbor at the beginning of a session. However, before sending the updates, the Foundry Layer 3 Switch also applies the filters and route maps you have configured to the list of routes. If the filters or route maps result in changes to the list of routes, the Layer 3 Switch sends updates to advertise, change, or even withdraw routes on the neighbor as needed. This ensures that the neighbor receives only the routes you want it to contain. Even if the neighbor already contains a route learned from the Layer 3 Switch that you later decided to filter out, using the soft-outbound option removes that route from the neighbor.

You can specify a single neighbor or a peer group.

To close a neighbor session and thus flush all the routes exchanged by the Layer 3 Switch and the neighbor, enter the following command:

```
FESX424 Router# clear ip bgp neighbor all
```

Syntax: clear ip bgp neighbor all | <ip-addr> | <peer-group-name> | <as-num> [soft-outbound | soft [in | out]]

The **all** | <ip-addr> | <peer-group-name> | <as-num> specifies the neighbor. The <ip-addr> parameter specifies a neighbor by its IP interface with the Layer 3 Switch. The <peer-group-name> specifies all neighbors in a specific peer group. The <as-num> parameter specifies all neighbors within the specified AS. The **all** parameter specifies all neighbors.

To resend routes to a neighbor without closing the neighbor session, enter a command such as the following:

```
FESX424 Router# clear ip bgp neighbor 10.0.0.1 soft out
```

Clearing and Resetting BGP4 Routes in the IP Route Table

To clear BGP4 routes from the IP route table and reset the routes, enter a command such as the following:

```
FESX424 Router# clear ip bgp routes
```

Syntax: clear ip bgp routes [<ip-addr>/<prefix-length>]

NOTE: The **clear ip bgp routes** command has the same effect as the **clear ip route** command, but applies only to routes that come from BGP4.

Clearing Traffic Counters

You can clear the counters (reset them to 0) for BGP4 messages. To do so, use one of the following methods.

To clear the BGP4 message counter for all neighbors, enter the following command:

```
FESX424 Router# clear ip bgp traffic
```

Syntax: clear ip bgp traffic

To clear the BGP4 message counter for a specific neighbor, enter a command such as the following:

```
FESX424 Router# clear ip bgp neighbor 10.0.0.1 traffic
```

To clear the BGP4 message counter for all neighbors within a peer group, enter a command such as the following:

```
FESX424 Router# clear ip bgp neighbor PeerGroup1 traffic
```

Syntax: clear ip bgp neighbor all | <ip-addr> | <peer-group-name> | <as-num> traffic

The **all** | <ip-addr> | <peer-group-name> | <as-num> specifies the neighbor. The <ip-addr> parameter specifies a neighbor by its IP interface with the Layer 3 Switch. The <peer-group-name> specifies all neighbors in a specific peer group. The <as-num> parameter specifies all neighbors within the specified AS. The **all** parameter specifies all neighbors.

Clearing Route Flap Dampening Statistics

To clear route flap dampening statistics, use the following CLI method.

NOTE: Clearing the dampening statistics for a route does not change the dampening status of the route.

To clear all the route dampening statistics, enter the following command at any level of the CLI:

```
FESX424 Router# clear ip bgp flap-statistics
```


Syntax: clear ip bgp flap-statistics [regular-expression <regular-expression> | <address> <mask> | neighbor <ip-addr>]

The parameters are the same as those for the **show ip bgp flap-statistics** command (except the **longer-prefixes** option is not supported). See "Displaying Route Flap Dampening Statistics" on page 21-63.

NOTE: The **clear ip bgp damping** command not only clears statistics but also un-suppresses the routes. See "Displaying Route Flap Dampening Statistics" on page 21-63.

Removing Route Flap Dampening

You can un-suppress routes by removing route flap dampening from the routes. The Layer 3 Switch allows you to un-suppress all routes at once or un-suppress individual routes.

To un-suppress all the suppressed routes, enter the following command at the Privileged EXEC level of the CLI:

```
FESX424 Router# clear ip bgp damping
```

Syntax: clear ip bgp damping [<ip-addr> <ip-mask>]

The <ip-addr> parameter specifies a particular network.

The <ip-mask> parameter specifies the network's mask.

To un-suppress a specific route, enter a command such as the following:

```
FESX424 Router# clear ip bgp damping 209.157.22.0 255.255.255.0
```

This command un-suppresses only the route(s) for network 209.157.22.0/24.

Clearing Diagnostic Buffers

The Layer 3 Switch stores the following BGP4 diagnostic information in buffers:

- The first 400 bytes of the last packet that contained an error
- The last NOTIFICATION message either sent or received by the Layer 3 Switch

To display these buffers, use options with the **show ip bgp neighbors** command. See "Displaying BGP4 Neighbor Information" on page 21-73.

This information can be useful if you are working with Foundry Technical Support to resolve a problem. The buffers do not identify the system time when the data was written to the buffer. If you want to ensure that diagnostic data in a buffer is recent, you can clear the buffers. You can clear the buffers for a specific neighbor or for all neighbors.

If you clear the buffer containing the first 400 bytes of the last packet that contained errors, all the bytes are changed to zeros. The Last Connection Reset Reason field of the BGP neighbor table also is cleared.

If you clear the buffer containing the last NOTIFICATION message sent or received, the buffer contains no data.

You can clear the buffers for all neighbors, for an individual neighbor, or for all the neighbors within a specific peer group.

To clear these buffers for neighbor 10.0.0.1, enter the following commands:

```
FESX424 Router# clear ip bgp neighbor 10.0.0.1 last-packet-with-error
FESX424 Router# clear ip bgp neighbor 10.0.0.1 notification-errors
```

Syntax: clear ip bgp neighbor all | <ip-addr> | <peer-group-name> | <as-num>
last-packet-with-error | notification-errors

The **all** | <ip-addr> | <peer-group-name> | <as-num> specifies the neighbor. The <ip-addr> parameter specifies a neighbor by its IP interface with the Layer 3 Switch. The <peer-group-name> specifies all neighbors in a specific

peer group. The <as-num> parameter specifies all neighbors within the specified AS. The **all** parameter specifies all neighbors.

Chapter 22
Configuring VRRP and VRRPE

This chapter describes how to configure Foundry Layer 3 Switches with the following router redundancy protocols:

- ***Virtual Router Redundancy Protocol (VRRP)*** – The standard router redundancy protocol described in RFC 2338.
- ***VRRP Extended (VRRPE)*** – An enhanced version of VRRP that overcomes limitations in the standard protocol.

This chapter contains the following information:

Table 22.1: Chapter Contents

NOTE: VRRP and VRRPE are separate protocols. You cannot use them together.

NOTE: You can use a Foundry Layer 3 Switch configured for VRRP with another Foundry Layer 3 Switch or a third-party router that is also configured for VRRP. However, you can use a Foundry Layer 3 Switch configured for VRRPE only with another Foundry Layer 3 Switch that also is configured for VRRPE.

For a summary of how these two router redundancy protocols differ, see "Comparison of VRRP and VRRPE" on page 22-8.

Overview

The following sections describe VRRP and VRRPE. The protocols both provide redundant paths for IP addresses. However, the protocols differ in a few important ways. For clarity, each protocol is described separately.

Configuration Note

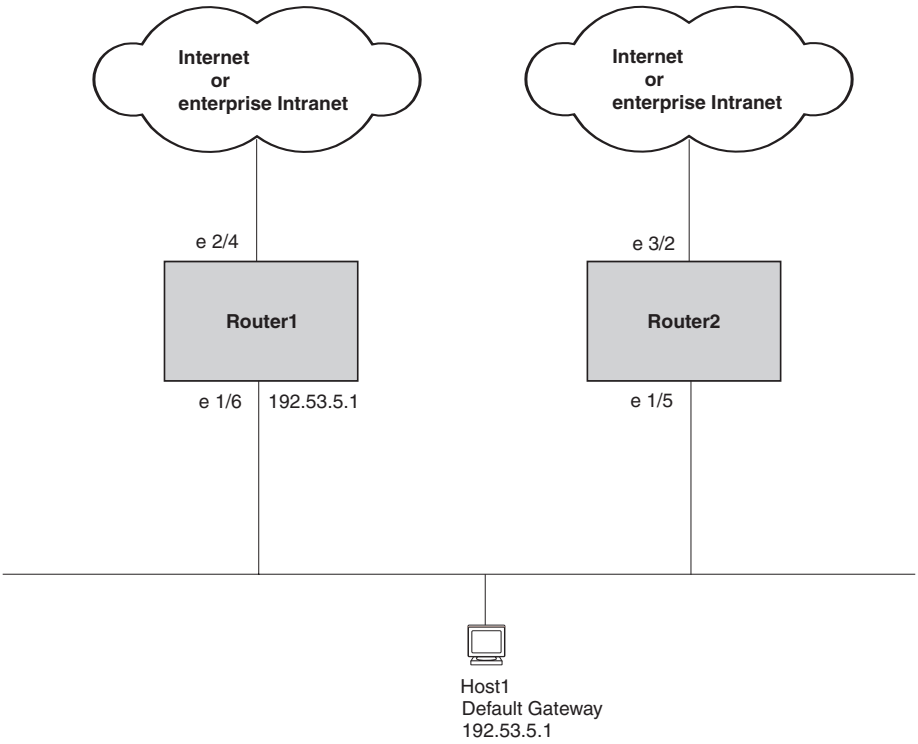
Release 02.4.00 adds support for VRRP in the base Layer 3 code. Previous releases support VRRP in the full Layer 3 code only. VRRP support in the base Layer 3 code is the same as in the full Layer 3 code.

NOTE: VRRP-E is supported in the full Layer 3 code only. It is not supported in the base Layer 3 code.

Overview of VRRP

VRRP is a protocol that provides redundancy to routers within a LAN. VRRP allows you to provide alternate router paths for a host without changing the IP address or MAC address by which the host knows its gateway. Consider the situation shown in Figure 22.1.

Figure 22.1 Router1 is Host1's default gateway but is a single point of failure

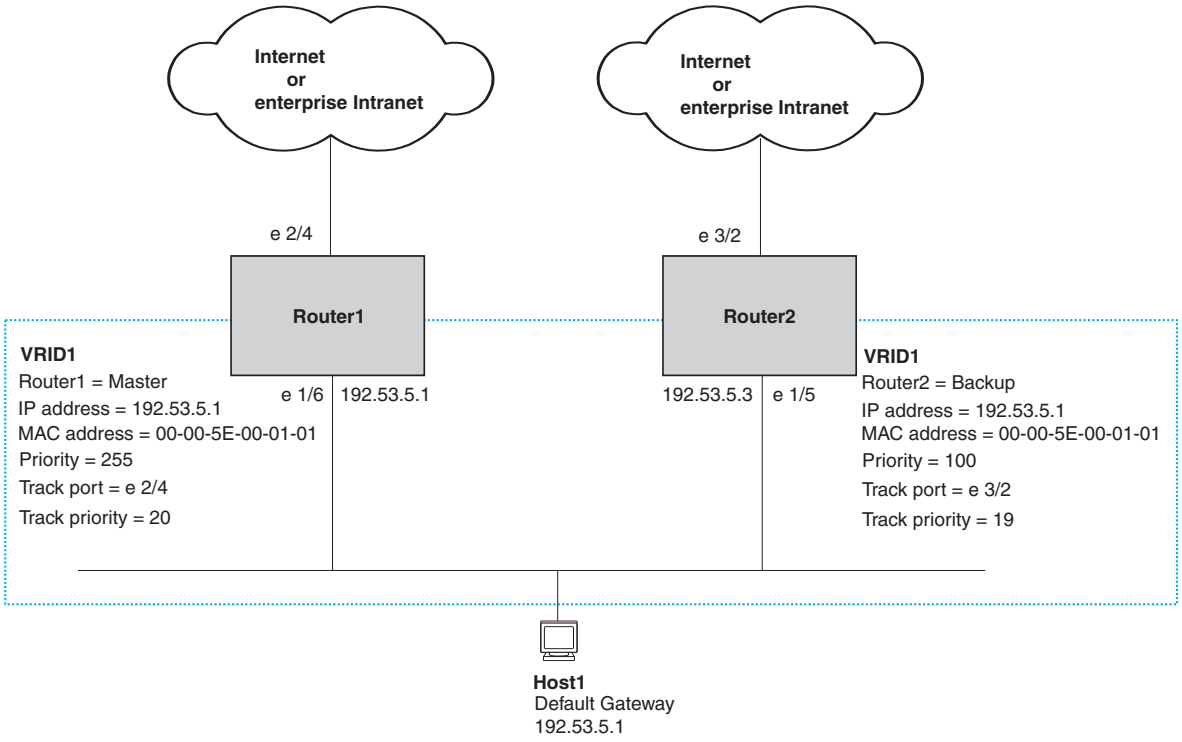


As shown in this example, Host1 uses 192.53.5.1 on Router1 as the host's default gateway out of the sub-net. If this interface goes down, Host1 is cut off from the rest of the network. Router1 is thus a single point of failure for Host1's access to other networks.

If Router1 fails, you could configure Host1 to use Router2. Configuring one host with a different default gateway might not require too much extra administration. However, consider a more realistic network with dozens or even hundreds of hosts per sub-net; reconfiguring the default gateways for all the hosts is impractical. It is much simpler to configure a VRRP virtual router on Router1 and Router2 to provide a redundant path for the host(s).

Figure 22.2 shows the same example network shown in Figure 22.1, but with a VRRP virtual router configured on Router1 and Router2.

Figure 22.2 Router1 and Router2 are configured as a VRRP virtual router to provide redundant network access for Host1



The dashed box in Figure 22.2 represents a VRRP virtual router. When you configure a virtual router, one of the configuration parameters is the virtual router ID (VRID), which can be a number from 1 – 255. In this example, the VRID is 1.

NOTE: You can provide more redundancy by also configuring a second VRID with Router2 as the Owner and Router1 as the Backup. This type of configuration is sometimes called Multigroup VRRP.

Virtual Router ID (VRID)

A ***VRID*** consists of one Master router and one or more Backup routers. The Master router is the router that owns the IP address(es) you associate with the VRID. For this reason, the Master router is sometimes called the "Owner". Configure the VRID on the router that owns the default gateway interface. The other router in the VRID does not own the IP address(es) associated with VRID but provides the backup path if the Master router becomes unavailable.

Virtual Router MAC Address

Notice the MAC address associated with VRID1. The first five octets of the address are the standard MAC prefix for VRRP packets, as described in RFC 2338. The last octet is the VRID. THE VRID number becomes the final octet in the virtual MAC address associated with the virtual router.

When you configure a VRID, the software automatically assigns its MAC address. When a VRID becomes active, the Master router broadcasts a gratuitous ARP request containing the virtual router's MAC address for each IP address associated with the virtual router. In Figure 22.2, Router1 sends a gratuitous ARP with MAC address 00-00-5e-00-01-01 and IP address 192.53.5.1. Hosts use the virtual router's MAC address in routed traffic they send to their default IP gateway (in this example, 192.53.5.1).

Virtual Router IP Address

VRRP does not use virtual IP addresses. Thus, there is no virtual IP address associated with a virtual router. Instead, you associate the virtual router with one or more real interface IP addresses configured on the router that owns the real IP address(es). In Figure 22.2, the virtual router with VRID1 is associated with real IP address 192.53.5.1, which is configured on interface e1/6 on Router1. VRIDs are interface-level parameters, not system-level parameters, so the IP address you associate with the VRID must already be a real IP address configured on the Owner's interface.

NOTE: You also can associate a virtual router with a virtual interface. A virtual interface is a named set of physical interfaces.

When you configure the Backup router for the VRID, specify the same IP address as the one you specify on the Owner. This is the IP address used by the host as its default gateway. The IP address cannot also exist on the Backup router. The interface on which you configure the VRID on the Backup router must have an IP address in the same sub-net.

NOTE: If you delete a real IP address used by a VRRP entry, the VRRP entry also is deleted automatically.

NOTE: When a Backup takes over forwarding responsibilities from a failed Master router, the Backup forwards traffic addressed to the VRID MAC address, which the host believes is the MAC address of the router interface for its default gateway. However, the Backup cannot reply to IP pings sent to the IP address(es) associated with the VRID. Because the IP address(es) are owned by the Owner, if the Owner is unavailable, the IP addresses are unavailable as packet destinations.

Master Negotiation

The routers within a VRID use the VRRP priority values associated with each router to determine which router becomes the Master. When you configure the VRID on a router interface, you specify whether the router is the Owner of the IP address(es) you plan to associate with the VRID or a Backup. If you indicate that the router is the Owner of the IP address(es), the software automatically sets the router's VRRP priority for the VRID to 255, the highest VRRP priority. The router with the highest priority becomes the Master.

Backup routers can have a priority from 3 – 254, which you assign when you configure the VRID on the Backup router's interfaces. The default VRRP priority for Backup routers is 100.

Because the router that owns the IP addresses associated with the VRID always has the highest priority, when all the routers in the virtual router are operating normally, the negotiation process results in the Owner of the VRID's IP address(es) becoming the Master router. Thus, the VRRP negotiation results in the normal case, in which the hosts' path to the default route is to the router that owns the interface for that route.

Hello Messages

VRRP routers use Hello messages for negotiation to determine the Master router. VRRP routers send Hello messages to IP Multicast address 224.0.0.18. The frequency with which the Master sends Hello messages is the Hello Interval. Only the Master sends Hello messages. However, a Backup uses the Hello interval you configure for the Backup if it becomes the Master.

The Backup routers wait for a period of time called the Dead Interval for a Hello message from the Master. If a Backup router does not receive a Hello message by the time the dead interval expires, the Backup router assumes

that the Master router is dead and negotiates with the other Backups to select a new Master router. The Backup router with the highest priority becomes the new Master.

If the Owner becomes unavailable, but then comes back online, the Owner again becomes the Master router. The Owner becomes the Master router again because it has the highest priority. The Owner always becomes the Master again when the Owner comes back online.

NOTE: If you configure a track port on the Owner and the track port is down, the Owner's priority is changed to the track priority. In this case, the Owner does not have a higher priority than the Backup that is acting as Master and the Owner therefore does not resume its position as Master. For more information about track ports, see "Track Ports and Track Priority" on page 22-5.

By default, if a Backup is acting as the Master, and the Master is still unavailable, another Backup can "preempt" the Backup that is acting as the Master. This can occur if the new Backup has a higher priority than the Backup who is acting as Master. You can disable this behavior if you want. When you disable preemption, a Backup router that has a higher priority than the router who is currently acting as Master does not preempt the new Master by initiating a new Master negotiation. See "Backup Preempt" on page 22-17.

NOTE: Regardless of the setting for the preempt parameter, the Owner always becomes the Master again when it comes back online.

Track Ports and Track Priority

The Foundry implementation of VRRP enhances the protocol by giving a VRRP router the capability to monitor the state of the interfaces on the other end of the route path through the router. For example, in Figure 22.2 on page 22-3, interface e1/6 on Router1 owns the IP address to which Host1 directs route traffic on its default gateway. The exit path for this traffic is through Router1's e2/4 interface.

Suppose interface e2/4 goes down. Even if interface e1/6 is still up, Host1 is nonetheless cut off from other networks. In conventional VRRP, Router1 would continue to be the Master router despite the unavailability of the exit interface for the path the router is supporting. However, if you configure interface e1/6 to track the state of interface e2/4, if e2/4 goes down, interface e1/6 responds by changing Router1's VRRP priority to the value of the track priority. In the configuration shown in Figure 22.2 on page 22-3, Router1's priority changes from 255 to 20. One of the parameters contained in the Hello messages the Master router sends to its Backups is the Master router's priority. If the track port feature results in a change in the Master router's priority, the Backup routers quickly become aware of the change and initiate a negotiation for Master router.

In Figure 22.2 on page 22-3, the track priority results in Router1's VRRP priority becoming lower than Router2's VRRP priority. As a result, when Router2 learns that it now has a higher priority than Router1, Router2 initiates negotiation for Master router and becomes the new Master router, thus providing an open path for Host1's traffic. To take advantage of the track port feature, make sure the track priorities are always lower than the VRRP priorities. The default track priority for the router that owns the VRID IP address(es) is 2. The default track priority for Backup routers is 1. If you change the track port priorities, make sure you assign a higher track priority to the Owner of the IP address(es) than the track priority you assign on the Backup routers.

Suppression of RIP Advertisements for Backed Up Interfaces

The Foundry implementation also enhances VRRP by allowing you to configure the protocol to suppress RIP advertisements for the backed up paths from Backup routers. Normally, a VRRP Backup router includes route information for the interface it is backing up in RIP advertisements. As a result, other routers receive multiple paths for the interface and might sometimes unsuccessfully use the path to the Backup rather than the path to the Master. If you enable the Foundry implementation of VRRP to suppress the VRRP Backup routers from advertising the backed up interface in RIP, other routers learn only the path to the Master router for the backed up interface.

Authentication

The Foundry implementation of VRRP can use simple passwords to authenticate VRRP packets. The VRRP authentication type is not a parameter specific to the VRID. Instead, VRRP uses the authentication type associated with the interfaces on which you define the VRID. For example, if you configure your router interfaces to use a simple password to authenticate traffic, VRRP uses the same simple password and VRRP packets that do not contain the password are dropped. If your interfaces do not use authentication, neither does VRRP.

NOTE: The MD5 authentication type is not supported for VRRP.

Independent Operation of VRRP alongside RIP, OSPF, and BGP4

VRRP operation is independent of the RIP, OSPF, and BGP4 protocols. Their operation is unaffected when VRRP is enabled on a RIP, OSPF, or BGP4 interface.

Dynamic VRRP Configuration

All VRRP global and interface parameters take effect immediately. You do not need to reset the system to place VRRP configuration parameters into effect.

Overview of VRRPE

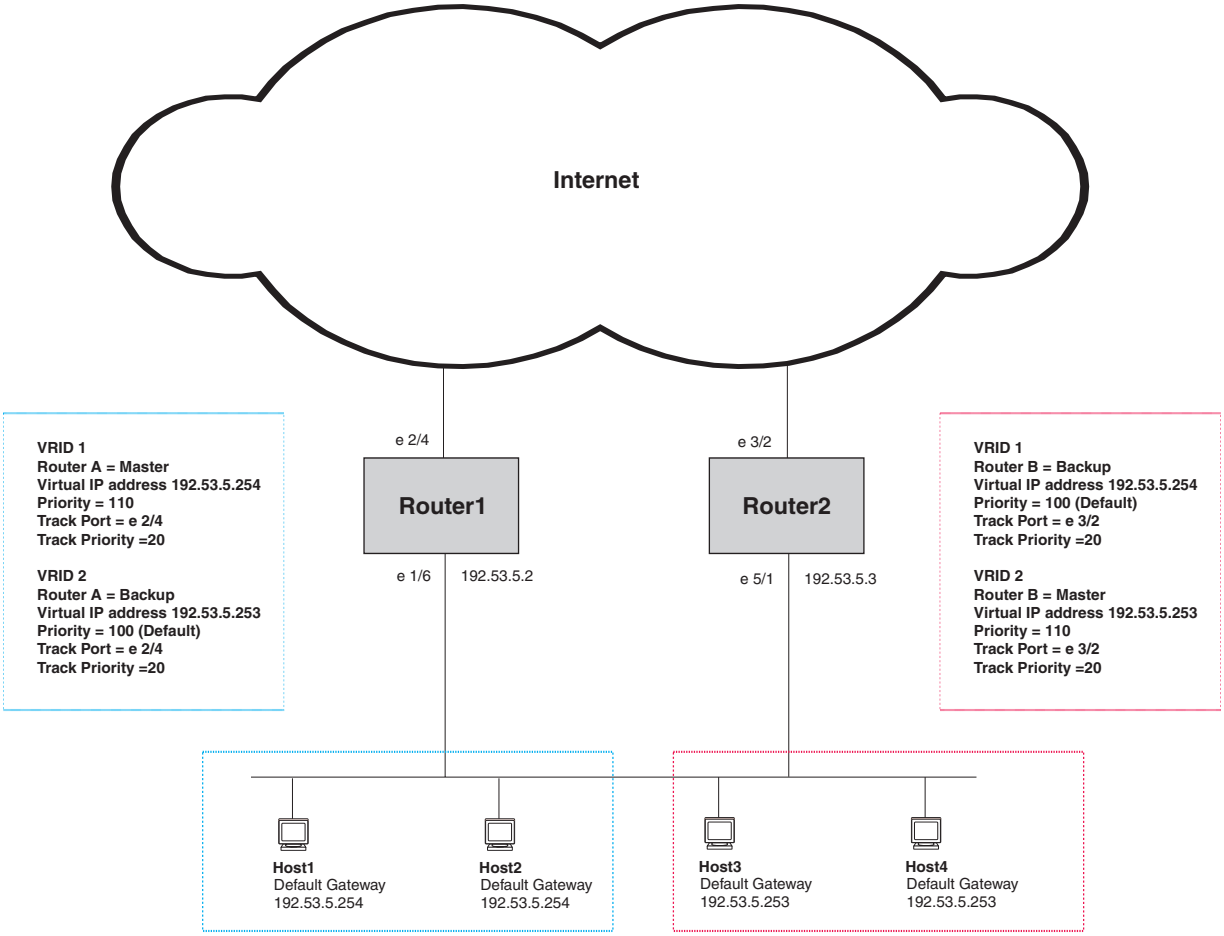
VRRPE is similar to VRRP, but differs in the following respects:

- Owners and Backups
  - VRRP has an Owner and one or more Backups for each VRID. The Owner is the router on which the VRID's IP address is also configured as a real address. All the other routers supporting the VRID are Backups.
  - VRRPE does not use Owners. All routers are Backups for a given VRID. The router with the highest priority becomes Master. If there is a tie for highest priority, the router with the highest IP address becomes Master. The elected Master owns the virtual IP address and answers ping and ARP requests and so on.
- VRID's IP address
  - VRRP requires that the VRID also be a real IP address configured on the VRID's interface on the Owner.
  - VRRPE requires only that the VRID be in the same sub-net as an interface configured on the VRID's interface. In fact, VRRPE does not allow you to specify a real IP address configured on the interface as the VRID IP address.
- VRID's MAC Address
  - VRRP source MAC is a virtual MAC address defined as 00-00-5E-00-01-<vrid>, where <vrid> is the VRID. The Master owns the Virtual MAC address.
  - VRRPE uses the interface's actual MAC address as the source MAC address. The MAC address is 02-E0-52-<hash-value>-<vrid>, where <hash-value> is a two-octet hashed value for the IP address and <vrid> is the VRID.
- Hello packets
  - VRRP sends Hello messages to IP Multicast address 224.0.0.18.
  - VRRPE uses UDP to send Hello messages in IP multicast messages. The Hello packets use the interface's actual MAC address and IP address as the source addresses. The destination MAC address is 01-00-5E-00-00-02, and the destination IP address is 224.0.0.2 (the well-known IP multicast address for "all routers"). Both the source and destination UDP port number is 8888. VRRP messages are encapsulated in the data portion of the packet.
- Track ports and track priority
  - VRRP changes the priority of the VRID to the track priority, which typically is lower than the VRID priority and lower than the VRID's priorities configured on the Backups. For example, if the VRRP interface's priority is 100 and a tracked interface with track priority 20 goes down, the software changes the VRRP interface's priority to 20.
  - VRRPE reduces the priority of a VRRPE interface by the amount of a tracked interface's priority if the tracked interface's link goes down. For example, if the VRRPE interface's priority is 200 and a tracked interface with track priority 20 goes down, the software changes the VRRPE interface's priority to 180. If another tracked interface goes down, the software reduces the VRID's priority again, by the amount of the tracked interface's track priority.

The most important difference is that all VRRPE routers are Backups. There is no Owner router. VRRPE overcomes the limitations in standard VRRP by removing the Owner.

Figure 22.3 shows an example of a VRRPE configuration.

Figure 22.3 Router1 and Router2 are configured to provide dual redundant network access for the host



In this example, RouterA and RouterB use VRRPE to load share as well as provide redundancy to the hosts. The load sharing is accomplished by creating two VRRPE groups. Each group has its own virtual IP addresses. Half of the clients point to VRID 1's virtual IP address as their default gateway and the other half point to VRID 2's virtual IP address as their default gateway. This will enable some of the outbound Internet traffic to go through RouterA and the rest to go through RouterB.

RouterA is the master for VRID 1 (backup priority = 110) and RouterB is the backup for VRID 1 (backup priority = 100). RouterA and RouterB both track the uplinks to the Internet. If an uplink failure occurs on RouterA, its backup priority is decremented by 20 (track priority = 20), so that all traffic destined to the Internet is sent through RouterB instead.

Similarly, RouterB is the master for VRID 2 (backup priority = 110) and RouterA is the backup for VRID 2 (backup priority = 100). RouterA and RouterB are both tracking the uplinks to the Internet. If an uplink failure occurs on RouterB, its backup priority is decremented by 20 (track priority = 20), so that all traffic destined to the internet is sent through RouterA instead.

Configuration Note

VRRP-E is supported in the full Layer 3 code only. It is not supported in the base Layer 3 code.

Comparison of VRRP and VRRPE

This section compares Foundry's router redundancy protocols.

VRRP

VRRP is a standards-based protocol, described in RFC 2338. The Foundry implementation of VRRP contains the features in RFC 2338. The Foundry implementation also provides the following additional features:

- Track ports – A Foundry feature that enables you to diagnose the health of all the Layer 3 Switch's ports used by the backed-up VRID, instead of only the port connected to the client sub-net. See "Track Ports and Track Priority" on page 22-5.
- Suppression of RIP advertisements on Backup routes for the backed up interface – You can enable the Layer 3 Switches to advertise only the path to the Master router for the backed up interface. Normally, a VRRP Backup router includes route information for the interface it is backing up in RIP advertisements.

Foundry Layer 3 Switches configured for VRRP can interoperate with third-party routers using VRRP.

VRRPE

VRRPE is a Foundry protocol that provides the benefits of VRRP without the limitations. VRRPE is unlike VRRP in the following ways:

- There is no "Owner" router. You do not need to use an IP address configured on one of the Layer 3 Switches as the virtual router ID (VRID), which is the address you are backing up for redundancy. The VRID is independent of the IP interfaces configured in the Layer 3 Switches. As a result, the protocol does not have an "Owner" as VRRP does.
- There is no restriction on which router can be the default master router. In VRRP, the "Owner" (the Layer 3 Switch on which the IP interface that is used for the VRID is configured) must be the default Master.

Foundry Layer 3 Switches configured for VRRPE can interoperate only with other Foundry Layer 3 Switches.

Architectural Differences

The protocols have the following architectural differences.

Management Protocol

- VRRP – VRRP routers send VRRP Hello and Hello messages to IP Multicast address 224.0.0.18.
- VRRPE – VRRPE sends messages to destination MAC address 01-00-5E-00-00-02 and destination IP address 224.0.0.2 (the standard IP multicast address for "all routers").

Virtual Router IP Address (the address you are backing up)

- VRRP – The virtual router IP address is the same as an IP address or virtual interface configured on one of the Layer 3 Switches, which is the "Owner" and becomes the default Master.
- VRRPE – The virtual router IP address is the gateway address you want to backup, but does not need to be an IP interface configured on one of the Layer 3 Switch's ports or a virtual interface.

Master and Backups

- VRRP – The "Owner" of the IP address of the VRID is the default Master and has the highest priority (255). The precedence of the Backups is determined by their priorities. The default Master is always the Owner of the IP address of the VRID.
- VRRPE – The Master and Backups are selected based on their priority. You can configure any of the Layer 3 Switches to be the Master by giving it the highest priority. There is no Owner.

VRRP and VRRPE Parameters

Table 22.2 lists the VRRP and VRRPE parameters. Most of the parameters and default values are the same for both protocols. The exceptions are noted in the table.

Table 22.2: VRRP and VRRPE Parameters

Parameter	Description	Default	See page...
Protocol	The Virtual Router Redundancy Protocol (VRRP) based on RFC 2338 or VRRP-Extended, Foundry's enhanced implementation of VRRP	Disabled **Note**: Only one of the protocols can be enabled at a time.	22-11 22-12
VRRP or VRRPE router	The Foundry Layer 3 Switch's active participation as a VRRP or VRRPE router. Enabling the protocol does not activate the Layer 3 Switch for VRRP or VRRPE. You must activate the device as a VRRP or VRRPE router after you configure the VRRP or VRRPE parameters.	Inactive	22-11 22-12
Virtual Router ID (VRID)	The ID of the virtual router you are creating by configuring multiple routers to back up an IP interface. You must configure the same VRID on each router that you want to use to back up the address. No default.	None	22-3 22-11 22-12
Virtual Router IP address	This is the address you are backing up. No default. • VRRP – The virtual router IP address must be a real IP address configured on the VRID interface on one of the VRRP routers. This router is the IP address Owner and is the default Master. • VRRPE – The virtual router IP address must be in the same sub-net as a real IP address configured on the VRRPE interface, but cannot be the same as a real IP address configured on the interface.	None	22-4 22-11 22-12
VRID MAC address	The source MAC address in VRRP or VRRPE packets sent from the VRID interface, and the destination for packets sent to the VRID. • VRRP – A virtual MAC address defined as 00-00-5e-00-01-<vrid>. The Master owns the Virtual MAC address. • VRRPE – A virtual MAC address defined as 02-E0-52-<hash-value>-<vrid>, where <hash-value> is a two-octet hashed value for the IP address and <vrid> is the VRID.	Not configurable	22-4

Table 22.2: VRRP and VRRPE Parameters (Continued)

Parameter	Description	Default	See page...
Authentication type	The type of authentication the VRRP or VRRPE routers use to validate VRRP or VRRPE packets. The authentication type must match the authentication type the VRID's port uses with other routing protocols such as OSPF. • No authentication – The interfaces do not use authentication. This is the VRRP default. • Simple – The interface uses a simple text-string as a password in packets sent on the interface. If the interface uses simple password authentication, the VRID configured on the interface must use the same authentication type and the same password. **Note**: MD5 is not supported by VRRP or VRRPE.	No authentication	22-5 22-13
Router type	Whether the router is an Owner or a Backup. • Owner (VRRP only) – The router on which the real IP address used by the VRID is configured. • Backup – Routers that can provide routing services for the VRID but do not have a real IP address matching the VRID.	VRRP – The Owner is always the router that has the real IP address used by the VRID. All other routers for the VRID are Backups. VRRPE – All routers for the VRID are Backups.	22-14
Backup priority	A numeric value that determines a Backup's preferability for becoming the Master for the VRID. During negotiation, the router with the highest priority becomes the Master. • VRRP – The Owner has the highest priority (255); other routers can have a priority from 3 – 254. • VRRPE – All routers are Backups and have the same priority by default. If two or more Backups are tied with the highest priority, the Backup interface with the highest IP address becomes the Master for the VRID.	VRRP – 255 for the Owner; 100 for each Backup VRRPE – 100 for all Backups	22-14
Suppression of RIP advertisements	A router that is running RIP normally advertises routes to a backed up VRID even when the router is not currently the active router for the VRID. Suppression of these advertisements helps ensure that other routers do not receive invalid route paths for the VRID.	Disabled	22-15
Hello interval	The number of seconds between Hello messages from the Master to the Backups for a given VRID. The interval can from 1 – 84 seconds.	One second	22-4 22-16

Table 22.2: VRRP and VRRPE Parameters (Continued)

Parameter	Description	Default	See page...
Dead interval	The number of seconds a Backup waits for a Hello message from the Master for the VRID before determining that the Master is no longer active. If the Master does not send a Hello message before the dead interval expires, the Backups negotiate (compare priorities) to select a new Master for the VRID.	Three times the Hello Interval plus one-half second	22-4 22-16
Backup Hello interval	The number of seconds between Hello messages from a Backup to the Master. The message interval can be from 60 – 3600 seconds. You must enable the Backup to send the messages. The messages are disabled by default on Backups. The current Master (whether the VRRP Owner or a Backup) sends Hello messages by default.	Disabled 60 seconds when enabled	22-4 22-16
Track port	Another Layer 3 Switch port or virtual interface whose link status is tracked by the VRID's interface. If the link for a tracked interface goes down, the VRRP or VRRPE priority of the VRID interface is changed, causing the devices to renegotiate for Master.	None	22-5 22-17
Track priority	A VRRP or VRRPE priority value assigned to the tracked port(s). If a tracked port's link goes down, the VRID port's VRRP or VRRPE priority changes. • VRRP – The priority changes to the value of the tracked port's priority. • VRRPE – The VRID port's priority is reduced by the amount of the tracked port's priority.	VRRP – 2 VRRPE – 5	22-5 22-17
Backup preempt mode	Prevents a Backup with a higher VRRP priority from taking control of the VRID from another Backup that has a lower priority but has already assumed control of the VRID.	Enabled	22-17

Configuring Basic VRRP Parameters

To implement a simple VRRP configuration using all the default values, enter commands such as the following.

Configuring the Owner

```
Router1(config)# router vrrp
Router1(config)# inter e 1/6
Router1(config-if-1/6)# ip address 192.53.5.1
Router1(config-if-1/6)# ip vrrp vrid 1
Router1(config-if-1/6-vrid-1)# owner
Router1(config-if-1/6-vrid-1)# ip-address 192.53.5.1
Router1(config-if-1/6-vrid-1)# activate
```

Configuring a Backup

```
Router2(config)# router vrrp
Router2(config)# inter e 1/5
Router2(config-if-1/5)# ip address 192.53.5.3
Router2(config-if-1/5)# ip vrrp vrid 1
Router2(config-if-1/5-vrid-1)# backup
Router2(config-if-1/5-vrid-1)# ip-address 192.53.5.1
Router2(config-if-1/5-vrid-1)# activate
```

Configuration Rules for VRRP

- The interfaces of all routers in a VRID must be in the same IP sub-net.
- The IP address(es) associated with the VRID must already be configured on the router that will be the Owner router.
- An IP address(es) associated with the VRID must be on only one router.
- The Hello interval must be set to the same value on both the Owner and Backup(s) for the VRID.
- The Dead interval must be set to the same value on both the Owner and Backup(s) for the VRID.
- The track priority on a router must be lower than the router's VRRP priority. Also, the track priority on the Owner must be higher than the track priority on the Backup(s).

Configuring Basic VRRPE Parameters

To implement a simple VRRPE configuration using all the default values, enter commands such as the following on each Layer 3 Switch.

```
Router2(config)# router vrrp-extended
Router2(config)# inter e 1/5
Router2(config-if-1/5)# ip address 192.53.5.3
Router2(config-if-1/5)# ip vrrp-extended vrid 1
Router2(config-if-1/5-vrid-1)# backup
Router2(config-if-1/5-vrid-1)# ip-address 192.53.5.254
Router2(config-if-1/5-vrid-1)# activate
```

NOTE: You also can use the **enable** command to activate the configuration. This command does the same thing as the **activate** command.

Configuration Rules for VRRPE

- The interfaces of all routers in a VRID must be in the same IP sub-net.
- The IP address(es) associated with the VRID cannot be configured on any of the Layer 3 Switches.
- The Hello interval must be set to the same value on all the Layer 3 Switches.
- The Dead interval must be set to the same value on all the Layer 3 Switches.
- The track priority for a VRID must be lower than the VRRPE priority.

Note Regarding Disabling VRRP or VRRPE

If you disable VRRP or VRRPE, the Layer 3 Switch removes all the configuration information for the disabled protocol from the running-config. Moreover, when you save the configuration to the startup-config file after disabling one of these protocols, all the configuration information for the disabled protocol is removed from the startup-config file.

The CLI displays a warning message such as the following:

```
Router1(config-vrrp-router)# no router vrrp
router vrrp mode now disabled. All vrrp config data will be lost when writing to
flash!
```

If you have disabled the protocol but have not yet saved the configuration to the startup-config file and reloaded the software, you can restore the configuration information by re-entering the command to enable the protocol (ex: **router vrrp**). If you have already saved the configuration to the startup-config file and reloaded the software, the information is gone.

If you are testing a VRRP or VRRPE configuration and are likely to disable and re-enable the protocol, you might want to make a backup copy of the startup-config file containing the protocol's configuration information. This way, if you remove the configuration information by saving the configuration after disabling the protocol, you can restore the configuration by copying the backup copy of the startup-config file onto the flash memory.

Configuring Additional VRRP and VRRPE Parameters

You can modify the following VRRP and VRRPE parameters on an individual VRID basis. These parameters apply to both protocols:

- Authentication type (if the interfaces on which you configure the VRID use authentication)
- Router type (Owner or Backup)

NOTE: For VRRP, change the router type only if you have moved the real IP address from one router to another or you accidentally configured the IP address Owner as a Backup.

For VRRPE, the router type is always Backup. You cannot change the type to Owner.

- Backup priority
- Suppression of RIP advertisements on Backup routes for the backed up interface
- Hello interval
- Dead interval
- Backup Hello messages and message timer (Backup advertisement)
- Track port
- Track priority
- Backup preempt mode

For information about the fields, see the parameter descriptions in the following sections.

See "VRRP and VRRPE Parameters" on page 22-9 for a summary of the parameters and their defaults.

Authentication Type

If the interfaces on which you configure the VRID use authentication, the VRRP or VRRPE packets on those interfaces also must use the same authentication. Foundry's implementation of VRRP and VRRPE supports the following authentication types:

- No authentication – The interfaces do not use authentication. This is the default for VRRP and VRRPE.
- Simple – The interfaces use a simple text-string as a password in packets sent on the interface. If the interfaces use simple password authentication, the VRID configured on the interfaces must use the same authentication type and the same password.

To configure the VRID interface on Router1 for simple-password authentication using the password "ourpword", enter the following commands:

Configuring Router 1

```
Router1(config)# inter e 1/6
Router1(config-if-1/6)# ip vrrp auth-type simple-text-auth ourpword
```

Configuring Router 2

```
Router2(config)# inter e 1/5
Router2(config-if-1/5)# ip vrrp auth-type simple-text-auth ourpword
```

VRRP Syntax

Syntax: ip vrrp auth-type no-auth | simple-text-auth <auth-data>

The **auth-type no-auth** parameter indicates that the VRID and the interface it is configured on do not use authentication.

The **auth-type simple-text-auth** <auth-data> parameter indicates that the VRID and the interface it is configured on use a simple text password for authentication. The <auth-data> parameter is the password. If you use this parameter, make sure all interfaces on all the routers supporting this VRID are configured for simple password authentication and use the same password.

VRRPE Syntax

Syntax: ip vrrp-extended auth-type no-auth | simple-text-auth <auth-data>

The parameter values are the same as for VRRP.

Router Type

A VRRP interface is either an Owner or a Backup for a given VRID. By default, the Owner becomes the Master following the negotiation. A Backup becomes the Master only if the Master becomes unavailable.

A VRRPE interface is always a Backup for its VRID. The Backup with the highest VRRP priority becomes the Master.

This section describes how to specify the interface type, how to change the type for VRRP, and how to set or change the interface's VRRP or VRRPE priority and track priority for the VRID.

NOTE: You can force a VRRP master router to abdicate (give away control) of the VRID to a Backup by temporarily changing the Master's VRRP priority to a value less than the Backup's. See "Forcing a Master Router To Abdicate to a Standby Router" on page 22-18.

NOTE: The type Owner is not applicable to VRRPE.

NOTE: The IP address(es) you associate with the Owner must be a real IP address (or addresses) on the interface on which you configure the VRID.

When you configure a Backup router, the router interface on which you are configuring the VRID must have a real IP address that is in the same sub-net as the address associated with the VRID by the Owner. However, the address cannot be the same.

To configure Router1 as a VRRP VRID's Owner, enter the following commands:

```
Router1(config)# inter e 1/6
Router1(config-if-1/6)# ip vrrp vrid 1
Router1(config-if-1/6-vrid-1)# owner
```

To configure Router2 as a VRRP Backup for the same VRID, enter the following commands:

```
Router2(config)# inter e 1/5
Router2(config-if-1/5)# ip vrrp vrid 1
Router2(config-if-1/5-vrid-1)# backup
```

To configure a VRRPE interface as a Backup for a VRID and set its VRRPE priority and track priority, enter commands such as the following:

```
FastIron SuperX Router(config)# inter e 1/1
FastIron SuperX Switch(config-if-1/1)# ip vrrp-extended vrid 1
FastIron SuperX Switch(config-if-1/1-vrid-1)# backup priority 50 track-priority 10
```

VRRP Syntax

Syntax: owner [track-priority <value>]

The **track-priority** <value> parameter changes the track-port priority for this interface and VRID from the default (2) to a value from 1 – 254.

Syntax: backup [priority <value>] [track-priority <value>]

The **priority** <value> parameter specifies the VRRP priority for this interface and VRID. You can specify a value from 3 – 254. The default is 100.

The **track-priority** <value> parameter is the same as above.

NOTE: You cannot set the priority of a VRRP Owner. The Owner's priority is always 255.

VRRPE Syntax

Syntax: backup [priority <value>] [track-priority <value>]

The software requires you to identify a VRRPE interface as a Backup for its VRID before you can activate the interface for the VRID. However, after you configure the VRID, you can use this command to change its priority or track priority. The parameter values are the same as for VRRP.

Suppression of RIP Advertisements on Backup Routers for the Backup Up Interface

Normally, a VRRP or VRRPE Backup includes route information for the virtual IP address (the backed up interface) in RIP advertisements. As a result, other routers receive multiple paths for the backed up interface and might sometimes unsuccessfully use the path to the Backup rather than the path to the Master.

You can prevent the Backups from advertising route information for the backed up interface by enabling suppression of the advertisements.

To suppress RIP advertisements for the backed up interface in Router2, enter the following commands:

```
Router2(config)# router rip
Router2(config-rip-router)# use-vrrp-path
```

Syntax: use-vrrp-path

The syntax is the same for VRRP and VRRPE.

Hello Interval

The Master periodically sends Hello messages to the Backups. The Backups use the Hello messages as verification that the Master is still on-line. If the Backup routers stop receiving the Hello messages for the period of time specified by the Dead interval, the Backup routers determine that the Master router is dead. At this point, the Backup router with the highest priority becomes the new Master router. The Hello interval can be from 1 – 84 seconds. The default is 1 second.

NOTE: The default Dead interval is three times the Hello Interval plus one-half second. Generally, if you change the Hello interval, you also should change the Dead interval on the Backup routers.

To change the Hello interval on the Master to 10 seconds, enter the following commands:

```
Router1(config)# inter e 1/6
Router1(config-if-1/6)# ip vrrp vrid 1
Router1(config-if-1/6-vrid-1)# hello-interval 10
```

Syntax: hello-interval <value>

The syntax is the same for VRRP and VRRPE.

Dead Interval

The Dead interval is the number of seconds a Backup waits for a Hello message from the Master before determining that the Master is dead. When Backups determine that the Master is dead, the Backup with the highest priority becomes the new Master. The Dead interval can be from 1 – 84 seconds. The default is 3.5 seconds. This is three times the default Hello interval (1 second) plus one-half second added by the router software. The software automatically adds one-half second to the Dead interval value you enter.

To change the Dead interval on a Backup to 30 seconds, enter the following commands:

```
Router2(config)# inter e 1/5
Router2(config-if-1/5)# ip vrrp vrid 1
Router2(config-if-1/5-vrid-1)# dead-interval 30
```

Syntax: dead-interval <value>

The syntax is the same for VRRP and VRRPE.

Backup Hello Message State and Interval

By default, Backup do not send Hello messages to advertise themselves to the Master. You can enable these messages if desired and also change the message interval.

To enable a Backup to send Hello messages to the Master, enter commands such as the following:

```
FESX424 Router(config)# router vrrp
FESX424 Router(config)# inter e 1/6
FESX424 Router(config-if-1/6)# ip vrrp vrid 1
FESX424 Router(config-if-1/6-vrid-1)# advertise backup
```

Syntax: [no] advertise backup

When you enable a Backup to send Hello messages, the Backup sends a Hello messages to the Master every 60 seconds by default. You can change the interval to be up to 3600 seconds. To do so, enter commands such as the following:

```
FESX424 Router(config)# router vrrp
FESX424 Router(config)# inter e 1/6
FESX424 Router(config-if-1/6)# ip vrrp vrid 1
FESX424 Router(config-if-1/6-vrid-1)# backup-hello-interval 180
```

Syntax: [no] backup-hello-interval <num>

The <num> parameter specifies the message interval and can be from 60 – 3600 seconds. The default is 60 seconds.

The syntax is the same for VRRP and VRRPE.

Track Port

You can configure the VRID on one interface to track the link state of another interface on the Layer 3 Switch. This capability is quite useful for tracking the state of the exit interface for the path for which the VRID is providing redundancy. See “Track Ports and Track Priority” on page 22-5.

To configure 1/6 on Router1 to track interface 2/4, enter the following commands:

```
Router1(config)# inter e 1/6
Router1(config-if-1/6)# ip vrrp vrid 1
Router1(config-if-1/6-vrid-1)# track-port e 2/4
```

Syntax: track-port ethernet [<slotnum>/]<portnum> | ve <num>

The syntax is the same for VRRP and VRRPE.

Track Priority

When you configure a VRID to track the link state of other interfaces, if one of the tracked interface goes down, the software changes the VRRP or VRRPE priority of the VRID interface.

- For VRRP, the software changes the priority of the VRID to the track priority, which typically is lower than the VRID priority and lower than the VRID’s priorities configured on the Backups. For example, if the VRRPE interface’s priority is 100 and a tracked interface with track priority 60 goes down, the software changes the VRRPE interface’s priority to 60.
- For VRRPE, the software reduces the VRID priority by the amount of the priority of the tracked interface that went down. For example, if the VRRPE interface’s priority is 100 and a tracked interface with track priority 60 goes down, the software changes the VRRPE interface’s priority to 40. If another tracked interface goes down, the software reduces the VRID’s priority again, by the amount of the tracked interface’s track priority.

The default track priority for a VRRP Owner is 2. The default track priority for Backups is 1.

You enter the track priority as a parameter with the **owner** or **backup** command. See “Track Port” on page 22-17.

Syntax: owner [track-priority <value>]

Syntax: backup [priority <value>] [track-priority <value>]

The syntax is the same for VRRP and VRRPE.

Backup Preempt

By default, a Backup that has a higher priority than another Backup that has become the Master can preempt the Master, and take over the role of Master. If you want to prevent this behavior, disable preemption.

Preemption applies only to Backups and takes effect only when the Master has failed and a Backup has assumed ownership of the VRID. The feature prevents a Backup with a higher priority from taking over as Master from another Backup that has a lower priority but has already become the Master of the VRID.

Preemption is especially useful for preventing flapping in situations where there are multiple Backups and a Backup with a lower priority than another Backup has assumed ownership, because the Backup with the higher priority was unavailable when ownership changed.

If you enable the non-preempt mode (thus disabling the preemption feature) on all the Backups, the Backup that becomes the Master following the disappearance of the Master continues to be the Master. The new Master is not preempted.

NOTE: In VRRP, regardless of the setting for the preempt parameter, the Owner always becomes the Master again when it comes back online.

To disable preemption on a Backup, enter commands such as the following:

```
Router1(config)# inter e 1/6
Router1(config-if-1/6)# ip vrrp vrid 1
Router1(config-if-1/6-vrid-1)# non-preempt-mode
```

Syntax: non-preempt-mode

The syntax is the same for VRRP and VRRPE.

Forcing a Master Router To Abdicate to a Standby Router

You can force a VRRP Master to abdicate (give away control) of a VRID to a Backup by temporarily changing the Master's priority to a value less than the Backup's.

The VRRP Owner always has priority 255. You can even use this feature to temporarily change the Owner's priority to a value from 1 – 254.

NOTE: When you change a VRRP Owner's priority, the change takes effect only for the current power cycle. The change is not saved to the startup-config file when you save the configuration and is not retained across a reload or reboot. Following a reload or reboot, the VRRP Owner again has priority 255.

To temporarily change the Master's priority, use the following CLI method.

To change the Master's priority, enter commands such as the following:

```
FastIron SuperX Router(config)# ip int eth 1/6
FastIron SuperX Router(config-if-1/6)# ip vrrp vrid 1
FastIron SuperX Router(config-if-1/6-vrid-1)# owner priority 99
```

Syntax: [no] owner priority | track-priority <num>

The <num> parameter specifies the new priority and can be a number from 1 – 254.

When you press Enter, the software changes the priority of the Master to the specified priority. If the new priority is lower than at least one Backup's priority for the same VRID, the Backup takes over and becomes the new Master until the next software reload or system reset.

To verify the change, enter the following command from any level of the CLI:

```
FastIron SuperX Router(config-if-1/6-vrid-1)# show ip vrrp
Total number of VRRP routers defined: 1
Interface ethernet 1/6
auth-type no authentication
VRID 1
state backup
administrative-status enabled
mode owner
priority 99
current priority 99
hello-interval 1 sec
ip-address 192.53.5.1
backup routers 192.53.5.2
```

This example shows that even though this Layer 3 Switch is the Owner of the VRID ("mode owner"), the Layer 3 Switch's priority for the VRID is only 99 and the state is now "backup" instead of "active". In addition, the administrative status is "enabled".

To change the Master's priority back to the default Owner priority 255, enter "no" followed by the command you entered to change the priority. For example, to change the priority of a VRRP Owner back to 255 from 99, enter the following command:

```
FastIron SuperX Router(config-if-1/6-vrid-1)# no owner priority 99
```

You cannot set the priority to 255 using the **owner priority** command.

Displaying VRRP and VRRPE Information

You can display the following information for VRRP or VRRPE:

- Summary configuration and status information
- Detailed configuration and status information
- VRRP and VRRPE Statistics
- CPU utilization statistics

Displaying Summary Information

To display summary information for a Layer 3 Switch, enter the following command at any level of the CLI:

```
FastIron SuperX Switch(config-if-e1000-1/6-vrid-1)# show ip vrrp brief

Total number of VRRP routers defined: 1
Interface VRID CurPri P State  Master addr   Backup addr       VIP
 1/6        1    255  P Init   192.53.5.1    192.53.5.3 192.53.5.1
```

This example is for VRRP. Here is an example for VRRPE:

```
FastIron SuperX Switch(config-if-e1000-1/6-vrid-1)# show ip vrrp-extended brief

Total number of VRRP-Extended routers defined: 1
Interface VRID CurPri P State  Master addr   Backup addr       VIP
 1/6        1    255  P Init   192.53.5.2    192.53.5.3 192.53.5.254
```

Syntax: show ip vrrp brief | ethernet [<slotnum>/]<portnum> | ve <num> | stat

Syntax: show ip vrrp-extended brief | ethernet [<slotnum>/]<portnum> | ve <num> | stat

The **brief** parameter displays the summary information. If you do not use this parameter, detailed information is displayed instead. See “Displaying Detailed Information” on page 22-20.

The <slotnum> parameter is required on chassis devices if you specify a port number.

The <portnum> parameter specifies an Ethernet port. If you use this parameter, the command displays VRRP or VRRPE information only for the specified port.

The **ve** <num> parameter specifies a virtual interface. If you use this parameter, the command displays VRRP or VRRPE information only for the specified virtual interface.

The **stat** parameter displays statistics. See “Displaying Statistics” on page 22-26.

This display shows the following information.

Table 22.3: CLI Display of VRRP or VRRPE Summary Information

This Field...	Displays...
Total number of VRRP (or VRRP-Extended) routers defined	The total number of VRIDs configured on this Layer 3 Switch. **Note**: The total applies only to the protocol the Layer 3 Switch is running. For example, if the Layer 3 Switch is running VRRPE, the total applies only to VRRPE routers.
Interface	The interface on which VRRP or VRRPE is configured. If VRRP or VRRPE is configured on multiple interfaces, information for each interface is listed separately.

Table 22.3: CLI Display of VRRP or VRRPE Summary Information (Continued)

This Field...	Displays...
VRID	The VRID configured on this interface. If multiple VRIDs are configured on the interface, information for each VRID is listed in a separate row.
CurPri	The current VRRP or VRRPE priority of this Layer 3 Switch for the VRID.
P	Whether the backup preempt mode is enabled. If the backup preempt mode is enabled, this field contains a "P". If the mode is disabled, this field is blank.
State	This Layer 3 Switch's VRRP or VRRPE state for the VRID. The state can be one of the following: • Init – The VRID is not enabled (activated). If the state remains Init after you activate the VRID, make sure that the VRID is also configured on the other routers and that the routers can communicate with each other. **Note**: If the state is Init and the mode is incomplete, make sure you have specified the IP address for the VRID. • Backup – This Layer 3 Switch is a Backup for the VRID. • Master – This Layer 3 Switch is the Master for the VRID.
Master addr	The IP address of the router interface that is currently the Master for the VRID.
Backup addr	The IP addresses of the router interfaces that are currently Backups for the VRID.
VIP	The virtual IP address that is being backed up by the VRID.

Displaying Detailed Information

To display detailed VRRP or VRRPE information, enter the following command at any level of the CLI:

```
FastIron SuperX Router(config)# show ip vrrp

Total number of VRRP routers defined: 1
Interface ethernet 1/6
 auth-type no authentication
 VRID 1
  state master
  administrative-status enabled
  mode owner
  priority 255
  current priority 255
  hello-interval 1 sec
  advertise backup: disabled
  track-port 2/4
```


This example is for a VRRP Owner. Here is an example for a VRRP Backup.

```
FastIron SuperX Router(config)# show ip vrrp

Total number of VRRP routers defined: 1
Interface ethernet 1/5
 auth-type no authentication
 VRID 1
  state backup
  administrative-status enabled
  mode non-owner(backup)
  priority 100
  current priority 100
  hello-interval 1 sec
  dead-interval 3.600 sec
  current dead-interval 3.600 sec
  preempt-mode true
  advertise backup: enabled
  backup router 192.53.5.3 expires in 00:00:03
  next hello sent in 00:00:02
  track-port 3/2
```

Here is an example for a VRRPE Backup.

```
FastIron SuperX Router(config)# show ip vrrp-extended

Total number of VRRP-Extended routers defined: 1
Interface ethernet 1/6
 auth-type no authentication
 VRID 1
  state master
  administrative-status enabled
  priority 200
  current priority 200
  hello-interval 1 sec
  dead-interval 3.600 sec
  current dead-interval 3.600 sec
  preempt-mode true
  virtual ip address 192.53.5.254
  advertise backup: enabled
  master router 192.53.5.2 expires in 00:00:03
  track-port 2/4
```

Syntax: show ip vrrp brief | ethernet [<slotnum>/]<portnum> | ve <num> | stat

Syntax: show ip vrrp-extended brief | ethernet [<slotnum>/]<portnum> | ve <num> | stat

The **brief** parameter displays summary information. See “Displaying Summary Information” on page 22-19.

The <portnum> parameter specifies an Ethernet port. If you use this parameter, the command displays VRRP or VRRPE information only for the specified port. Also, you must specify the <slotnum> on chassis devices.

The **ve** <num> parameter specifies a virtual interface. If you use this parameter, the command displays VRRP or VRRPE information only for the specified virtual interface.

The **stat** parameter displays statistics. See “Displaying Statistics” on page 22-26.

This display shows the following information.

Table 22.4: CLI Display of VRRP or VRRPE Detailed Information

This Field...	Displays...
Total number of VRRP (or VRRP-Extended) routers defined	The total number of VRIDs configured on this Layer 3 Switch. **Note**: The total applies only to the protocol the Layer 3 Switch is running. For example, if the Layer 3 Switch is running VRRPE, the total applies only to VRRPE routers.
Interface parameters	
Interface	The interface on which VRRP or VRRPE is configured. If VRRP or VRRPE is configured on multiple interfaces, information for each interface is listed separately.
auth-type	The authentication type enabled on the interface.
VRID parameters	
VRID	The VRID configured on this interface. If multiple VRIDs are configured on the interface, information for each VRID is listed separately.
state	This Layer 3 Switch's VRRP or VRRPE state for the VRID. The state can be one of the following: • initialize – The VRID is not enabled (activated). If the state remains "initialize" after you activate the VRID, make sure that the VRID is also configured on the other routers and that the routers can communicate with each other. **Note**: If the state is "initialize" and the mode is incomplete, make sure you have specified the IP address for the VRID. • backup – This Layer 3 Switch is a Backup for the VRID. • master – This Layer 3 Switch is the Master for the VRID.
administrative-status	The administrative status of the VRID. The administrative status can be one of the following: • disabled – The VRID is configured on the interface but VRRP or VRRPE has not been activated on the interface. • enabled – VRRP or VRRPE has been activated on the interface.
mode	Indicates whether the Layer 3 Switch is the Owner or a Backup for the VRID. **Note**: If "incomplete" appears after the mode, configuration for this VRID is incomplete. For example, you might not have configured the virtual IP address that is being backup up by the VRID. **Note**: This field applies only to VRRP. All Layer 3 Switches configured for VRRPE are Backups.

Table 22.4: CLI Display of VRRP or VRRPE Detailed Information (Continued)

This Field...	Displays...
priority	The device's preferability for becoming the Master for the VRID. During negotiation, the router with the highest priority becomes the Master. If two or more devices are tied with the highest priority, the Backup interface with the highest IP address becomes the active router for the VRID.
current priority	The current VRRP or VRRPE priority of this Layer 3 Switch for the VRID. The current priority can differ from the configured priority (see the row above) for the following reasons: • The VRID is still in the initialization stage and has not become a Master or Backup yet. In this case, the current priority is 0. • The VRID is configured with track ports and the link on a tracked interface has gone down. See "Track Ports and Track Priority" on page 22-5.
hello-interval	The number of seconds between Hello messages from the Master to the Backups for a given VRID.
dead-interval	The configured value for the dead interval. The dead interval is the number of seconds a Backup waits for a Hello message from the Master for the VRID before determining that the Master is no longer active. If the Master does not send a Hello message before the dead interval expires, the Backups negotiate (compare priorities) to select a new Master for the VRID. **Note**: If the value is 0, then you have not configured this parameter. **Note**: This field does not apply to VRRP Owners.
current dead-interval	The current value of the dead interval. This is the value actually in use by this interface for the VRID. **Note**: This field does not apply to VRRP Owners.
preempt-mode	Whether the backup preempt mode is enabled. **Note**: This field does not apply to VRRP Owners.
virtual ip address	The virtual IP addresses that this VRID is backing up.
advertise backup	The IP addresses of Backups that have advertised themselves to this Layer 3 Switch by sending Hello messages. **Note**: Hello messages from Backups are disabled by default. You must enable the Hello messages on the Backup for the Backup to advertise itself to the current Master. See "Hello Messages" on page 22-4.

Table 22.4: CLI Display of VRRP or VRRPE Detailed Information (Continued)

This Field...	Displays...
backup router <ip-addr> expires in <time>	The IP addresses of Backups that have advertised themselves to this Master by sending Hello messages. The <time> value indicates how long before the Backup expires. A Backup expires if you disable the advertise backup option on the Backup or the Backup becomes unavailable. Otherwise, the Backup's next Hello message arrives before the Backup expires. The Hello message resets the expiration timer. An expired Backup does not necessarily affect the Master. However, if you have not disabled the advertise backup option on the Backup, then the expiration may indicate a problem with the Backup. **Note**: This field applies only when Hello messages are enabled on the Backups (using the advertise backup option).
next hello sent in <time>	How long until the Backup sends its next Hello message. **Note**: This field applies only when this Layer 3 Switch is the Master and the Backup is configured to send Hello messages (the advertise backup option is enabled).
master router <ip-addr> expires in <time>	The IP address of the Master and the amount of time until the Master's dead interval expires. If the Backup does not receive a Hello message from the Master by the time the interval expires, either the IP address listed for the Master will change to the IP address of the new Master, or this Layer 3 Switch itself will become the Master. **Note**: This field applies only when this Layer 3 Switch is a Backup.
track port	The interfaces that the VRID's interface is tracking. If the link for a tracked interface goes down, the VRRP or VRRPE priority of the VRID interface is changed, causing the devices to renegotiate for Master. **Note**: This field is displayed only if track interfaces are configured for this VRID.

Displaying Detailed Information for an Individual VRID

You can display information about the settings configured for a specified VRRP Virtual Router ID (VRID). For example, to display information about VRID 1:

```
FastIron SuperX Router(config)# show ip vrrp vrid 1
VRID 1
  Interface ethernet 3/11
  state initialize
  administrative-status disabled
  mode non-owner(backup)incomplete
  priority 12
  current priority 12
  track-priority 22
  hello-interval 1 sec
  dead-interval 0 sec
  current dead-interval 3.900 sec
  preempt-mode true
  advertise backup: disabled
```


Syntax: show ip vrrp vrid <num> [ethernet <num> | ve <num>]

The <num> parameter specifies the VRID.

The **ethernet** <num> | **ve** <num> specifies an interface on which the VRID is configured. If you specify an interface, VRID information is displayed for that interface only. Otherwise, information is displayed for all the interfaces on which the specified VRID is configured.

This display shows the following information.

Table 22.5: Output from the show ip vrrp vrid command

This Field...	Displays...
VRID	The specified VRID.
Interface	The interface on which VRRP is configured.
State	This Layer 3 Switch's VRRP state for the VRID. The state can be one of the following: • Init – The VRID is not enabled (activated). If the state remains Init after you activate the VRID, make sure that the VRID is also configured on the other routers and that the routers can communicate with each other. **Note**: If the state is Init and the mode is incomplete, make sure you have specified the IP address for the VRID. • Backup – This Layer 3 Switch is a Backup for the VRID. • Master – This Layer 3 Switch is the Master for the VRID.
priority	The configured VRRP priority of this Layer 3 Switch for the VRID.
current priority	The current VRRP priority of this Layer 3 Switch for the VRID.
track-priority	The new VRRP priority that the router receives for this VRID if the interface goes down
hello-interval	How often the Master router sends Hello messages to the Backups.
dead-interval	The configured number of seconds a Backup waits for a Hello message from the Master before determining that the Master is dead.
current dead-interval	The current Dead interval. The software automatically adds one-half second to the Dead interval value you enter.
preempt-mode	Whether the backup preempt mode is enabled. If the backup preempt mode is enabled, this field contains "true". If the mode is disabled, this field contains "false".
advertise backup	Whether Backup routers send Hello messages to the Master.

Displaying Statistics

To display statistics on most Foundry devices, enter a command such as the following at any level of the CLI:

```
FastIron SuperX Router(config-if-e1000-1/5-vrid-1)# show ip vrrp statistic

Interface ethernet 1/5
 rxed vrrp header error count = 0
 rxed vrrp auth error count = 0
 rxed vrrp auth passwd mismatch error count = 0
 rxed vrrp vrid not found error count = 0
 VRID 1
 rxed arp packet drop count = 0
 rxed ip packet drop count = 0
 rxed vrrp port mismatch count = 0
 rxed vrrp ip address mismatch count = 0
 rxed vrrp hello interval mismatch count = 0
 rxed vrrp priority zero from master count = 0
 rxed vrrp higher priority count = 0
 transitioned to master state count = 1
 transitioned to backup state count = 1
```

The same statistics are listed for VRRP and VRRPE.

Syntax: show ip vrrp brief | ethernet [<slotnum>/]<portnum> | ve <num> | statistic

Syntax: show ip vrrp-extended brief | ethernet [<slotnum>/]<portnum> | ve <num> | stat

The **brief** parameter displays summary information. See “Displaying Summary Information” on page 22-19.

If you specify a port, the <slotnum> parameter is required on chassis devices.

The <portnum> parameter specifies an Ethernet port. If you use this parameter, the command displays detailed VRRP or VRRPE information only for the specified port. See “Displaying Detailed Information” on page 22-20.

The **ve** <num> parameter specifies a virtual interface. If you use this parameter, the command displays detailed VRRP or VRRPE information only for the specified virtual interface. See “Displaying Detailed Information” on page 22-20.

The **statistic** parameter displays statistics. This parameter is required for displaying the statistics.

This display shows the following information.

Table 22.6: CLI Display of VRRP or VRRPE Statistics

This Field...	Displays...
Interface Statistics	
Interface	The interface on which VRRP or VRRPE is configured. If VRRP or VRRPE is configured on more than one interface, the display lists the statistics separately for each interface.
rxed vrrp header error count	The number of VRRP or VRRPE packets received by the interface that had a header error.
rxed vrrp auth error count	The number of VRRP or VRRPE packets received by the interface that had an authentication error.

Table 22.6: CLI Display of VRRP or VRRPE Statistics (Continued)

This Field...	Displays...
rxed vrrp auth passwd mismatch error count	The number of VRRP or VRRPE packets received by the interface that had a password value that does not match the password used by the interface for authentication.
rxed vrrp vrid not found error count	The number of VRRP or VRRPE packets received by the interface that contained a VRID that is not configured on this interface.
VRID Statistics	
rxed arp packet drop count	The number of ARP packets addressed to the VRID that were dropped.
rxed ip packet drop count	The number of IP packets addressed to the VRID that were dropped.
rxed vrrp port mismatch count	The number of packets received that did not match the configuration for the receiving interface.
rxed vrrp ip address mismatch count	The number of packets received that did not match the configured IP addresses.
rxed vrrp hello interval mismatch count	The number of packets received that did not match the configured Hello interval.
rxed vrrp priority zero from master count	The current Master has resigned.
rxed vrrp higher priority count	The number of VRRP or VRRPE packets received by the interface that had a higher backup priority for the VRID than this Layer 3 Switch's backup priority for the VRID.
transitioned to master state count	The number of times this Layer 3 Switch has changed from the backup state to the master state for the VRID.
transitioned to backup state count	The number of times this Layer 3 Switch has changed from the master state to the backup state for the VRID.

Clearing VRRP or VRRPE Statistics

Use the following methods to clear VRRP or VRRPE statistics.

To clear VRRP or VRRPE statistics, enter the following command at the Privileged EXEC level or any configuration level of the CLI:

```
Router1(config)# clear ip vrrp-stat
```

Syntax: clear ip vrrp-stat

Displaying CPU Utilization Statistics

You can display CPU utilization statistics for VRRP and other IP protocols.

To display CPU utilization statistics for the previous one-second, one-minute, five-minute, and fifteen-minute intervals, enter the following command at any level of the CLI:

```
FESX424 Router# show process cpu
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.01      0.03      0.09      0.22             9
BGP              0.04      0.06      0.08      0.14            13
GVRP             0.00      0.00      0.00      0.00             0
ICMP             0.00      0.00      0.00      0.00             0
IP               0.00      0.00      0.00      0.00             0
OSPF             0.00      0.00      0.00      0.00             0
RIP              0.00      0.00      0.00      0.00             0
STP              0.00      0.00      0.00      0.00             0
VRRP             0.03      0.07      0.09      0.10             8
```

If the software has been running less than 15 minutes (the maximum interval for utilization statistics), the command indicates how long the software has been running. Here is an example:

```
FESX424 Router# show process cpu
The system has only been up for 6 seconds.
Process Name   5Sec(%)   1Min(%)   5Min(%)   15Min(%)   Runtime(ms)
ARP              0.01      0.00      0.00      0.00             0
BGP              0.00      0.00      0.00      0.00             0
GVRP             0.00      0.00      0.00      0.00             0
ICMP             0.01      0.00      0.00      0.00             1
IP               0.00      0.00      0.00      0.00             0
OSPF             0.00      0.00      0.00      0.00             0
RIP              0.00      0.00      0.00      0.00             0
STP              0.00      0.00      0.00      0.00             0
VRRP             0.00      0.00      0.00      0.00             0
```

To display utilization statistics for a specific number of seconds, enter a command such as the following:

```
FESX424 Router# show process cpu 2
Statistics for last 1 sec and 80 ms
Process Name   Sec(%)   Time(ms)
ARP              0.00          0
BGP              0.00          0
GVRP             0.00          0
ICMP             0.01          1
IP               0.00          0
OSPF             0.00          0
RIP              0.00          0
STP              0.01          0
VRRP             0.00          0
```

When you specify how many seconds' worth of statistics you want to display, the software selects the sample that most closely matches the number of seconds you specified. In this example, statistics are requested for the previous two seconds. The closest sample available is actually for the previous 1 second plus 80 milliseconds.

Syntax: show process cpu [<num>]

The <num> parameter specifies the number of seconds and can be from 1 – 900. If you use this parameter, the command lists the usage statistics only for the specified number of seconds. If you do not use this parameter, the

command lists the usage statistics for the previous one-second, one-minute, five-minute, and fifteen-minute intervals.

Configuration Examples

The following sections contain the CLI commands for implementing the VRRP and VRRPE configurations shown in Figure 22.2 on page 22-3 and Figure 22.3 on page 22-7.

VRRP Example

To implement the VRRP configuration shown in Figure 22.2 on page 22-3, use the following method.

Configuring Router1

To configure VRRP Router1, enter the following commands:

```
Router1(config)# router vrrp
Router1(config)# inter e 1/6
Router1(config-if-1/6)# ip address 192.53.5.1
Router1(config-if-1/6)# ip vrrp vrid 1
Router1(config-if-1/6-vrid-1)# owner track-priority 20
Router1(config-if-1/6-vrid-1)# track-port ethernet 2/4
Router1(config-if-1/6-vrid-1)# ip-address 192.53.5.1
Router1(config-if-1/6-vrid-1)# activate
```

NOTE: When you configure the Master (Owner), the address you enter with the **ip-address** command must already be configured on the interface.

The **ip vrrp owner** command specifies that this router owns the IP address you are associating with the VRID. Because this router owns the IP address, this router is the default Master router and its VRRP priority is thus 255.

Configuring Router2

To configure Router2 in Figure 22.2 on page 22-3 after enabling VRRP, enter the following commands:

```
Router2(config)# router vrrp
Router2(config)# inter e 1/5
Router2(config-if-1/5)# ip address 192.53.5.3
Router2(config-if-1/5)# ip vrrp vrid 1
Router2(config-if-1/5-vrid-1)# backup priority 100 track-priority 19
Router2(config-if-1/5-vrid-1)# track-port ethernet 3/2
Router2(config-if-1/5-vrid-1)# ip-address 192.53.5.1
Router2(config-if-1/5-vrid-1)# activate
```

The **backup** command specifies that this router is a VRRP Backup for virtual router VRID1. The IP address entered with the **ip-address** command is the same IP address as the one entered when configuring Router1. In this case, the IP address cannot also exist on Router2, but the interface on which you are configuring the VRID Backup must have an IP address in the same sub-net. By entering the same IP address as the one associated with this VRID on the Owner, you are configuring the Backup to back up the address, but you are not duplicating the address.

NOTE: When you configure a Backup router, the router interface on which you are configuring the VRID must have a real IP address that is in the same sub-net as the address associated with the VRID by the Owner. However, the address cannot be the same.

The **priority** parameter establishes the router's VRRP priority in relation to the other VRRP router(s) in this virtual router. The **track-priority** parameter specifies the new VRRP priority that the router receives for this VRID if the interface goes down. See "Track Ports and Track Priority" on page 22-5.

The **activate** command activates the VRID configuration on this interface. The interface does not provide backup service for the virtual IP address until you activate the VRRP configuration.

Syntax: router vrrp

Syntax: ip vrrp vrid <vrid>

Syntax: owner [track-priority <value>]

Syntax: backup [priority <value>] [track-priority <value>]

Syntax: track-port ethernet [<slotnum>/]<portnum> | ve <num>

Syntax: ip-address <ip-addr>

Syntax: activate

VRRPE Example

To implement the VRRPE configuration shown in Figure 22.3 on page 22-7, use the following CLI method.

Configuring Router1

To configure VRRP Router1 in Figure 22.3 on page 22-7, enter the following commands:

```
Router1(config)# router vrrp-extended
Router1(config)# interface ethernet 1/6
Router1(config-if-1/6)# ip address 192.53.5.2/24
Router1(config-if-1/6)# ip vrrp-extended vrid 1
Router1(config-if-1/6-vrid-1)# backup priority 110 track-priority 20
Router1(config-if-1/6-vrid-1)# track-port ethernet 2/4
Router1(config-if-1/6-vrid-1)# ip-address 192.53.5.254
Router1(config-if-1/6-vrid-1)# activate
VRRP router 1 for this interface is activating
Router1(config-if-1/6-vrid-1)# exit
Router1(config)# interface ethernet 1/6
Router1(config-if-1/6)# ip vrrp-extended vrid 2
Router1(config-if-1/6-vrid-1)# backup priority 100 track-priority 20
Router1(config-if-1/6-vrid-1)# track-port ethernet 2/4
Router1(config-if-1/6-vrid-1)# ip-address 192.53.5.253
Router1(config-if-1/6-vrid-1)# activate
VRRP router 2 for this interface is activating
```

NOTE: The address you enter with the **ip-address** command cannot be the same as a real IP address configured on the interface.

Configuring Router2

To configure Router2, enter the following commands:

```
Router1(config)# router vrrp-extended
Router1(config)# interface ethernet 5/1
Router1(config-if-5/1)# ip address 192.53.5.3/24
Router1(config-if-5/1)# ip vrrp-extended vrid 1
Router1(config-if-5/1-vrid-1)# backup priority 100 track-priority 20
Router1(config-if-5/1-vrid-1)# track-port ethernet 3/2
Router1(config-if-5/1-vrid-1)# ip-address 192.53.5.254
Router1(config-if-5/1-vrid-1)# activate
VRRP router 1 for this interface is activating
Router1(config-if-5/1-vrid-1)# exit
```



```
Router1(config)# interface ethernet 5/1
Router1(config-if-5/1)# ip vrrp-extended vrid 2
Router1(config-if-5/1-vrid-1)# backup priority 110 track-priority 20
Router1(config-if-5/1-vrid-1)# track-port ethernet 2/4
Router1(config-if-5/1-vrid-1)# ip-address 192.53.5.253
Router1(config-if-5/1-vrid-1)# activate
VRRP router 2 for this interface is activating
```

The **backup** command specifies that this router is a VRRPE Backup for virtual router VRID1. The IP address entered with the **ip-address** command is the same IP address as the one entered when configuring Router1. In this case, the IP address cannot also exist on Router2, but the interface on which you are configuring the VRID Backup must have an IP address in the same sub-net. By entering the same IP address as the one associated with this VRID on the Owner, you are configuring the Backup to back up the address, but you are not duplicating the address.

NOTE: When you configure a Backup router, the router interface on which you are configuring the VRID must have a real IP address that is in the same sub-net as the address associated with the VRID by the Owner. However, the address cannot be the same.

The **priority** parameter establishes the router's VRRPE priority in relation to the other VRRPE router(s) in this virtual router. The **track-priority** parameter specifies the new VRRPE priority that the router receives for this VRID if the interface goes down. See "Track Ports and Track Priority" on page 22-5.

The **activate** command activates the VRID configuration on this interface. The interface does not provide backup service for the virtual IP address until you activate the VRRPE configuration. Alternatively, you can use the **enable** command. The **activate** and **enable** commands do the same thing.

Syntax: router vrrp-extended

Syntax: ip vrrp-extended vrid <vrid>

Syntax: backup [priority <value>] [track-priority <value>]

Syntax: track-port ethernet [<slotnum>/]<portnum> | ve <num>

Syntax: ip-address <ip-addr>

Syntax: activate

Chapter 23

Updating Software Images and Configuration Files

This chapter describes how to copy and save configuration files and software image files.

This chapter contains the following information:

Table 23.1: Chapter Contents

Overview

For easy software image management, all Foundry devices support the download and upload of software images between the flash modules on the devices and a Trivial File Transfer Protocol (TFTP) server on the network.

Foundry devices have two flash memory modules:

- ***Primary flash*** – The default local storage device for image files and configuration files.
- ***Secondary flash*** – A second flash storage device. You can use the secondary flash to store redundant images for additional booting reliability or to preserve one software image while testing another one.

Only one flash device is active at a time. By default, the primary image will become active upon reload.

You can update the software contained on a flash module using TFTP to copy the update image from a TFTP server onto the flash module. In addition, you can copy software images and configuration files from a flash module to a TFTP server.

NOTE: Foundry devices are TFTP clients but not TFTP servers. You must perform the TFTP transaction from the Foundry device. You cannot "put" a file onto the Foundry device using the interface of your TFTP server.

NOTE: If you are attempting to transfer a file using TFTP but have received an error message, see "Diagnostic Error Codes and Remedies for TFTP Transfers" on page 23-13.

Determining the Software Versions Installed and Running on a Device

Use the following methods to display the software versions running on the device and the versions installed in flash memory.

Determining the Flash Image Version Running on the Device

To determine the flash image version running on a device, enter the **show version** command at any level of the CLI. Some examples are shown below.

FESX, and FWSX Devices

To determine the flash image version running on a FESX, or FWSX device, enter the **show version** command at any level of the CLI. The following shows an example output.

```
FESX424 Router# show version
  SW: Version 03.0.00T53 Copyright (c) 1996-2002 Foundry Networks, Inc.
      Compiled on Mar 26 2003 at 13:50:31 labeled as FER03000
      (3089381 bytes) from Primary fer03000.bin
  HW: Stackable FES2402-PREM-ILP
==========================================================================
  330 MHz Power PC processor 8245 (version 129/1014) 66 MHz bus
  512 KB boot flash memory
16384 KB code flash memory
  128 MB DRAM
Monitor Option is on
The system uptime is 4 days 4 hours 8 minutes 33 seconds
The system : started=warm start
```

The version information is shown in bold type in this example.

- "03.0.00T53" indicates the flash code version number. The "T53" is used by Foundry for record keeping.
- "labeled as FER03000" indicates the flash code image label. The label indicates the image type and version and is especially useful if you change the image file name.
- "Primary fer03000.bin" indicates the flash code image file name that was loaded.

FSX Devices

To determine the flash image version running on a FSX, enter the **show version** command at any level of the CLI. The following is an example output.

```
FastIron SuperX Switch# show version

SW: Version 02.0.00T2e1 Copyright (c) 1996-2004 Foundry Networks, Inc.
      Compiled on Dec 20 2004 at 16:08:06 labeled as SXS02000
      (2294152 bytes) from Primary sxs02000.bin
      BootROM: Version 02.0.00T2e5 (FEv2)
  HW: Chassis FastIron SuperX
==========================================================================
SL 6: SX-F42XG    2-port 10G Module
      Serial #:   0000000000
      P-ASIC 10: type 01D1, rev 00
      P-ASIC 11: type 01D1, rev 00
==========================================================================
SL 8: SX-F424C    24-port Gig Copper Module
      Serial #: Non-exist
      P-ASIC 14: type 00D1, rev D1
      P-ASIC 15: type 00D1, rev D1
==========================================================================
SL 9: SX-F12GM1-4 12-port Management Module
      Serial #: Non-exist
      P-ASIC 16: type 00D1, rev D2
==========================================================================
  400 MHz Power PC processor 8245 (version 129/1014) 66 MHz bus
  512 KB boot flash memory
16384 KB code flash memory
  128 MB DRAM
Monitor Option is on
The system uptime is 26 seconds
The system : started=warm start   reloaded=by "reload"
```

The version information is shown in bold type in this example.

- “02.0.00Te1” indicates the flash code version number. The “Te1” is used by Foundry for record keeping.
- “labeled as SXS02000” indicates the flash code image label. The label indicates the image type and version and is especially useful if you change the image file name.
- “Primary sxs02000.bin” indicates the flash code image file name that was loaded.

Determining the Boot Image Version Running on the Device

To determine the boot image running on a device, enter the **show flash** command at any level of the CLI. The following shows an example output.

```
FESX424 Router> show flash
FCompressed Pri Code size = 3089381, Version 03.0.00b104Tc3 (fer03000.bin)
Compressed Sec Code size = 1730031, Version 03.0.00b92Tc1 (fes03000.bin)
Boot Monitor Image size = 87088, Version 02.00.00b6Tc4 (BLDR-Rev1a)
Code Flash Free Space = 11468800
```

The boot code version is shown in bold type.

Determining the Image Versions Installed in Flash Memory

Enter the **show flash** command to display the boot and flash images installed on the device. An example of the command's output is shown in "Determining the Boot Image Version Running on the Device".

- The "Compressed Pri Code size" line lists the flash code version installed in the primary flash area.
- The "Compressed Sec Code size" line lists the flash code version installed in the secondary flash area.
- The "Boot Monitor Image size" line lists the boot code version installed in flash memory. The device does not have separate primary and secondary flash areas for the boot image. The flash memory module contains only one boot image.

Image File Types

This section lists the boot and flash image file types supported on the FastIron family of switches and how to install them. For information about a specific version of code, see the release notes.

Table 23.2: Software Image Files

Product	Boot Image[a]	Flash Image
FESX	FEXZ*xxxxx*.bin	FEXS*xxxxx*.bin
FWSX	FWXZ*xxxxx*.bin	FWXS*xxxxx*.bin
FSX	SXZ*xxxxx*.bin	SXS*xxxxx*.bin (Layer 2) or SXL*xxxxx*.bin (Base Layer 3) or SXR*xxxxx*.bin (Full Layer 3)

a.These images are applicable to these devices only and are not interchangeable. For example, you cannot load FESX boot or flash images on a FSX device, and vice versa. Also, you cannot load other images, such as B2R or B2S, for BigIron devices, on the FastIron family of switches.

Upgrading Software

Use the following procedures to upgrade the software.

NOTE: This section does not describe how to upgrade a FESX or FSX base model to a premium (PREM) model. To perform this upgrade, you need an upgrade kit. Contact Foundry Networks for information.

Migrating to the New Release

Beginning with release 02.3.01, FESX and FSX devices share the same flash images. In releases prior to 02.3.01, FESX and FSX flash images were separate and were issued via separate software releases. Starting with release 02.3.01, the flash images for these devices were merged and are now issued in the same software release.

The new, combined flash images may create unique software upgrade circumstances for FESX and FSX devices. (FWSX devices are not affected by the software merge.) If your device is currently running software release 02.2.00 or later (FESX devices), or 02.2.01a or later (FSX devices), your device is not affected by the software merge. However, if your FESX or FSX device is running a release earlier than these versions, you must first upgrade the software on your device to FESX release 02.2.00 or later, or FSX release 02.2.01a or later, *before* loading the new software image. Earlier releases will not allow you to load the 02.3.01 or later software image.

To determine which software version is running on your device, use the **show version** command.

See the following sections for information on how to upgrade the software images on your device.

Upgrading from FESX pre-02.2.00 or FSX pre-02.2.01a to the New Release

If your device is running a software release earlier than FESX 02.2.00 or FSX 02.2.01a, you must first upgrade it to FESX 02.2.00 or later, or FSX 02.2.01a or later, before you can upgrade it to the new release. Follow the instructions, below.

1. Upgrade your device to software release FESX 02.2.00 or later, or FSX 02.2.01a or later. Follow the steps presented in "Upgrading Software" on page 23-4 and "Upgrading the Flash Code" on page 23-5. Make sure you reload the software after loading the flash code.
2. Upgrade your device to the new software release. Refer to one of the following sections:
   - FESX – "Upgrading from FESX 02.2.00 or later to the New Release" on page 23-5.
   - FSX – "Upgrading from FSX 02.2.01a or later to the New Release" on page 23-5.

Upgrading from FESX 02.2.00 or later to the New Release

1. Upgrade the boot code to the new version (FEXZ0*xxxx*.bin) using the steps presented in "Upgrading Software" on page 23-4.
2. Upgrade the flash code to the new version using the steps presented in "Upgrading the Flash Code" on page 23-5.

Upgrading from FSX 02.2.01a or later to the New Release

1. Upgrade the boot code to the new version (SXZ0*xxxx*.bin) using the steps presented in "Upgrading Software" on page 23-4.
2. Upgrade the flash code to the new version using the steps presented in "Upgrading the Flash Code" on page 23-5.

Upgrading the Boot Code

NOTE: If you are upgrading a FESX or FSX device, see "Migrating to the New Release" on page 23-4 before performing the steps in this section.

1. Place the new boot code on a TFTP server to which the Foundry device has access.
2. Enter the following command at the Privileged EXEC level of the CLI (example: `FESX448 Switch#`) to copy the boot code from the TFTP server into flash memory:
   - **copy tftp flash** <ip-addr> <image-file-name> **bootrom**
3. Verify that the code has been successfully copied by entering the following command at any level of the CLI:
   - **show flash**

   The output will display the compressed boot ROM code size and the boot code version.
4. Upgrade the flash code as instructed in the following section.

Upgrading the Flash Code

NOTE: If you are upgrading a FESX or FSX device, see "Migrating to the New Release" on page 23-4 before performing the steps in this section.

1. Place the new flash code on a TFTP server to which the Foundry device has access.
2. Enter the following command at the Privileged EXEC level of the CLI (example: `FESX448 Switch#`) to copy the flash code from the TFTP server into the flash memory:
   - **copy tftp flash** <ip-addr> <image-file-name> **primary** | **secondary**

3. Verify that the flash code has been successfully copied by entering the following command at any level of the CLI:
   - **show flash**
4. If the flash code version is correct, go to Step 5. Otherwise, go to Step 1.
5. Reload the software by entering one of the following commands:
   - **reload** (this command boots from the default boot source, which is the primary flash area by default)
   - **boot system flash primary | secondary**

Using SNMP to Upgrade Software

You can use a third-party SNMP management application such as HP OpenView to upgrade software on a Foundry device.

NOTE: The syntax shown in this section assumes that you have installed HP OpenView in the "/usr" directory.

NOTE: Foundry recommends that you make a backup copy of the startup-config file before you upgrade the software. If you need to run an older release, you will need to use the backup copy of the startup-config file.

1. Configure a read-write community string on the Foundry device, if one is not already configured. To configure a read-write community string, enter the following command from the global CONFIG level of the CLI:

   snmp-server community <string> **ro | rw**

   where <string> is the community string and can be up to 32 characters long.
2. On the Foundry device, enter the following command from the global CONFIG level of the CLI:

   no snmp-server pw-check

   This command disables password checking for SNMP set requests. If a third-party SNMP management application does not add a password to the password field when it sends SNMP set requests to a Foundry device, by default the Foundry device rejects the request.
3. From the command prompt in the UNIX shell, enter the following command:

   /usr/OV/bin/snmpset -c <rw-community-string> <fdry-ip-addr> **1.3.6.1.4.1.1991.1.1.2.1.5.0 ipaddress** <tftp-ip-addr> **1.3.6.1.4.1.1991.1.1.2.1.6.0 octetstringascii** <file-name> **1.3.6.1.4.1.1991.1.1.2.1.7.0 integer** <command-integer>

   where:

   <rw-community-string> is a read-write community string configured on the Foundry device.

   <fdry-ip-addr> is the Foundry device's IP address.

   <tftp-ip-addr> is the TFTP server's IP address.

   <file-name> is the image file name.

   <command-integer> is one of the following:

   20 – Download the flash code into the device's primary flash area.

   22 – Download the flash code into the device's secondary flash area.

Changing the Block Size for TFTP File Transfers

When you use TFTP to copy a file to or from a Foundry device, the device transfers the data in blocks of 8192 bytes by default. You can change the block size to one of the following if needed:

- 4096
- 2048
- 1024
- 512
- 256
- 128
- 64
- 32
- 16

To change the block size for TFTP file transfers, enter a command such as the following at the global CONFIG level of the CLI:

```
FESX424 Router(config)# flash 2047
set flash copy block size to 2048
```

Syntax: [no] flash <num>

The software rounds up the <num> value you enter to the next valid power of two, and displays the resulting value. In this example, the software rounds the value up to 2048.

NOTE: If the value you enter is one of the valid powers of two for this parameter, the software still rounds the value up to the next valid power of two. Thus, if you enter 2048, the software rounds the value up to 4096.

Rebooting

You can use boot commands to immediately initiate software boots from a software image stored in primary or secondary flash on a Foundry device or from a BootP or TFTP server. You can test new versions of code on a Foundry device or choose the preferred boot source from the console boot prompt without requiring a system reset.

NOTE: It is very important that you verify a successful TFTP transfer of the boot code ***before*** you reset the system. If the boot code is not transferred successfully but you try to reset the system, the system will not have the boot code with which to successfully boot.

By default, the Foundry device first attempts to boot from the image stored in its primary flash, then its secondary flash, and then from a TFTP server. You can modify this booting sequence at the global CONFIG level of the CLI using the **boot system…** command.

To initiate an immediate boot from the CLI, enter one of the **boot system…** commands as described in the *Foundry Switch and Router Command Line Interface Reference*.

Loading and Saving Configuration Files

For easy configuration management, all Foundry devices support both the download and upload of configuration files between the devices and a TFTP server on the network.

You can upload either the startup configuration file or the running configuration file to the TFTP server for backup and use in booting the system.

- ***Startup configuration file*** – This file contains the configuration information that is currently saved in flash.

To display this file, enter the **show configuration** command at any CLI prompt.

- ***Running configuration file*** – This file contains the configuration active in the system RAM but not yet saved to flash. These changes could represent a short-term requirement or general configuration change. To display this file, enter the **show running-config** or **write terminal** command at any CLI prompt.

Each device can have one startup configuration file and one running configuration file. The startup configuration file is shared by both flash modules. The running configuration file resides in DRAM.

When you load the startup-config file, the CLI parses the file three times.

1. During the first pass, the parser searches for **system-max** commands. A **system-max** command changes the size of statically configured memory.
2. During the second pass, the parser implements the **system-max** commands if present and also implements trunk configuration commands (**trunk** command) if present.
3. During the third pass, the parser implements the remaining commands.

Replacing the Startup Configuration with the Running Configuration

After you make configuration changes to the active system, you can save those changes by writing them to flash memory. When you write configuration changes to flash memory, you replace the startup configuration with the running configuration.

To replace the startup configuration with the running configuration, enter the following command at any Enable or CONFIG command prompt:

```
FESX424 Switch# write memory
```

Replacing the Running Configuration with the Startup Configuration

If you want to back out of the changes you have made to the running configuration and return to the startup configuration, enter the following command at the Privileged EXEC level of the CLI:

```
FESX424 Switch# reload
```

Logging Changes to the Startup-Config File

You can configure a Foundry device to generate a Syslog message when the startup-config file is changed. The trap is enabled by default.

The following Syslog message is generated when the startup-config file is changed:

```
startup-config was changed
```

If the startup-config file was modified by a valid user, the following Syslog message is generated:

`startup-config was changed by` <username>

To disable or re-enable Syslog messages when the startup-config file is changed, use the following command:

Syntax: [no] logging enable config-changed

Copying a Configuration File to or from a TFTP Server

To copy the startup-config or running-config file to or from a TFTP server, use one of the following methods.

NOTE: You can name the configuration file when you copy it to a TFTP server. However, when you copy a configuration file from the server to a Foundry device, the file is always copied as “startup-config” or “running-config”, depending on which type of file you saved to the server.

To initiate transfers of configuration files to or from a TFTP server using the CLI, enter one of the following commands:

- **copy startup-config tftp** <tftp-ip-addr> <filename> – Use this command to upload a copy of the startup configuration file from the Layer 2 Switch or Layer 3 Switch to a TFTP server.

- **copy running-config tftp** <tftp-ip-addr> <filename> – Use this command to upload a copy of the running configuration file from the Layer 2 Switch or Layer 3 Switch to a TFTP server.
- **copy tftp startup-config** <tftp-ip-addr> <filename> – Use this command to download a copy of the startup configuration file from a TFTP server to a Layer 2 Switch or Layer 3 Switch.

Dynamic Configuration Loading

You can load dynamic configuration commands (commands that do not require a reload to take effect) from a file on a TFTP server into a Foundry device's running-config. You can make configuration changes off-line, then load the changes directly into the device's running-config, without reloading the software.

Usage Considerations

- Use this feature only to load configuration information that does not require a software reload to take effect. For example, you cannot use this feature to change statically configured memory (**system-max** command) or to enter trunk group configuration information into the running-config.
- Do not use this feature if you have deleted a trunk group but have not yet placed the changes into effect by saving the configuration and then reloading. When you delete a trunk group, the command to configure the trunk group is removed from the device's running-config, but the trunk group remains active. To finish deleting a trunk group, save the configuration (to the startup-config file), then reload the software. After you reload the software, then you can load the configuration from the file.
- Do not load port configuration information for secondary ports in a trunk group. Since all ports in a trunk group use the port configuration settings of the primary port in the group, the software cannot implement the changes to the secondary port.

Preparing the Configuration File

A configuration file that you create must follow the same syntax rules as the startup-config file the device creates.

- The configuration file is a script containing CLI configuration commands. The CLI reacts to each command entered from the file in the same way the CLI reacts to the command if you enter it. For example, if the command results in an error message or a change to the CLI configuration level, the software responds by displaying the message or changing the CLI level.
- The software retains the running-config that is currently on the device, and changes the running-config only by adding new commands from the configuration file. If the running config already contains a command that is also in the configuration file you are loading, the CLI rejects the new command as a duplicate and displays an error message. For example, if the running-config already contains a a command that configures ACL 1, the software rejects ACL 1 in the configuration file, and displays a message that ACL 1 is already configured.
- The file can contain global CONFIG commands or configuration commands for interfaces, routing protocols, and so on. You cannot enter User EXEC or Privileged EXEC commands.
- The default CLI configuration level in a configuration file is the global CONFIG level. Thus, the first command in the file must be a global CONFIG command or " ! ". The ! (exclamation point) character means "return to the global CONFIG level".

NOTE: You can enter text following " ! " as a comment. However, the " !" is not a comment marker. It returns the CLI to the global configuration level.

NOTE: If you copy-and-paste a configuration into a management session, the CLI ignores the " ! " instead of changing the CLI to the global CONFIG level. As a result, you might get different results if you copy-and-paste a configuration instead of loading the configuration using TFTP.

- Make sure you enter each command at the correct CLI level. Since some commands have identical forms at both the global CONFIG level and individual configuration levels, if the CLI's response to the configuration file results in the CLI entering a configuration level you did not intend, then you can get unexpected results.

  For example, if a trunk group is active on the device, and the configuration file contains a command to disable

STP on one of the secondary ports in the trunk group, the CLI rejects the commands to enter the interface configuration level for the port and moves on to the next command in the file you are loading. If the next command is a spanning-tree command whose syntax is valid at the global CONFIG level as well as the interface configuration level, then the software applies the command globally. Here is an example:

The configuration file contains these commands:

```
interface ethernet 2
no spanning-tree
```

The CLI responds like this:

```
FESX424 Switch(config)# interface ethernet 2
Error - cannot configure secondary ports of a trunk
FESX424 Switch(config)# no spanning-tree
FESX424 Switch(config)#
```

- If the file contains commands that must be entered in a specific order, the commands must appear in the file in the required order. For example, if you want to use the file to replace an IP address on an interface, you must first remove the old address using "no" in front of the **ip address** command, then add the new address. Otherwise, the CLI displays an error message and does not implement the command. Here is an example:

  The configuration file contains these commands:

  ```
  interface ethernet 11
  ip address 10.10.10.69/24
  ```

  The running-config already has a command to add an address to port 11, so the CLI responds like this:

  ```
  FESX424 Switch(config)# interface ethernet 11
  FESX424 Switch(config-if-e1000-11)# ip add 10.10.10.69/24
  Error: can only assign one primary ip address per subnet
  FESX424 Switch(config-if-e1000-11)#
  ```

  To successfully replace the address, enter commands into the file as follows:

  ```
  interface ethernet 11
  no ip address 20.20.20.69/24
  ip address 10.10.10.69/24
  ```

  This time, the CLI accepts the command, and no error message is displayed:

  ```
  FESX424 Switch(config)# interface ethernet 11
  FESX424 Switch(config-if-e1000-11)# no ip add 20.20.20.69/24
  FESX424 Switch(config-if-e1000-111)# ip add 10.10.10.69/24
  FESX424 Switch(config-if-e1000-11)
  ```

- Always use the **end** command at the end of the file. The **end** command must appear on the last line of the file, by itself.

Loading the Configuration Information into the Running-Config

To load the file from a TFTP server, use either of the following commands:

- **copy tftp running-config** <ip-addr> <filename>
- **ncopy tftp** <ip-addr> <filename> **running-config**

Maximum File Sizes for Startup-Config File and Running-Config

Each Foundry device has a maximum allowable size for the running-config and the startup-config file. If you use TFTP to load additional information into a device's running-config or startup-config file, it is possible to exceed the maximum allowable size. If this occurs, you will not be able to save the configuration changes.

The maximum size for the running-config and the startup-config file is 64K each.

To determine the size of a Foundry device's running-config or startup-config file, copy it to a TFTP server, then use the directory services on the server to list the size of the copied file. To copy the running-config or startup-config file to a TFTP server, use one of the following commands.

- Commands to copy the running-config to a TFTP server:
  - **copy running-config tftp** <ip-addr> <filename>
  - **ncopy running-config tftp** <ip-addr> <from-name>
- Commands to copy the startup-config file to a TFTP server:
  - **copy startup-config tftp** <ip-addr> <filename>
  - **ncopy startup-config tftp** <ip-addr> <from-name>

Using SNMP to Save and Load Configuration Information

You can use a third-party SNMP management application such as HP OpenView to save and load a Foundry device's configuration. To save and load configuration information using HP OpenView, use the following procedure.

NOTE: The syntax shown in this section assumes that you have installed HP OpenView in the "/usr" directory.

1. Configure a read-write community string on the Foundry device, if one is not already configured. To configure a read-write community string, enter the following command from the global CONFIG level of the CLI:

   snmp-server community <string> **ro | rw**

   where <string> is the community string and can be up to 32 characters long.

2. On the Foundry device, enter the following command from the global CONFIG level of the CLI:

   no snmp-server pw-check

   This command disables password checking for SNMP set requests. If a third-party SNMP management application does not add a password to the password field when it sends SNMP set requests to a Foundry device, by default the Foundry device rejects the request.

3. From the command prompt in the UNIX shell, enter the following command:

   /usr/OV/bin/snmpset -c <rw-community-string> <fdry-ip-addr> **1.3.6.1.4.1.1991.1.1.2.1.5.0 ipaddress** <tftp-ip-addr> **1.3.6.1.4.1.1991.1.1.2.1.8.0 octetstringascii** <config-file-name> **1.3.6.1.4.1.1991.1.1.2.1.9.0 integer** <command-integer>

   where:

   <rw-community-string> is a read-write community string configured on the Foundry device.

   <fdry-ip-addr> is the Foundry device's IP address.

   <tftp-ip-addr> is the TFTP server's IP address.

   <config-file-name> is the configuration file name.

   <command-integer> is one of the following:

   20 – Upload the startup-config file from the Foundry device's flash memory to the TFTP server.

   21 – Download a startup-config file from a TFTP server to the Foundry device's flash memory.

   22 – Upload the running-config from the Foundry device's flash memory to the TFTP server.

23 – Download a configuration file from a TFTP server into the Foundry device's running-config.

NOTE: Command option **23** adds configuration information to the running-config on the device, and does not replace commands. If you want to replace configuration information in the device, use "no" forms of the configuration commands to remove the configuration information, then use configuration commands to create the configuration information you want. Follow the guidelines in "Dynamic Configuration Loading" on page 23-9.

Erasing Image and Configuration Files

To erase software images or configuration files, use the commands described below. These commands are valid at the Privileged EXEC level of the CLI.

- **erase flash primary** erases the image stored in primary flash of the system.
- **erase flash secondary** erases the image stored in secondary flash of the system.
- **erase startup-config** erases the configuration stored in the startup configuration file; however, the running configuration remains intact until system reboot.

Scheduling a System Reload

In addition to reloading the system manually, you can configure the Foundry device to reload itself at a specific time or after a specific amount of time has passed.

NOTE: The scheduled reload feature requires the system clock. You can use a Simple Network Time Protocol (SNTP) server to set the clock or you can set the device clock manually. See "Specifying a Simple Network Time Protocol (SNTP) Server" on page 3-8 or "Setting the System Clock" on page 3-10.

Reloading at a Specific Time

To schedule a system reload for a specific time, use the **reload at** command. For example, to schedule a system reload from the primary flash module for 6:00:00 AM, April 1, 2003, enter the following command at the global CONFIG level of the CLI:

```
FESX424 Switch# reload at 06:00:00 04-01-03
```

Syntax: reload at <hh:mm:ss> <mm-dd-yy> [primary | secondary]

<hh:mm:ss> is the hours, minutes, and seconds.

<mm-dd-yy> is the month, day, and year.

primary | secondary specifies whether the reload is to occur from the primary code flash module or the secondary code flash module. The default is **primary**.

Reloading after a Specific Amount of Time

To schedule a system reload to occur after a specific amount of time has passed on the system clock, use **reload after** command. For example, to schedule a system reload from the secondary flash one day and 12 hours later, enter the following command at the global CONFIG level of the CLI:

```
FESX424 Switch# reload after 01:12:00 secondary
```

Syntax: reload after <dd:hh:mm> [primary | secondary]

<dd:hh:mm> is the number of days, hours, and minutes.

primary | secondary specifies whether the reload is to occur from the primary code flash module or the secondary code flash module.

Displaying the Amount of Time Remaining Before a Scheduled Reload

To display how much time is remaining before a scheduled system reload, enter the following command from any level of the CLI:

```
FESX424 Switch# show reload
```

Canceling a Scheduled Reload

To cancel a scheduled system reload using the CLI, enter the following command at the global CONFIG level of the CLI:

```
FESX424 Switch# reload cancel
```

Diagnostic Error Codes and Remedies for TFTP Transfers

If an error occurs with a TFTP transfer to or from a Foundry Layer 2 Switch or Layer 3 Switch, one of the following error codes displays on the console.

Error code	Message	Explanation and action
1	Flash read preparation failed.	A flash error occurred during the download. Retry the download. If it fails again, contact customer support.
2	Flash read failed.	
3	Flash write preparation failed.	
4	Flash write failed.	
5	TFTP session timeout.	TFTP failed because of a time out. Check IP connectivity and make sure the TFTP server is running.
6	TFTP out of buffer space.	The file is larger than the amount of room on the device or TFTP server. If you are copying an image file to flash, first copy the other image to your TFTP server, then delete it from flash. (Use the **erase flash...** CLI command at the Privileged EXEC level to erase the image in the flash.) If you are copying a configuration file to flash, edit the file to remove unneeded information, then try again.
7	TFTP busy, only one TFTP session can be active.	Another TFTP transfer is active on another CLI session, or Web management session, or IronView Network Manager session. Wait, then retry the transfer.
8	File type check failed.	You accidentally attempted to copy the incorrect image code into the system. For example, you might have tried to copy a Chassis image into a Stackable device. Retry the transfer using the correct image.

Error code	Message	Explanation and action
16	TFTP remote - general error.	The TFTP configuration has an error. The specific error message describes the error. Correct the error, then retry the transfer.
17	TFTP remote - no such file.	
18	TFTP remote - access violation.	
19	TFTP remote - disk full.	
20	TFTP remote - illegal operation.	
21	TFTP remote - unknown transfer ID.	
22	TFTP remote - file already exists.	
23	TFTP remote - no such user.	

Appendix A
Using Syslog

This appendix describes how to display Syslog messages and how to configure the Syslog facility, and lists the Syslog messages that a Foundry device can display during standard operation.

NOTE: This appendix does not list Syslog messages that can be displayed when a debug option is enabled.

This chapter contains the topics listed in Table A.1.

Table A.1: Chapter Contents

Overview

A Foundry device's software can write syslog messages to provide information at the following severity levels:

- Emergencies
- Alerts
- Critical
- Errors
- Warnings
- Notifications
- Informational
- Debugging

The device writes the messages to a local buffer. The buffer can hold up to 1000 entries.

You also can specify the IP address or host name of up to six Syslog servers. When you specify a Syslog server, the Foundry device writes the messages both to the system log and to the Syslog server.

Using a Syslog server ensures that the messages remain available even after a system reload. The Foundry device's local Syslog buffer is cleared during a system reload or reboot, but the Syslog messages sent to the Syslog server remain on the server.

The Syslog service on a Syslog server receives logging messages from applications on the local host or from devices such as a Layer 2 Switch or Layer 3 Switch. Syslog adds a time stamp to each received message and directs messages to a log file. Most Unix workstations come with Syslog configured. Some third party vendor products also provide Syslog running on NT.

Syslog uses UDP port 514 and each Syslog message thus is sent with destination port 514. Each Syslog message is one line with Syslog message format. The message is embedded in the text portion of the Syslog format. There are several subfields in the format. Keywords are used to identify each subfield, and commas are delimiters. The subfield order is insensitive except that the text subfield should be the last field in the message. All the subfields are optional.

Displaying Syslog Messages

To display the Syslog messages in the device's local buffer, enter the following command at any level of the CLI:

```
FESX424 Router> show logging

Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Buffer logging: level ACDMEINW, 3 messages logged
    level code: A=alert C=critical D=debugging M=emergency E=error
                I=informational N=notification W=warning

Static Log Buffer:
Dec 15 19:04:14:A:Fan 1, fan on right connector, failed

Dynamic Log Buffer (50 entries):
Dec 15 18:46:17:I:Interface ethernet 4, state up
Dec 15 18:45:21:I:Bridge topology change, vlan 4095, interface 4, changed
state to forwarding
Dec 15 18:45:15:I:Warm start
```

For information about the Syslog configuration information, time stamps, and dynamic and static buffers, see "Displaying the Syslog Configuration" on page A-4.

Enabling Real-Time Display of Syslog Messages

By default, to view Syslog messages generated by a Foundry device, you need to display the Syslog buffer or the log on a Syslog server used by the Foundry device.

You can enable real-time display of Syslog messages on the management console. When you enable this feature, the software displays a Syslog message on the management console when the message is generated.

When you enable the feature, the software displays Syslog messages on the serial console when they occur. However, to enable display of real-time Syslog messages in Telnet or SSH sessions, you also must enable display within the individual sessions.

To enable real-time display of Syslog messages, enter the following command at the global CONFIG level of the CLI:

```
FESX424 Router(config)# logging console
```

Syntax: [no] logging console

This command enables the real-time display of Syslog messages on the serial console. You can enter this command from the serial console or a Telnet or SSH session.

To also enable the real-time display for a Telnet or SSH session, enter the following command from the Privileged EXEC level of the session:

```
telnet@FESX424 Router# terminal monitor
Syslog trace was turned ON
```

Syntax: terminal monitor

Notice that the CLI displays a message to indicate the status change for the feature. To disable the feature in the management session, enter the **terminal monitor** command again. The command toggles the feature on and off.

```
telnet@FESX424 Router# terminal monitor
Syslog trace was turned OFF
```

Here is an example of how the Syslog messages are displayed:

```
telnet@FESX424 Router# terminal monitor
Syslog trace was turned ON
SYSLOG: <9>FESX424 Router, Power supply 2, power supply on left connector, failed

SYSLOG: <14>FESX424 Router, Interface ethernet 6, state down

SYSLOG: <14>FESX424 Router, Interface ethernet 2, state up
```

Configuring the Syslog Service

The procedures in this section describe how to perform the following Syslog configuration tasks:

- Specify a Syslog server. You can configure the Foundry device to use up to six Syslog servers. (Use of a Syslog server is optional. The system can hold up to 100 Syslog messages in an internal buffer.)
- Change the level of messages the system logs.
- Change the number of messages the local Syslog buffer can hold.
- Display the Syslog configuration.
- Clear the local Syslog buffer.

Logging is enabled by default, with the following settings:

- Messages of all severity levels (Emergencies – Debugging) are logged.
- By default, up to 50 messages are retained in the local Syslog buffer. This can be changed.
- No Syslog server is specified.

Displaying the Syslog Configuration

To display the Syslog parameters currently in effect on a Foundry device, enter the following command from any level of the CLI:

```
FESX424 Router> show logging

Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Buffer logging: level ACDMEINW, 3 messages logged
    level code: A=alert C=critical D=debugging M=emergency E=error
                I=informational N=notification W=warning

Static Log Buffer:
Dec 15 19:04:14:A:Fan 1, fan on right connector, failed

Dynamic Log Buffer (50 entries):
Dec 15 18:46:17:I:Interface ethernet 1/4, state up
Dec 15 18:45:21:I:Bridge topology change, vlan 4095, interface 4, changed
state to forwarding
Dec 15 18:45:15:I:Warm start
```

Syntax: show logging

The Syslog display shows the following configuration information, in the rows above the log entries themselves.

Table A.2: CLI Display of Syslog Buffer Configuration

This Field...	Displays...
Syslog logging	The state (enabled or disabled) of the Syslog buffer.
messages dropped	The number of Syslog messages dropped due to user-configured filters. By default, the software logs messages for all Syslog levels. You can disable individual Syslog levels, in which case the software filters out messages at those levels. See “Disabling Logging of a Message Level” on page A-7. Each time the software filters out a Syslog message, this counter is incremented.
flushes	The number of times the Syslog buffer has been cleared by the **clear logging** command or equivalent Web management interface option. See “Clearing the Syslog Messages from the Local Buffer” on page A-9.
overruns	The number of times the dynamic log buffer has filled up and been cleared to hold new entries. For example, if the buffer is set for 100 entries, the 101st entry causes an overrun. After that, the 201st entry causes a second overrun.
level	The message levels that are enabled. Each letter represents a message type and is identified by the key (level code) below the value. If you disable logging of a message level, the code for that level is not listed.
messages logged	The total number of messages that have been logged since the software was loaded.
level code	The message levels represented by the one-letter codes.

Static and Dynamic Buffers

The software provides two separate buffers:

- Static – logs power supply failures, fan failures, and temperature warning or shutdown messages
- Dynamic – logs all other message types

In the static log, new messages replace older ones, so only the most recent message is displayed. For example, only the most recent temperature warning message will be present in the log. If multiple temperature warning messages are sent to the log, the latest one replaces the previous one. The static buffer is not configurable.

The message types that appear in the static buffer do not appear in the dynamic buffer. The dynamic buffer contains up to the maximum number of messages configured for the buffer (50 by default), then begins removing the oldest messages (at the bottom of the log) to make room for new ones.

The static and dynamic buffers are both displayed when you display the log.

```
FESX424 Router(config)# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Buffer logging: level ACDMEINW, 3 messages logged
    level code: A=alert C=critical D=debugging M=emergency E=error
          I=informational N=notification W=warning

Static Log Buffer:
Dec 15 19:04:14:A:Fan 1, fan on right connector, failed
Dec 15 19:00:14:A:Fan 2, fan on left connector, failed

Dynamic Log Buffer (50 entries):
Dec 15 18:46:17:I:Interface ethernet 4, state up
Dec 15 18:45:21:I:Bridge topology change, vlan 4095, interface 4, changed
state to forwarding
Dec 15 18:45:15:I:Warm start
```

Notice that the static buffer contains two separate messages for fan failures. Each message of each type has its own buffer. Thus, if you replace fan 1 but for some reason that fan also fails, the software replaces the first message about the failure of fan 1 with the newer message. The software does not overwrite the message for fan 2, unless the software sends a newer message for fan 2.

When you clear log entries, you can selectively clear the static or dynamic buffer, or you can clear both. For example, to clear only the dynamic buffer, enter the following command at the Privileged EXEC level:

```
FESX424 Router# clear logging dynamic-buffer
```

Syntax: clear logging [dynamic-buffer | static-buffer]

You can specify **dynamic-buffer** to clear the dynamic buffer or **static-buffer** to clear the static buffer. If you do not specify a buffer, both buffers are cleared.

Time Stamps

The contents of the time stamp differ depending on whether you have set the time and date on the onboard system clock.

- If you have set the time and date on the onboard system clock, the date and time are shown in the following format:

  mm dd hh:*mm*:*ss*

  where:

- *mm* – abbreviation for the name of the month
- *dd* – day
- *hh* – hours
- *mm* – minutes
- *ss* – seconds

For example, “Oct 15 17:38:03” means October 15 at 5:38 PM and 3 seconds.

- If you have not set the time and date on the onboard system clock, the time stamp shows the amount of time that has passed since the device was booted, in the following format:

  <*num*>d<*num*>h<*num*>m<*num*>s

  where:

  - <*num*>d – day
  - <*num*>h – hours
  - <*num*>m – minutes
  - <*num*>s – seconds

  For example, “188d1h01m00s” means the device had been running for 188 days, 11 hours, one minute, and zero seconds when the Syslog entry with this time stamp was generated.

Example of Syslog Messages on a Device Whose Onboard Clock Is Set

The example shows the format of messages on a device whose onboard system clock has been set. Each time stamp shows the month, the day, and the time of the system clock when the message was generated. For example, the system time when the most recent message (the one at the top) was generated was October 15 at 5:38 PM and 3 seconds.

```
FESX424 Router(config)# show log

Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Buffer logging: level ACDMEINW, 38 messages logged
    level code: A=alert C=critical D=debugging M=emergency E=error
                I=informational N=notification W=warning

Static Log Buffer:
Dec 15 19:04:14:A:Fan 1, fan on right connector, failed
Dec 15 19:00:14:A:Fan 2, fan on left connector, failed

Dynamic Log Buffer (50 entries):
Oct 15 17:38:03:warning:list 101 denied tcp 209.157.22.191(0)(Ethernet 18
0010.5a1f.77ed) -> 198.99.4.69(http), 1 event(s)

Oct 15 07:03:30:warning:list 101 denied tcp 209.157.22.26(0)(Ethernet 18
0010.5a1f.77ed) -> 198.99.4.69(http), 1 event(s)

Oct 15 06:58:30:warning:list 101 denied tcp 209.157.22.198(0)(Ethernet 18
0010.5a1f.77ed) -> 198.99.4.69(http), 1 event(s)
```

Example of Syslog Messages on a Device Whose Onboard Clock Is Not Set

The example shows the format of messages on a device whose onboard system clock is not set. Each time stamp shows the amount of time the device had been running when the message was generated. For example, the most

recent message, at the top of the list of messages, was generated when the device had been running for 21 days, seven hours, two minutes, and 40 seconds.

```
FESX424 Router(config)# show log

Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Buffer logging: level ACDMEINW, 38 messages logged
    level code: A=alert C=critical D=debugging M=emergency E=error
                I=informational N=notification W=warning

Static Log Buffer:

Dynamic Log Buffer (50 entries):
21d07h02m40s:warning:list 101 denied tcp 209.157.22.191(0)(Ethernet 4/18
0010.5a1f.77ed) -> 198.99.4.69(http), 1 event(s)

19d07h03m30s:warning:list 101 denied tcp 209.157.22.26(0)(Ethernet 4/18
0010.5a1f.77ed) -> 198.99.4.69(http), 1 event(s)

17d06h58m30s:warning:list 101 denied tcp 209.157.22.198(0)(Ethernet 4/18
0010.5a1f.77ed) -> 198.99.4.69(http), 1 event(s)
```

Disabling or Re-Enabling Syslog

Syslog is enabled by default. To disable it, enter the following command at the global CONFIG level:

```
FESX424 Router(config)# no logging on
```

Syntax: [no] logging on [<udp-port>]

The <udp-port> parameter specifies the application port used for the Syslog facility. The default is 514.

To re-enable logging, enter the following command:

```
FESX424 Router(config)# logging on
```

This command enables local Syslog logging with the following defaults:

- Messages of all severity levels (Emergencies – Debugging) are logged.
- Up to 50 messages are retained in the local Syslog buffer.
- No Syslog server is specified.

Specifying a Syslog Server

To specify a Syslog server, enter a command such as the following:

```
FESX424 Router(config)# logging host 10.0.0.99
```

Syntax: logging host <ip-addr> | <server-name>

Specifying an Additional Syslog Server

To specify an additional Syslog server, enter the **logging host** <ip-addr> command again, as in the following example. You can specify up to six Syslog servers.

```
FESX424 Router(config)# logging host 10.0.0.99
```

Syntax: logging host <ip-addr> | <server-name>

Disabling Logging of a Message Level

To change the message level, disable logging of specific message levels. You must disable the message levels on an individual basis.

For example, to disable logging of debugging and informational messages, enter the following commands:

```
FESX424 Router(config)# no logging buffered debugging
FESX424 Router(config)# no logging buffered informational
```

Syntax: [no] logging buffered <level> | <num-entries>

The <level> parameter can have one of the following values:

- alerts
- critical
- debugging
- emergencies
- errors
- informational
- notifications
- warnings

The commands in the example above change the log level to notification messages or higher. The software will not log informational or debugging messages. The changed message level also applies to the Syslog servers.

Changing the Number of Entries the Local Buffer Can Hold

You also can use the **logging buffered** command to change the number of entries the local Syslog buffer can store. For example:

```
FESX424 Router(config)# logging buffered 100
```

The default number of messages is 50. The value can be from 1 – 1000 on Layer 2 Switches and Layer 3 Switches. The change takes effect immediately and does not require you to reload the software.

NOTE: If you decrease the size of the buffer, the software clears the buffer before placing the change into effect. If you increase the size of the buffer, the software does not clear existing entries.

Changing the Log Facility

The Syslog daemon on the Syslog server uses a facility to determine where to log the messages from the Foundry device. The default facility for messages the Foundry device sends to the Syslog server is "user". You can change the facility using the following command.

NOTE: You can specify only one facility. If you configure the Foundry device to use two Syslog servers, the device uses the same facility on both servers.

```
FESX424 Router(config)# logging facility local0
```

Syntax: logging facility <facility-name>

The <facility-name> can be one of the following:

- kern – kernel messages
- user – random user-level messages
- mail – mail system
- daemon – system daemons
- auth – security/authorization messages
- syslog – messages generated internally by Syslog

- lpr – line printer subsystem
- news – netnews subsystem
- uucp – uucp subsystem
- sys9 – cron/at subsystem
- sys10 – reserved for system use
- sys11 – reserved for system use
- sys12 – reserved for system use
- sys13 – reserved for system use
- sys14 – reserved for system use
- cron – cron/at subsystem
- local0 – reserved for local use
- local1 – reserved for local use
- local2 – reserved for local use
- local3 – reserved for local use
- local4 – reserved for local use
- local5 – reserved for local use
- local6 – reserved for local use
- local7 – reserved for local use

Clearing the Syslog Messages from the Local Buffer

To clear the Syslog messages stored in the Foundry device's local buffer, enter the following command:

```
FESX424 Router# clear logging
```

Syntax: clear logging

Syslog Messages

Table A.3 lists all of the Syslog messages. Note that some of the messages apply only to Layer 3 Switches. The messages are listed by message level, in the following order:

- Emergencies (none)
- Alerts
- Critical
- Errors
- Warnings
- Notifications
- Informational
- Debugging

Table A.3: Foundry Syslog Messages

Message Level	Message	Explanation
Alert	<num-modules> modules and 1 power supply, need more power supply!!	Indicates that the chassis needs more power supplies to run the modules in the chassis. The <num-modules> parameter indicates the number of modules in the chassis.
Alert	Fan <num>, <location>, failed	A fan has failed. The <num> is the fan number. The <location> describes where the failed fan is in the chassis.
Alert	ISIS MEMORY USE EXCEEDED	IS-IS is requesting more memory than is available.
Alert	MAC Authentication failed for <mac-address> on <portnum>	RADIUS authentication was successful for the specified <mac-address> on the specified <portnum>; however, the VLAN returned in the RADIUS Access-Accept message did not refer to a valid VLAN or VLAN ID on the Foundry device. This is treated as an authentication failure.
Alert	MAC Authentication failed for <mac-address> on <portnum> (Invalid User)	RADIUS authentication failed for the specified <mac-address> on the specified <portnum> because the MAC address sent to the RADIUS server was not found in the RADIUS server's users database.
Alert	MAC Authentication failed for <mac-address> on <portnum> (No VLAN Info received from RADIUS server)	RADIUS authentication was successful for the specified <mac-address> on the specified <portnum>; however, dynamic VLAN assignment was enabled for the port, but the RADIUS Access-Accept message did not include VLAN information. This is treated as an authentication failure.
Alert	MAC Authentication failed for <mac-address> on <portnum> (Port is already in another radius given vlan)	RADIUS authentication was successful for the specified <mac-address> on the specified <portnum>; however, the RADIUS Access-Accept message specified a VLAN ID, although the port had previously been moved to a different RADIUS-assigned VLAN. This is treated as an authentication failure.
Alert	MAC Authentication failed for <mac-address> on <portnum> (RADIUS given vlan does not exist)	RADIUS authentication was successful for the specified <mac-address> on the specified <portnum>; however, the RADIUS Access-Accept message specified a VLAN that does not exist in the Foundry device's configuration. This is treated as an authentication failure.

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Alert	MAC Authentication failed for <mac-address> on <portnum> (RADIUS given VLAN does not match with TAGGED vlan)	Multi-device port authentication failed for the <mac-address> on a tagged port because the packet with this MAC address as the source was tagged with a VLAN ID different from the RADIUS-supplied VLAN ID.
Alert	Management module at slot <slot-num> state changed from <module-state> to <module-state>.	Indicates a state change in a management module. The <slot-num> indicates the chassis slot containing the module. The <module-state> can be one of the following: • active • standby • crashed • coming-up • unknown
Alert	OSPF LSA Overflow, LSA Type = <lsa-type>	Indicates an LSA database overflow. The <lsa-type> parameter indicates the type of LSA that experienced the overflow condition. The LSA type is one of the following: • 1 – Router • 2 – Network • 3 – Summary • 4 – Summary • 5 – External
Alert	OSPF Memory Overflow	OSPF has run out of memory.
Alert	Power supply <num>, <location>, failed	A power supply has failed. The <num> is the power supply number. The <location> describes where the failed power supply is in the chassis.
Alert	Temperature <degrees> C degrees, warning level <warn-degrees> C degrees, shutdown level <shutdown-degrees> C degrees	Indicates an overtemperature condition on the active module. The <degrees> value indicates the temperature of the module. The <warn-degrees> value is the warning threshold temperature configured for the module. The <shutdown-degrees> value is the shutdown temperature configured for the module.

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Critical	Authentication shut down <portnum> due to DOS attack	Denial of Service (DoS) attack protection was enabled for multi-device port authentication on the specified <portnum>, and the per-second rate of RADIUS authentication attempts for the port exceeded the configured limit. The Foundry device considers this to be a DoS attack and disables the port.
Debug	BGP4: Not enough memory available to run BGP4	The device could not start the BGP4 routing protocol because there is not enough memory available.
Debug	DOT1X: Not enough memory	There is not enough system memory for 802.1X authentication to take place. Contact Foundry Technical Support.
Error	No of prefixes received from BGP peer <ip-addr> exceeds maximum prefix-limit...shutdown	The Layer 3 Switch has received more than the specified maximum number of prefixes from the neighbor, and the Layer 3 Switch is therefore shutting down its BGP4 session with the neighbor.
Informational	<user-name> login to PRIVILEGED mode	A user has logged into the Privileged EXEC mode of the CLI. The <user-name> is the user name.
Informational	<user-name> login to USER EXEC mode	A user has logged into the USER EXEC mode of the CLI. The <user-name> is the user name.
Informational	<user-name> logout from PRIVILEGED mode	A user has logged out of Privileged EXEC mode of the CLI. The <user-name> is the user name.
Informational	<user-name> logout from USER EXEC mode	A user has logged out of the USER EXEC mode of the CLI. The <user-name> is the user name.
Informational	ACL <acl id> added | deleted | modified from console | telnet | ssh | web | snmp session	A user created, modified, deleted, or applied an ACL via the Web, SNMP, console, SSH, or Telnet session.
Informational	Bridge is new root, vlan <vlan-id>, root ID <root-id>	A Spanning Tree Protocol (STP) topology change has occurred, resulting in the Foundry device becoming the root bridge. The <vlan-id> is the ID of the VLAN in which the STP topology change occurred. The <root-id> is the STP bridge root ID.

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Informational	Bridge root changed, vlan <vlan-id>, new root ID <string>, root interface <portnum>	A Spanning Tree Protocol (STP) topology change has occurred. The <vlan-id> is the ID of the VLAN in which the STP topology change occurred. The <root-id> is the STP bridge root ID. The <portnum> is the number of the port connected to the new root bridge.
Informational	Bridge topology change, vlan <vlan-id>, interface <portnum>, changed state to <stp-state>	A Spanning Tree Protocol (STP) topology change has occurred on a port. The <vlan-id> is the ID of the VLAN in which the STP topology change occurred. The <portnum> is the port number. The <stp-state> is the new STP state and can be one of the following: • disabled • blocking • listening • learning • forwarding • unknown
Informational	Cold start	The device has been powered on.
Informational	DOT1X : port <portnum> - mac <mac address> Cannot apply an ACL or MAC filter on a port member of a VE (virtual interface)	The RADIUS server returned an IP ACL or MAC address filter, but the port is a member of a virtual routing interface (VE).
Informational	DOT1X : port <portnum> - mac <mac address> cannot remove inbound ACL	An error occurred while removing the inbound ACL.
Informational	DOT1X : port <portnum> - mac <mac address> cannot remove outbound ACL	An error occurred while removing the outbound ACL.
Informational	DOT1X : port <portnum> - mac <mac address> Downloading a MAC filter, but MAC filter have no effect on router port	The RADIUS server returned an MAC address filter, but the <portnum> is a router port (it has one or more IP addresses).
Informational	DOT1X : port <portnum> - mac <mac address> Downloading an IP ACL, but IP ACL have no effect on a switch port	The RADIUS server returned an IP ACL, but the <portnum> is a switch port (no IP address).
Informational	DOT1X : port <portnum> - mac <mac address> Error - could not add all MAC filters	The Foundry device was unable to implement the MAC address filters returned by the RADIUS server.
Informational	DOT1X : port <portnum> - mac <mac address> Invalid MAC filter ID - this ID doesn't exist	The MAC address filter ID returned by the RADIUS server does not exist in the Foundry device's configuration.

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Informational	DOT1X : port <portnum> - mac <mac address> Invalid MAC filter ID - this ID is user defined and cannot be used	The port was assigned a MAC address filter ID that had been dynamically created by another user.
Informational	DOT1X : port <portnum> - mac <mac address> is unauthorized because system resource is not enough or the invalid information to set the dynamic assigned IP ACLs or MAC address filters	802.1X authentication failed for the Client with the specified <mac address> on the specified <portnum> either due to insufficient system resources on the device, or due to invalid IP ACL or MAC address filter information returned by the RADIUS server.
Informational	DOT1X : port <portnum> - mac <mac address> Port is already bound with MAC filter	The RADIUS server returned a MAC address filter, but a MAC address filter had already been applied to the port.
Informational	DOT1X : port <portnum> - mac <mac address> This device doesn't support ACL with MAC Filtering on the same port	The RADIUS server returned a MAC address filter while an IP ACL was applied to the port, or returned an IP ACL while a MAC address filter was applied to the port.
Informational	DOT1X Port <portnum> is unauthorized because system resource is not enough or the invalid information to set the dynamic assigned IP ACLs or MAC address filters	802.1X authentication could not take place on the port. This happened because strict security mode was enabled and one of the following occurred: • Insufficient system resources were available on the device to apply an IP ACL or MAC address filter to the port • Invalid information was received from the RADIUS server (for example, the Filter-ID attribute did not refer to an existing IP ACL or MAC address filter)
Informational	DOT1X: Port <portnum> currently used vlan-id changes to <vlan-id> due to dot1x-RADIUS vlan assignment	A user has completed 802.1X authentication. The profile received from the RADIUS server specifies a VLAN ID for the user. The port to which the user is connected has been moved to the VLAN indicated by <vlan-id>.
Informational	DOT1X: Port <portnum> currently used vlan-id is set back to port default vlan-id <vlan-id>	The user connected to <portnum> has disconnected, causing the port to be moved back into its default VLAN, <vlan-id>.
Informational	DOT1X: Port <portnum>, AuthControlledPortStatus change: authorized	The status of the interface's controlled port has changed from unauthorized to authorized.
Informational	DOT1X: Port <portnum>, AuthControlledPortStatus change: unauthorized	The status of the interface's controlled port has changed from authorized to unauthorized.

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Informational	Enable super | port-config | read-only password deleted | added | modified from console | telnet | ssh | web | snmp OR Line password deleted | added | modified from console | telnet | ssh | web | snmp	A user created, re-configured, or deleted an Enable or Line password via the Web, SNMP, console, SSH, or Telnet session.
Informational	Interface <portnum>, line protocol down	The line protocol on a port has gone down. The <portnum> is the port number.
Informational	Interface <portnum>, line protocol up	The line protocol on a port has come up. The <portnum> is the port number.
Informational	Interface <portnum>, state down	A port has gone down. The <portnum> is the port number.
Informational	Interface <portnum>, state up	A port has come up. The <portnum> is the port number.
Informational	MAC Filter added | deleted | modified from console | telnet | ssh | web | snmp session filter id = <MAC filter ID>, src mac = <Source MAC address> | any, dst mac = <Destination MAC address> | any	A user created, modified, deleted, or applied this MAC filter via the Web, SNMP, console, SSH, or Telnet session.
Informational	Port <p> priority changed to <n>	A port's priority has changed.
Informational	Port <portnum>, srcip-security max-ipaddr-per-int reached.Last IP=<ipaddr>	The address limit specified by the **srcip-security max-ipaddr-per-interface** command has been reached for the port.
Informational	Port <portnum>, srcip-security max-ipaddr-per-int reached.Last IP=<ipaddr>	The address limit specified by the **srcip-security max-ipaddr-per-interface** command has been reached for the port.
Informational	Security: console login by <username> to USER | PRIVILEGE EXEC mode	The specified user logged into the device console into the specified EXEC mode.
Informational	Security: console logout by <username>	The specified user logged out of the device console.
Informational	Security: telnet | SSH login by <username> from src IP <ip-address>, src MAC <mac-address> to USER | PRIVILEGE EXEC mode	The specified user logged into the device using Telnet or SSH from the specified IP address and/or MAC address. The user logged into the specified EXEC mode.
Informational	Security: telnet | SSH logout by <username> from src IP <ip-address>, src MAC <mac-address> to USER | PRIVILEGE EXEC mode	The specified user logged out of the device. The user was using Telnet or SSH to access the device from the specified IP address and/or MAC address. The user logged out of the specified EXEC mode.

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Informational	SNMP read-only community | read-write community | contact | location | user | group | view | engineId | trap [host] [<value -str>] deleted | added | modified from console | telnet | ssh | web | snmp session	A user made SNMP configuration changes via the Web, SNMP, console, SSH, or Telnet session. [<value-str>] does not appear in the message if SNMP **community** or **engineId** is specified.
Informational	SNMP Auth. failure, intruder IP: <ip-addr>	A user has tried to open a management session with the device using an invalid SNMP community string. The <ip-addr> is the IP address of the host that sent the invalid community string.
Informational	SSH | telnet server enabled | disabled from console | telnet | ssh | web | snmp session [by user <username>]	A user enabled or disabled an SSH or Telnet session, or changed the SSH enable/disable configuration via the Web, SNMP, console, SSH, or Telnet session.
Informational	startup-config was changed or startup-config was changed by <user-name>	A configuration change was saved to the startup-config file. The <user-name> is the user's ID, if they entered a user ID to log in.
Informational	Syslog server <IP-address> deleted | added | modified from console | telnet | ssh | web | snmp OR Syslog operation enabled | disabled from console | telnet | ssh | web | snmp	A user made Syslog configuration changes to the specified Syslog server address, or enabled or disabled a Syslog operation via the Web, SNMP, console, SSH, or Telnet session.
Informational	System: Fan speed changed automatically to <fan speed>	The system automatically changed the fan speed to the speed specified in this message.
Informational	telnet | SSH | web access [by <username>] from src IP <source ip address>, src MAC <source MAC address> rejected, <n> attempt(s)	There were failed web, SSH, or Telnet login access attempts from the specified source IP and MAC address. • [by <user> <username>] does not appear if **telnet** or **SSH** clients are specified. • <n> is the number of times this SNMP trap occurred in the last five minutes, or other configured number of minutes.
Informational	Trunk group (<ports>) created by 802.3ad link-aggregation module.	802.3ad link aggregation is configured on the device, and the feature has dynamically created a trunk group (aggregate link). The <ports> is a list of the ports that were aggregated to make the trunk group.

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Informational	user <username> added | deleted | modified from console | telnet | ssh | web | snmp	A user created, modified, or deleted a local user account via the Web, SNMP, console, SSH, or Telnet session.
Informational	vlan <vlan id> added | deleted | modified from console | telnet | ssh | web | snmp session	A user created, modified, or deleted a VLAN via the Web, SNMP, console, SSH, or Telnet session.
Informational	Warm start	The system software (flash code) has been reloaded.
Informational	vlan <vlan-id> Bridge is RootBridge <mac-address> (MgmtPriChg)	802.1W changed the current bridge to be the root bridge of the given topology due to administrative change in bridge priority.
Informational	vlan <vlan-id> Bridge is RootBridge <mac-address> (MsgAgeExpiry)	The message age expired on the Root port so 802.1W changed the current bridge to be the root bridge of the topology.
Informational	vlan <vlan-id> interface <portnum> Bridge TC Event (DOT1wTransition)	802.1W recognized a topology change event in the bridge. The topology change event is the forwarding action that started on a non-edge Designated port or Root port.
Informational	vlan <vlan-id> interface <portnum> STP state -> <state> (DOT1wTransition)	802.1W changed the state of a port to a new state: forwarding, learning, blocking. If the port changes to blocking, the bridge port is in discarding state.
Informational	vlan <vlan-id> New RootBridge <mac-address> RootPort <portnum> (BpduRcvd)	802.1W selected a new root bridge as a result of the BPDUs received on a bridge port.
Informational	vlan <vlan-id> New RootPort <portnum> (RootSelection)	802.1W changed the port's role to Root port, using the root selection computation.
Notification	ACL exceed max DMA L4 cam resource, using flow based ACL instead	The port does not have enough Layer 4 CAM entries for the ACL. To correct this condition, allocate more Layer 4 CAM entries. To allocate more Layer 4 CAM entries, enter the following command at the CLI configuration level for the interface: **ip access-group max-l4-cam** <num>
Notification	ACL insufficient L4 cam resource, using flow based ACL instead	The port does not have a large enough CAM partition for the ACLs.
Notification	ACL insufficient L4 session resource, using flow based ACL instead	The device does not have enough Layer 4 session entries. To correct this condition, allocate more memory for sessions. To allocate more memory, enter the following command at the global CONFIG level of the CLI interface: **system-max session-limit** <num>

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Notification	ACL port fragment packet inspect rate <rate> exceeded on port <portnum>	The fragment rate allowed on an individual interface has been exceeded. The <rate> indicates the maximum rate allowed. The <portnum> indicates the port. This message can occur if fragment thottling is enabled.
Notification	ACL system fragment packet inspect rate <rate> exceeded	The fragment rate allowed on the device has been exceeded. The <rate> indicates the maximum rate allowed. This message can occur if fragment thottling is enabled.
Notification	Authentication Disabled on <portnum>	The multi-device port authentication feature was disabled on the on the specified <portnum>.
Notification	Authentication Enabled on <portnum>	The multi-device port authentication feature was enabled on the on the specified <portnum>.
Notification	BGP Peer <ip-addr> DOWN (IDLE)	Indicates that a BGP4 neighbor has gone down. The <ip-addr> is the IP address of the neighbor's BGP4 interface with the Foundry device.
Notification	BGP Peer <ip-addr> UP (ESTABLISHED)	Indicates that a BGP4 neighbor has come up. The <ip-addr> is the IP address of the neighbor's BGP4 interface with the Foundry device.
Notification	DOT1X issues software but not physical port down indication of Port <portnum> to other software applications	The device has indicated that the specified is no longer authorized, but the actual port may still be active.
Notification	DOT1X issues software but not physical port up indication of Port <portnum> to other software applications	The device has indicated that the specified port has been authenticated, but the actual port may not be active.
Notification	ISIS ENTERED INTO OVERLOAD STATE	The Layer 3 Switch has set the overload bit to on (1), indicating that the Layer 3 Switch's IS-IS resources are overloaded.
Notification	ISIS EXITING FROM OVERLOAD STATE	The Layer 3 Switch has set the overload bit to off (0), indicating that the Layer 3 Switch's IS-IS resources are no longer overloaded.

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Notification	ISIS L1 ADJACENCY DOWN <system-id> on circuit <circuit-id>	The Layer 3 Switch's adjacency with this Level-1 IS has gone down. The <system-id> is the system ID of the IS. The <circuit-id> is the ID of the circuit over which the adjacency was established.
Notification	ISIS L1 ADJACENCY UP <system-id> on circuit <circuit-id>	The Layer 3 Switch's adjacency with this Level-1 IS has come up. The <system-id> is the system ID of the IS. The <circuit-id> is the ID of the circuit over which the adjacency was established.
Notification	ISIS L2 ADJACENCY DOWN <system-id> on circuit <circuit-id>	The Layer 3 Switch's adjacency with this Level-2 IS has gone down. The <system-id> is the system ID of the IS. The <circuit-id> is the ID of the circuit over which the adjacency was established.
Notification	ISIS L2 ADJACENCY UP <system-id> on circuit <circuit-id>	The Layer 3 Switch's adjacency with this Level-2 IS has come up. The <system-id> is the system ID of the IS. The <circuit-id> is the ID of the circuit over which the adjacency was established.
Notification	Local ICMP exceeds <burst-max> burst packets, stopping for <lockup> seconds!!	The number of ICMP packets exceeds the <burst-max> threshold set by the **ip icmp burst** command. The Foundry device may be the victim of a Denial of Service (DoS) attack. All ICMP packets will be dropped for the number of seconds specified by the <lockup> value. When the lockup period expires, the packet counter is reset and measurement is restarted.
Notification	Local TCP exceeds <burst-max> burst packets, stopping for <lockup> seconds!!	The number of TCP SYN packets exceeds the <burst-max> threshold set by the **ip tcp burst** command. The Foundry device may be the victim of a TCP SYN DoS attack. All TCP SYN packets will be dropped for the number of seconds specified by the <lockup> value. When the lockup period expires, the packet counter is reset and measurement is restarted.

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Notification	Local TCP exceeds <num> burst packets, stopping for <num> seconds!!	Threshold parameters for local TCP traffic on the device have been configured, and the maximum burst size for TCP packets has been exceeded. The first <num> is the maximum burst size (maximum number of packets allowed). The second <num> is the number of seconds during which additional TCP packets will be blocked on the device. **Note**: This message can occur in response to an attempted TCP SYN attack.
Notification	MAC Authentication succeeded for <mac-address> on <portnum>	RADIUS authentication was successful for the specified <mac-address> on the specified <portnum>.
Notification	Module was inserted to slot <slot-num>	Indicates that a module was inserted into a chassis slot. The <slot-num> is the number of the chassis slot into which the module was inserted.
Notification	Module was removed from slot <slot-num>	Indicates that a module was removed from a chassis slot. The <slot-num> is the number of the chassis slot from which the module was removed.
Notification	OSPF interface state changed, rid <router-id>, intf addr <ip-addr>, state <ospf-state>	Indicates that the state of an OSPF interface has changed. The <router-id> is the router ID of the Foundry device. The <ip-addr> is the interface's IP address. The <ospf-state> indicates the state to which the interface has changed and can be one of the following: • down • loopback • waiting • point-to-point • designated router • backup designated router • other designated router • unknown

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Notification	OSPF intf authen failure, rid <router-id>, intf addr <ip-addr>, pkt src addr <src-ip-addr>, error type <error-type>, pkt type <pkt-type>	Indicates that an OSPF interface authentication failure has occurred. The <router-id> is the router ID of the Foundry device. The <ip-addr> is the IP address of the interface on the Foundry device. The <src-ip-addr> is the IP address of the interface from which the Foundry device received the authentication failure. The <error-type> can be one of the following: • bad version • area mismatch • unknown NBMA neighbor • unknown virtual neighbor • authentication type mismatch • authentication failure • network mask mismatch • hello interval mismatch • dead interval mismatch • option mismatch • unknown The <packet-type> can be one of the following: • hello • database description • link state request • link state update • link state ack • unknown

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Notification	OSPF intf config error, rid <router-id>, intf addr <ip-addr>, pkt src addr <src-ip-addr>, error type <error-type>, pkt type <pkt-type>	Indicates that an OSPF interface configuration error has occurred. The <router-id> is the router ID of the Foundry device. The <ip-addr> is the IP address of the interface on the Foundry device. The <src-ip-addr> is the IP address of the interface from which the Foundry device received the error packet. The <error-type> can be one of the following: • bad version • area mismatch • unknown NBMA neighbor • unknown virtual neighbor • authentication type mismatch • authentication failure • network mask mismatch • hello interval mismatch • dead interval mismatch • option mismatch • unknown The <packet-type> can be one of the following: • hello • database description • link state request • link state update • link state ack • unknown

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Notification	OSPF intf rcvd bad pkt, rid <router-id>, intf addr <ip-addr>, pkt src addr <src-ip-addr>, pkt type <pkt-type>	Indicates that an OSPF interface received a bad packet. The <router-id> is the router ID of the Foundry device. The <ip-addr> is the IP address of the interface on the Foundry device. The <src-ip-addr> is the IP address of the interface from which the Foundry device received the authentication failure. The <packet-type> can be one of the following: • hello • database description • link state request • link state update • link state ack • unknown
Notification	OSPF intf rcvd bad pkt: Bad Checksum, rid <ip-addr>, intf addr <ip-addr>, pkt size <num>, checksum <num>, pkt src addr <ip-addr>, pkt type <type>	The device received an OSPF packet that had an invalid checksum. The rid <ip-addr> is Foundry device's router ID. The intf addr <ip-addr> is the IP address of the Foundry interface that received the packet. The pkt size <num> is the number of bytes in the packet. The checksum <num> is the checksum value for the packet. The pkt src addr <ip-addr> is the IP address of the neighbor that sent the packet. The pkt type <type> is the OSPF packet type and can be one of the following: • hello • database description • link state request • link state update • link state acknowledgement • unknown (indicates an invalid packet type)

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Notification	OSPF intf rcvd bad pkt: Bad Packet type, rid <ip-addr>, intf addr <ip-addr>, pkt size <num>, checksum <num>, pkt src addr <ip-addr>, pkt type <type>	The device received an OSPF packet with an invalid type. The parameters are the same as for the Bad Checksum message. The pkt type <type> value is “unknown”, indicating that the packet type is invalid.
Notification	OSPF intf rcvd bad pkt: Invalid packet size, rid <ip-addr>, intf addr <ip-addr>, pkt size <num>, checksum <num>, pkt src addr <ip-addr>, pkt type <type>	The device received an OSPF packet with an invalid packet size. The parameters are the same as for the Bad Checksum message.
Notification	OSPF intf rcvd bad pkt: Unable to find associated neighbor, rid <ip-addr>, intf addr <ip-addr>, pkt size <num>, checksum <num>, pkt src addr <ip-addr>, pkt type <type>	The neighbor IP address in the packet is not on the Foundry device’s list of OSPF neighbors. The parameters are the same as for the Bad Checksum message.
Notification	OSPF intf retransmit, rid <router-id>, intf addr <ip-addr>, nbr rid <nbr-router-id>, pkt type is <pkt-type>, LSA type <lsa-type>, LSA id <lsa-id>, LSA rid <lsa-router-id>	An OSPF interface on the Foundry device has retransmitted a Link State Advertisement (LSA). The <router-id> is the router ID of the Foundry device. The <ip-addr> is the IP address of the interface on the Foundry device. The <nbr-router-id> is the router ID of the neighbor router. The <packet-type> can be one of the following: • hello • database description • link state request • link state update • link state ack • unknown The <lsa-type> is the type of LSA. The <lsa-id> is the LSA ID. The <lsa-router-id> is the LSA router ID.
Notification	OSPF LSDB approaching overflow, rid <router-id>, limit <num>	The software is close to an LSDB condition. The <router-id> is the router ID of the Foundry device. The <num> is the number of LSAs.

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Notification	OSPF LSDB overflow, rid <router-id>, limit <num>	A Link State Database Overflow (LSDB) condition has occurred. The <router-id> is the router ID of the Foundry device. The <num> is the number of LSAs.
Notification	OSPF max age LSA, rid <router-id>, area <area-id>, LSA type <lsa-type>, LSA id <lsa-id>, LSA rid <lsa-router-id>	An LSA has reached its maximum age. The <router-id> is the router ID of the Foundry device. The <area-id> is the OSPF area. The <lsa-type> is the type of LSA. The <lsa-id> is the LSA ID. The <lsa-router-id> is the LSA router ID.
Notification	OSPF nbr state changed, rid <router-id>, nbr addr <ip-addr>, nbr rid <nbr-router-Id>, state <ospf-state>	Indicates that the state of an OSPF neighbor has changed. The <router-id> is the router ID of the Foundry device. The <ip-addr> is the IP address of the neighbor. The <nbr-router-id> is the router ID of the neighbor. The <ospf-state> indicates the state to which the interface has changed and can be one of the following: • down • attempt • initializing • 2-way • exchange start • exchange • loading • full • unknown

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Notification	OSPF originate LSA, rid <router-id>, area <area-id>, LSA type <lsa-type>, LSA id <lsa-id>, LSA router id <lsa-router-id>	An OSPF interface has originated an LSA. The <router-id> is the router ID of the Foundry device. The <area-id> is the OSPF area. The <lsa-type> is the type of LSA. The <lsa-id> is the LSA ID. The <lsa-router-id> is the LSA router ID.
Notification	OSPF virtual intf authen failure, rid <router-id>, intf addr <ip-addr>, pkt src addr <src-ip-addr>, error type <error-type>, pkt type <pkt-type>	Indicates that an OSPF virtual routing interface authentication failure has occurred. The <router-id> is the router ID of the Foundry device. The <ip-addr> is the IP address of the interface on the Foundry device. The <src-ip-addr> is the IP address of the interface from which the Foundry device received the authentication failure. The <error-type> can be one of the following: • bad version • area mismatch • unknown NBMA neighbor • unknown virtual neighbor • authentication type mismatch • authentication failure • network mask mismatch • hello interval mismatch • dead interval mismatch • option mismatch • unknown The <packet-type> can be one of the following: • hello • database description • link state request • link state update • link state ack • unknown

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Notification	OSPF virtual intf config error, rid <router-id>, intf addr <ip-addr>, pkt src addr <src-ip-addr>, error type <error-type>, pkt type <pkt-type>	Indicates that an OSPF virtual routing interface configuration error has occurred. The <router-id> is the router ID of the Foundry device. The <ip-addr> is the IP address of the interface on the Foundry device. The <src-ip-addr> is the IP address of the interface from which the Foundry device received the error packet. The <error-type> can be one of the following: • bad version • area mismatch • unknown NBMA neighbor • unknown virtual neighbor • authentication type mismatch • authentication failure • network mask mismatch • hello interval mismatch • dead interval mismatch • option mismatch • unknown The <packet-type> can be one of the following: • hello • database description • link state request • link state update • link state ack • unknown

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Notification	OSPF virtual intf rcvd bad pkt, rid <router-id>, intf addr <ip-addr>, pkt src addr <src-ip-addr>, pkt type <pkt-type>	Indicates that an OSPF interface received a bad packet. The <router-id> is the router ID of the Foundry device. The <ip-addr> is the IP address of the interface on the Foundry device. The <src-ip-addr> is the IP address of the interface from which the Foundry device received the authentication failure. The <packet-type> can be one of the following: • hello • database description • link state request • link state update • link state ack • unknown
Notification	OSPF virtual intf retransmit, rid <router-id>, intf addr <ip-addr>, nbr rid <nbr-router-id>, pkt type is <pkt-type>, LSA type <lsa-type>, LSA id <lsa-id>, LSA rid <lsa-router-id>	An OSPF interface on the Foundry device has retransmitted a Link State Advertisement (LSA). The <router-id> is the router ID of the Foundry device. The <ip-addr> is the IP address of the interface on the Foundry device. The <nbr-router-id> is the router ID of the neighbor router. The <packet-type> can be one of the following: • hello • database description • link state request • link state update • link state ack • unknown The <lsa-type> is the type of LSA. The <lsa-id> is the LSA ID. The <lsa-router-id> is the LSA router ID.

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Notification	OSPF virtual intf state changed, rid <router-id>, area <area-id>, nbr <ip-addr>, state <ospf-state>	Indicates that the state of an OSPF virtual routing interface has changed. The <router-id> is the router ID of the router the interface is on. The <area-id> is the area the interface is in. The <ip-addr> is the IP address of the OSPF neighbor. The <ospf-state> indicates the state to which the interface has changed and can be one of the following: • down • loopback • waiting • point-to-point • designated router • backup designated router • other designated router • unknown
Notification	OSPF virtual nbr state changed, rid <router-id>, nbr addr <ip-addr>, nbr rid <nbr-router-id>, state <ospf-state>	Indicates that the state of an OSPF virtual neighbor has changed. The <router-id> is the router ID of the Foundry device. The <ip-addr> is the IP address of the neighbor. The <nbr-router-id> is the router ID of the neighbor. The <ospf-state> indicates the state to which the interface has changed and can be one of the following: • down • attempt • initializing • 2-way • exchange start • exchange • loading • full • unknown

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Notification	Transit ICMP in interface <portnum> exceeds <num> burst packets, stopping for <num> seconds!!	Threshold parameters for ICMP transit (through) traffic have been configured on an interface, and the maximum burst size for ICMP packets on the interface has been exceeded. The <portnum> is the port number. The first <num> is the maximum burst size (maximum number of packets allowed). The second <num> is the number of seconds during which additional ICMP packets will be blocked on the interface. **Note**: This message can occur in response to an attempted Smurf attack.
Notification	Transit TCP in interface <portnum> exceeds <num> burst packets, stopping for <num> seconds!!	Threshold parameters for TCP transit (through) traffic have been configured on an interface, and the maximum burst size for TCP packets on the interface has been exceeded. The <portnum> is the port number. The first <num> is the maximum burst size (maximum number of packets allowed). The second <num> is the number of seconds during which additional TCP packets will be blocked on the interface. **Note**: This message can occur in response to an attempted TCP SYN attack.
Notification	VRRP intf state changed, intf <portnum>, vrid <virtual-router-id>, state <vrrp-state>	A state change has occurred in a Virtual Router Redundancy Protocol (VRRP) interface. The <portnum> is the port. The <virtual-router-id> is the virtual router ID (VRID) configured on the interface. The <vrrp-state> can be one of the following: • init • master • backup • unknown
Warning	DOT1X security violation at port <portnum>, malicious mac address detected: <mac-address>	A security violation was encountered at the specified port number.

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Warning	Dup IP <ip-addr> detected, sent from MAC <mac-addr> interface <portnum>	Indicates that the Foundry device received a packet from another device on the network with an IP address that is also configured on the Foundry device. The <ip-addr> is the duplicate IP address. The <mac-addr> is the MAC address of the device with the duplicate IP address. The <portnum> is the Foundry port that received the packet with the duplicate IP address. The address is the packet's source IP address.
Warning	list <acl-num> denied <ip-proto> <src-ip-addr> (<src-tcp/udp-port>) (Ethernet <portnum> <mac-addr>) -> <dst-ip-addr> (<dst-tcp/udp-port>), 1 event(s)	Indicates that an Access Control List (ACL) denied (dropped) packets. The <acl-num> indicates the ACL number. Numbers 1 – 99 indicate standard ACLs. Numbers 100 – 199 indicate extended ACLs. The <ip-proto> indicates the IP protocol of the denied packets. The <src-ip-addr> is the source IP address of the denied packets. The <src-tcp/udp-port> is the source TCP or UDP port, if applicable, of the denied packets. The <portnum> indicates the port number on which the packet was denied. The <mac-addr> indicates the source MAC address of the denied packets. The <dst-ip-addr> indicates the destination IP address of the denied packets. The <dst-tcp/udp-port> indicates the destination TCP or UDP port number, if applicable, of the denied packets.
Warning	Locked address violation at interface e<portnum>, address <mac-address>	Indicates that a port on which you have configured a lock-address filter received a packet that was dropped because the packet's source MAC address did not match an address learned by the port before the lock took effect. The e<portnum> is the port number. The <mac-address> is the MAC address that was denied by the address lock. Assuming that you configured the port to learn only the addresses that have valid access to the port, this message indicates a security violation.

Table A.3: Foundry Syslog Messages (Continued)

Message Level	Message	Explanation
Warning	mac filter group denied packets on port <portnum> src macaddr <mac-addr>, <num> packets	Indicates that a Layer 2 MAC filter group configured on a port has denied packets. The <portnum> is the port on which the packets were denied. The <mac-addr> is the source MAC address of the denied packets. The <num> indicates how many packets matching the values above were dropped during the five-minute interval represented by the log entry.
Warning	No of prefixes received from BGP peer <ip-addr> exceeds warning limit <num>	The Layer 3 Switch has received more than the allowed percentage of prefixes from the neighbor. The <ip-addr> is the IP address of the neighbor. The <num> is the number of prefixes that matches the percentage you specified. For example, if you specified a threshold of 100 prefixes and 75 percent as the warning threshold, this message is generated if the Layer 3 Switch receives a 76th prefix from the neighbor.
Warning	NTP server <ip-addr> failed to respond	Indicates that a Simple Network Time Protocol (SNTP) server did not respond to the device's query for the current time. The <ip-addr> indicates the IP address of the SNTP server.
Warning	rip filter list <list-num> <direction> V1 | V2 denied <ip-addr>, <num> packets	Indicates that a RIP route filter denied (dropped) packets. The <list-num> is the ID of the filter list. The <direction> indicates whether the filter was applied to incoming packets or outgoing packets. The value can be one of the following: • in • out The V1 or V2 value specifies the RIP version (RIPv1 or RIPv2). The <ip-addr> indicates the network number in the denied updates. The <num> indicates how many packets matching the values above were dropped during the five-minute interval represented by the log entry.

Appendix B
Remote Network Monitoring

This appendix describes the remote monitoring features available on Foundry products:

Table B.1: Chapter Contents

Basic Management

The following sections contain procedures for basic system management tasks.

Viewing System Information

You can access software and hardware specifics for a Foundry Layer 2 Switch or Layer 3 Switch.

To view the software and hardware details for the system, enter the **show version** command:

```
FESX424 Router# show version
```

Syntax: show version

Viewing Configuration Information

You can view a variety of configuration details and statistics with the show option. The **show** option provides a convenient way to check configuration changes before saving them to flash.

The show options available will vary for Layer 2 Switches and Layer 3 Switches and by configuration level.

To determine the available show commands for the system or a specific level of the CLI, enter the following command:

```
FESX424 Router# show ?
```

Syntax: show <option>

You also can enter “show” at the command prompt, then press the TAB key.

NOTE: For a complete summary of all available **show...** CLI commands and their displays, see the *Foundry Switch and Router Command Line Interface Reference*.

Viewing Port Statistics

Port statistics are polled by default every 10 seconds.

You can view statistics for ports by entering the following **show** commands:

- show interfaces
- show configuration
- show statistics

To display the statistics, enter a command such as the following:

```
FastIronSuperX Switch(config)# show statistics ethernet 1/3
Port  Link State     Dupl Speed Trunk Tag Priori MAC            Name
1/3    Up    Forward   Half 100M  None  No  level0 00e0.5200.0102

 Port 1/3 Counters:
         InOctets                3200          OutOctets                256
           InPkts                  50            OutPkts                  4
  InBroadcastPkts                   0    OutBroadcastPkts                3
  InMulticastPkts                  48    OutMulticastPkts                0
    InUnicastPkts                   2      OutUnicastPkts                1
        InBadPkts                   0
      InFragments                   0
       InDiscards                   0          OutErrors                 0
              CRC                   0         Collisions                 0
         InErrors                   0      LateCollisions                0
      InGiantPkts                   0
      InShortPkts                   0
         InJabber                   0
   InFlowCtrlPkts                   0     OutFlowCtrlPkts                0
     InBitsPerSec                 264       OutBitsPerSec               16
     InPktsPerSec                   0       OutPktsPerSec                0
     InUtilization              0.00%      OutUtilization            0.00%
```

Syntax: show statistics [ethernet <slotnum> <portnum>]

Table B.2 lists the statistics displayed in the output of the **show statistics** command.

Table B.2: Port Statistics

This Line...	Displays...
Port Configuration	
Port	The port number.
Link	The link state.
State	The STP state.
Dupl	The mode (full-duplex or half-duplex).
Speed	The port speed (10M, 100M, or 1000M).
Trunk	The trunk group number, if the port is a member of a trunk group.
Tag	Whether the port is a tagged member of a VLAN.
Priori	The QoS forwarding priority of the port (level0 – level7).
MAC	The MAC address of the port.
Name	The name of the port, if you assigned a name.
Statistics	
InOctets	The total number of good octets and bad octets received.
OutOctets	The total number of good octets and bad octets sent.
InPkts	The total number of packets received. The count includes rejected and local packets that are not sent to the switching core for transmission.
OutPkts	The total number of good packets sent. The count includes unicast, multicast, and broadcast packets.
InBroadcastPkts	The total number of good broadcast packets received.
OutBroadcastPkts	The total number of good broadcast packets sent.
InMulticastPkts	The total number of good multicast packets received.
OutMulticastPkts	The total number of good multicast packets sent.
InUnicastPkts	The total number of good unicast packets received.
OutUnicastPkts	The total number of good unicast packets sent.
InBadPkts	The total number of packets received for which one of the following is true: • The CRC was invalid. • The packet was oversized. • Jabbers: The packets were longer than 1518 octets and had a bad FCS. • Fragments: The packets were less than 64 octets long and had a bad FCS. • The packet was undersized (short).

Table B.2: Port Statistics (Continued)

This Line...	Displays...
InFragments	The total number of packets received for which both of the following was true: • The length was less than 64 bytes. • The CRC was invalid.
InDiscards	The total number of packets that were received and then dropped due to a lack of receive buffers.
OutErrors	The total number of packets with internal transmit errors such as TX underruns.
CRC	The total number of packets received for which all of the following was true: • The data length was between 64 bytes and the maximum allowable frame size. • No Collision or Late Collision was detected. • The CRC was invalid.
Collisions	The total number of packets received in which a Collision event was detected.
InErrors	The total number of packets received that had Alignment errors or phy errors.
LateCollisions	The total number of packets received in which a Collision event was detected, but for which a receive error (Rx Error) event was not detected.
InGiantPkts	The total number of packets for which all of the following was true: • The data length was longer than the maximum allowable frame size. • No Rx Error was detected. **Note**: Packets are counted for this statistic regardless of whether the CRC is valid or invalid.
InShortPkts	The total number of packets received for which all of the following was true: • The data length was less than 64 bytes. • No Rx Error was detected. • No Collision or Late Collision was detected. **Note**: Packets are counted for this statistic regardless of whether the CRC is valid or invalid.
InJabber	The total number of packets received for which all of the following was true: • The data length was longer than the maximum allowable frame size. • No Rx Error was detected. • The CRC was invalid.

Table B.2: Port Statistics (Continued)

This Line...	Displays...
InFlowCtrlPkts	The total number of flow control packets received.
OutFlowCtrlPkts	The total number of flow control packets transmitted.
InBitsPerSec	The number of bits received per second.
OutBitsPerSec	The number of bits sent per second.
InPktsPerSec	The number of packets received per second.
OutPktsPerSec	The number of packets sent per second.
InUtilization	The percentage of the port's bandwidth used by received traffic.
OutUtilization	The percentage of the port's bandwidth used by sent traffic.

Viewing STP Statistics

You can view a summary of STP statistics for Layer 2 Switches and Layer 3 Switches. STP statistics are by default polled every 10 seconds.

To view spanning tree statistics, enter the **show span** command. To view STP statistics for a VLAN, enter the **span vlan** command.

Clearing Statistics

You can clear statistics for many parameters with the clear option.

To determine the available **clear** commands for the system, enter the following command:

```
FESX424 Router# clear ?
```

Syntax: clear <option>

You also can enter "clear" at the command prompt, then press the TAB key.

For a complete summary of all available **clear...** CLI commands and their displays, see the *Foundry Switch and Router Command Line Interface Reference*.

NOTE: Clear commands are found at the Privileged EXEC level.

RMON Support

The Foundry RMON agent supports the following groups. The group numbers come from the RMON specification (RFC 1757).

- Statistics (RMON Group 1)
- History (RMON Group 2)
- Alarms (RMON Group 3)
- Events (RMON Group 9)

The CLI allows you to make configuration changes to the control data for these groups, but you need a separate RMON application to view and display the data graphically.

Statistics (RMON Group 1)

Count information on multicast and broadcast packets, total packets sent, undersized and oversized packets, CRC alignment errors, jabbers, collision, fragments and dropped events is collected for each port on a Foundry Layer 2 Switch or Layer 3 Switch.

No configuration is required to activate collection of statistics for the Layer 2 Switch or Layer 3 Switch. This activity is by default automatically activated at system start-up.

You can view a textual summary of the statistics for all ports by entering the following CLI command:

```
FastIron SuperX Router(config)# show rmon statistics
Ethernet statistics 1 is active, owned by monitor
 Interface 1/1 (ifIndex 1) counters
                   Octets          0
                Drop events        0                   Packets          0
             Broadcast pkts        0            Multicast pkts          0
      CRC alignment errors         0            Undersize pkts          0
              Oversize pkts        0                 Fragments          0
                   Jabbers         0               Collisions          0
             64 octets pkts        0     65 to 127 octets pkts          0
    128 to 255 octets pkts         0    256 to 511 octets pkts          0
   512 to 1023 octets pkts         0  1024 to 1518 octets pkts          0
```

Syntax: show rmon statistics [[<slotnum>/]<portnum>]

The <portnum> parameter specifies the port number. You can use the physical port number or the SNMP port number. The physical port number is based on the product. If you specify a physical port on a chassis device, you must also enter the slot number.

- If the product is a Stackable device, the ports are numbered sequentially starting with 1.
- If the product is a Chassis device, the ports are numbered according to slot and port. For example, the first port in slot 1 is 1/1. The third port in slot 7 is 7/3.

The SNMP numbers of the ports start at 1 and increase sequentially. For example, if you are using a Chassis device and slot 1 contains an 8-port module, the SNMP number of the first port in slot 2 is 9. The physical port number of the same port is 2/1.

This command shows the following information.

Table B.3: Export Configuration and Statistics

This Line...	Displays...
Octets	The total number of octets of data received on the network. This number includes octets in bad packets. This number does not include framing bits but does include Frame Check Sequence (FCS) octets.
Drop events	Indicates an overrun at the port. The port logic could not receive the traffic at full line rate and had to drop some packets as a result. The counter indicates the total number of events in which packets were dropped by the RMON probe due to lack of resources. This number is not necessarily the number of packets dropped, but is the number of times an overrun condition has been detected.

Table B.3: Export Configuration and Statistics (Continued)

This Line...	Displays...
Packets	The total number of packets received. This number includes bad packets, broadcast packets, and multicast packets.
Broadcast pkts	The total number of good packets received that were directed to the broadcast address. This number does not include multicast packets.
Multicast pkts	The total number of good packets received that were directed to a multicast address. This number does not include packets directed to the broadcast address.
CRC alignment errors	The total number of packets received that were from 64 – 1518 octets long, but had either a bad FCS with an integral number of octets (FCS Error) or a bad FCS with a non-integral number of octets (Alignment Error). The packet length does not include framing bits but does include FCS octets.
Undersize pkts	The total number of packets received that were less than 64 octets long and were otherwise well formed. This number does not include framing bits but does include FCS octets.
Fragments	The total number of packets received that were less than 64 octets long and had either a bad FCS with an integral number of octets (FCS Error) or a bad FCS with a non-integral number of octets (Alignment Error). It is normal for this counter to increment, since it counts both runts (which are normal occurrences due to collisions) and noise hits. This number does not include framing bits but does include FCS octets.
Oversize packets	The total number of packets received that were longer than 1518 octets and were otherwise well formed. This number does not include framing bits but does include FCS octets.
Jabbers	The total number of packets received that were longer than 1518 octets and had either a bad FCS with an integral number of octets (FCS Error) or a bad FCS with a non-integral number of octets (Alignment Error). **Note**: This definition of jabber is different from the definition in IEEE-802.3 section 8.2.1.5 (10BASE5) and section 10.3.1.4 (10BASE2). These documents define jabber as the condition where any packet exceeds 20 ms. The allowed range to detect jabber is between 20 ms and 150 ms. This number does not include framing bits but does include FCS octets.

Table B.3: Export Configuration and Statistics (Continued)

This Line...	Displays...
Collisions	The best estimate of the total number of collisions on this Ethernet segment.
64 octets pkts	The total number of packets received that were 64 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets.
65 to 127 octets pkts	The total number of packets received that were 65 – 127 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets.
128 to 255 octets pkts	The total number of packets received that were 128 – 255 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets.
256 to 511 octets pkts	The total number of packets received that were 256 – 511 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets.
512 to 1023 octets pkts	The total number of packets received that were 512 – 1023 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets.
1024 to 1518 octets pkts	The total number of packets received that were 1024 – 1518 octets long. This number includes bad packets. This number does not include framing bits but does include FCS octets.

History (RMON Group 2)

All active ports by default will generate two history control data entries per active Foundry Layer 2 Switch port or Layer 3 Switch interface. An active port is defined as one with a link up. If the link goes down the two entries are automatically deleted.

Two history entries are generated for each device:

- a sampling of statistics every 30 seconds
- a sampling of statistics every 30 minutes

The history data can be accessed and displayed using any of the popular RMON applications

A sample RMON history command and its syntax is shown below:


```
FESX424 Router(config)# rmon history 1 interface 1 buckets 10 interval 10 owner
nyc02
```

Syntax: rmon history <entry-number> interface [<slotnum>/]<portnum> buckets <number> interval <sampling-interval> owner <text-string>

You can modify the sampling interval and the bucket (number of entries saved before overwrite) using the CLI. In the above example, owner refers to the RMON station that will request the information.

NOTE: To review the control data entry for each port or interface, enter the **show rmon history** command.

Alarm (RMON Group 3)

Alarm is designed to monitor configured thresholds for any SNMP integer, time tick, gauge or counter MIB object. Using the CLI, you can define what MIB objects are monitored, the type of thresholds that are monitored (falling, rising or both), the value of those thresholds, and the sample type (absolute or delta).

An alarm event is reported each time that a threshold is exceeded. The alarm entry also indicates the action (event) to be taken if the threshold be exceeded.

A sample CLI alarm entry and its syntax is shown below:

```
FESX424 Router(config)# rmon alarm 1 ifInOctets.6 10 delta rising-threshold 100 1
falling threshold 50 1 owner nyc02
```

Syntax: rmon alarm <entry-number> <MIB-object.interface-num> <sampling-time> <sample-type> <threshold-type> <threshold-value> <event-number> <threshold-type> <threshold-value> <event-number> owner <text-string>

Event (RMON Group 9)

There are two elements to the Event Group—the ***event control table*** and the ***event log table***.

The event control table defines the action to be taken when an alarm is reported. Defined events can be found by entering the CLI command, show event. The Event Log Table collects and stores reported events for retrieval by an RMON application.

A sample entry and syntax of the event control table is shown below:

```
FESX424 Router(config)# rmon event 1 description 'testing a longer string' log-and-
trap public owner nyc02
```

Syntax: rmon event <event-entry> description <text-string> log | trap | log-and-trap owner <rmon-station>

sFlow

sFlow is a system for observing traffic flow patterns and quantities within and among a set of Layer 2 Switches and Layer 3 Switches. To support sFlow, participating Layer 2 and Layer 3 devices:

- Sample packet flows
- Collect the packet headers from sampled packets and collect ingress-egress information on these packets
- Compose the collected information into flow sample messages
- Relay these messages to an external device known as a collector

Participating devices also relay byte and packet counter data (counter samples) for ports to the collector.

sFlow is described in RFC 3176, “InMon Corporation's sFlow: A Method for Monitoring Traffic in Switched and Routed Networks”. Refer to this RFC to determine the contents of the sampled packet.

Configuration Considerations

Hardware Support

- FESX, FWSX, and FSX devices support sFlow packet sampling of inbound traffic only. These devices do not sample outbound packets.
- sFlow is supported on all Ethernet ports (10/100, Gigabit, and 10 Gigabit)

On these devices, sample data is collected from inbound traffic on ports enabled for sFlow. Outbound traffic is sampled on the FastIron Edge Switches only. However, both traffic directions are counted for byte and packet counter statistics sent to the collector.

Source Address

The sampled sFlow data sent to the collectors includes an agent_address field. This field identifies the IP address of the device that sent the data.

- On a Layer 2 Switch, agent_address is the Layer 2 Switch's management IP address. You must configure the management IP address in order to export sFlow data from the device.
- On a Layer 3 Switch, sFlow looks for an IP address in the following order, and uses the first address found:
  - The router ID configured by the **ip router-id** command
  - The first IP address on the lowest-numbered loopback interface
  - The first IP address on the lowest-numbered virtual interface
  - The first IP address on any interface

  NOTE: The device uses the router ID only if the device also has an IP interface with the same address.

NOTE: If an IP address in not already configured when you enable sFlow, the feature uses the source address 0.0.0.0. To display the agent_address, enable sFlow, then enter the **show sflow** command. See "Enabling sFlow Forwarding" on page B-14 and "Displaying sFlow Information" on page B-15.

NOTE: If you change the address sFlow will use for the agent_address, you must disable and re-enable sFlow to enable the feature to use the changed address.

Sampling Rate

The ***sampling rate*** is the average ratio of the number of packets incoming on an sflow enabled port, to the number of flow samples taken from those packets. sFlow sampling can affect performance in some configurations.

Note that on the X-Series devices, the configured sampling rate and the actual rate are the same. The software does not adjust the configured sampling rate as on other Foundry devices.

Port Monitoring

- FESX and FWSX devices running software release 02.2.01 or later support port monitoring and sFlow together on the same device. The caveat is that these features cannot be configured together within the same port region. See "About Port Regions" on page 4-2 for a list of valid port regions.
- FSX devices running software release 02.2.00 or later support port monitoring and sFlow together on the same device. The caveat is that these features cannot be configured together within the same port region.

sflow Support for IPv6 Packets

Foundry's implementation of sFlow features provide support for IPv6 packets. This support includes extended router information and extended gateway information in the sampled packet. Note that sFlow support for IPv6 packets exists only on devices running software that supports IPv6.

Extended Router Information

Extended router information contains information for the next hop router. This information includes the next hop router's IP address and the outgoing VLAN ID. Extended router information also includes the source IP address prefix length and the destination IP address prefix length.

Note that in IPv4, prefix length of source and destination IP addresses is collected only if BGP is configured on the devices. In IPv6, the information is collected if BGP is configured and once the route lookup is complete.

To obtain extended router information in IPv6 sampled packets, use "struct extended_router" as presented in RFC 3176.

Extended Gateway Information

Extended gateway information is included in an sFlow sampled packet if BGP is enabled. The extended gateway information includes the following BGP information about the packet's destination route:

- This router's autonomous system (AS) number
- The route's source IP AS
- The route's source peer AS
- The AS path to the destination

NOTE: AS communities and local preferences are not included in the sampled packets.

To obtain extended gateway information use "struct extended_gateway" as described in RFC 3176.

Configuring and Enabling sFlow

To configure sFlow:

- Specify collector information. The collector is the external device to which you are exporting the sFlow data. You can specify up to four collectors.
- Optional – Change the polling interval.
- Optional – Change the sampling rate.
- Enable sFlow globally.
- Enable sFlow forwarding on individual interfaces.

NOTE: If you change the router ID or other IP address value that sFlow uses for its agent_address, you need to disable and then re-enable sFlow to cause the feature to use the new source address.

Specifying the Collector

sFlow exports traffic statistics to an external collector. You can specify up to four collectors. You can specify more than one collector with the same IP address if the UDP port numbers are unique. You can have up to four unique combinations of IP address and UDP port number.

To specify sFlow collectors, enter a command such as the following:

```
FESX424 Router(config)# sflow destination 10.10.10.1
```

This command specifies a collector with IP address 10.10.10.1, listening for sFlow data on UDP port 6343.

Syntax: [no] sflow destination <ip-addr> [<dest-udp-port>]

The <ip-addr> parameter specifies the collector's IP address.

The <dest-udp-port> parameter specifies the UDP port on which the sFlow collector will be listening for exported sFlow data. The default port number is 6343.

The sampled sFlow data sent to the collectors includes an agent_address field. This field identifies the device that sent the data. See "Source Address" on page B-10.

Changing the Polling Interval

The polling interval defines how often sFlow byte and packet counter data for a port are sent to the sFlow collector(s). If multiple ports are enabled for sFlow, the Foundry device staggers transmission of the counter data to smooth performance. For example, if sFlow is enabled on two ports and the polling interval is 20 seconds, the Foundry device sends counter data every ten seconds. The counter data for one of the ports are sent after ten seconds, and counter data for the other port are sent after an additional ten seconds. Ten seconds later, new counter data for the first port are sent. Similarly, if sFlow is enabled on five ports and the polling interval is 20 seconds, the Foundry device sends counter data every four seconds.

The default polling interval is 20 seconds. You can change the interval to a value from 1 to any higher value. The interval value applies to all interfaces on which sFlow is enabled. If you set the polling interval to 0, counter data sampling is disabled.

To change the polling interval, enter a command such as the following at the global CONFIG level of the CLI:

```
FESX424 Router(config)# sflow polling-interval 30
```

Syntax: [no] sflow polling-interval <secs>

The <secs> parameter specifies the interval and can be from 1 to any higher value. The default is 20 seconds. If you specify 0, counter data sampling is disabled.

Changing the Sampling Rate

The sampling rate is the average ratio of the number of packets incoming on an sFlow-enabled port, to the number of flow samples taken from those packets.

You can change the default (global) sampling rate. You also can change the rate on an individual port, overriding the default sampling rate of 512. With a sampling rate of 512, on average, one in every 512 packets forwarded on an interface is sampled.

Configuration Considerations

The sampling rate is a fraction in the form 1/N, meaning that, on average, one out of every N packets will be sampled. The **sflow sample** command at the global level or port level specifies N, the denominator of the fraction. Thus a higher number for the denominator means a lower sampling rate since fewer packets are sampled. Likewise, a lower number for the denominator means a higher sampling rate because more packets are sampled. For example, if you change the denominator from 512 to 128, the sampling rate increases because four times as many packets will be sampled.

NOTE: Foundry recommends that you do not change the denominator to a value lower than the default. Sampling requires CPU resources. Using a low denominator for the sampling rate can cause high CPU utilization.

Configured Rate and Actual Rate

When you enter a sampling rate value, this value is the ***configured rate***. The software rounds the value you enter to the next higher odd power of 2 to obtain the ***actual rate***. This value becomes the actual sampling rate. For example, if the configured sampling rate is 1000, then the actual rate is 2048 and 1 in 2048 packets are sampled by the hardware.

Change to Global Rate

If you change the global sampling rate, the change is applied to all sFlow-enabled ports ***except*** those ports on which you have already explicitly set the sampling rate. For example, suppose that sFlow is enabled on ports 1/1, 1/2, and 5/1. If you configure the sampling rate on port 1/1 but leave the other two ports using the default rate, then a change to the global sampling rate applies to ports 1/2 and 5/1 but not port 1/1. sFlow assumes that you want to continue using the sampling rate you explicitly configured on an individual port even if you globally change the sampling rate for the other ports.

Module Rate

While different ports on a module may be configured to have different sampling rates, the hardware for the module will be programmed to take samples at a single rate (the module sampling rate). The module sampling rate will be the highest sampling rate (i.e. lowest number) configured for any of the ports on the module.

When ports on a given module are configured with different sampling rates, the CPU discards some of the samples supplied by the hardware for ports with configured sampling rates which are lower than the module sampling rate. This is referred to as subsampling, and the ratio between the port sampling rate and the module sampling rate is known as the subsampling factor. For example, if the module in slot 4 has sFlow enabled on ports 4/2 and 4/8, and port 4/2 is using the default sampling rate of 512, and port 4/8 is configured explicitly for a rate of 2048, then the module sampling rate will be 512 because this is this highest port sampling rate (lowest number). The subsampling factor for port 4/2 will be 1, meaning that every sample taken by the hardware will be exported, while the subsampling factor for port 4/8 will be 4, meaning that one out of every four samples taken by the hardware will be exported. Whether a port's sampling rate is configured explicitly, or whether it uses the global default setting, has no effect on the calculations.

You do not need to perform any of these calculations to change a sampling rate. For simplicity, the syntax information in this section lists the valid sampling rates. In addition, the software will round the value you enter up to the nearest value listed. You can display the rates you entered (the configured rates) as well as the rates rounded up to by the software (the actual rates) for the default sampling rate, module rates, and all sFlow-enabled ports by entering the **show sflow** command. See “Displaying sFlow Information” on page B-15.

Sampling Rate for New Ports

When you enable sFlow on a port, the port's sampling rate is set to the global default sampling rate. This also applies to ports on which you disable and then re-enable sFlow. The port does not retain the sampling rate it had when you disabled sFlow on the port, even if you had explicitly set the sampling rate on the port.

Changing the Default Sampling Rate

To change the default (global) sampling rate, enter a command such as the following at the global CONFIG level of the CLI:

```
FESX424 Router(config)# sflow sample 2048
```

Syntax: [no] sflow sample <num>

The <num> parameter specifies the average number of packets from which each sample will be taken. The software rounds the value you enter to the next higher odd power of 2. This value becomes the actual default sampling rate and is one of the following.

- 2
- 8
- 32
- 128
- 512
- 2048
- 8192
- 32768
- 131072
- 524288
- 2097152
- 8388608
- 33554432
- 134217728
- 536870912
- 2147483648

For example, if the configured sampling rate is 1000, then the actual rate is 2048 and 1 in 2048 packets are sampled by the hardware.

Changing the Sampling Rate of a Module

You cannot change a module's sampling rate directly. You can change a module's sampling rate only by changing the sampling rate of a port on that module.

Changing the Sampling Rate on a Port

You can configure an individual port to use a different sampling rate than the global default sampling rate. This is useful in cases where ports have different bandwidths. For example, if you are using sFlow on 10/100 ports and Gigabit Ethernet ports, you might want to configure the Gigabit ports to use a higher sampling rate (and thus gather fewer samples per number of packets) than the 10/100 ports.

To change the sampling rate on an individual port, enter a command such as the following at the configuration level for the port:

```
FastIron SuperX Switch(config-if-1/1)# sflow sample 8192
```

Syntax: [no] sflow sample <num>

The <num> parameter specifies the average number of packets from which each sample will be taken. The software rounds the value you enter up to the next odd power of 2. The actual sampling rate becomes one of the values listed in "Changing the Default Sampling Rate".

Enabling sFlow Forwarding

sFlow exports data only for the interfaces on which you enable sFlow forwarding. You can enable sFlow forwarding on Ethernet interfaces.

To enable sFlow forwarding:

- Globally enable the sFlow feature.
- Enable sFlow forwarding on individual interfaces.

NOTE: Before you enable sFlow, make sure the device has an IP address that sFlow can use as its source address. See "Source Address" on page B-10 for the source address requirements.

NOTE: When you enable sFlow forwarding on an 802.1X-enabled interface, the samples taken from the interface include the username used to obtain access to the inbound and/or outbound ports, if that information is available. For information about 802.1X, see the *Foundry Security Guide*.

Enabling sFlow Forwarding

To enable sFlow forwarding, enter commands such as the following:

```
FESX424 Router(config)# sflow enable
FESX424 Router(config)# interface ethernet 1/1 to 1/8
FESX424 Router(config-mif-1/1-1/8)# sflow forwarding
```

These commands globally enable sFlow, then enable sFlow forwarding on Ethernet ports 1/1 – 1/8. You must use both the **sflow enable** and **sflow forwarding** commands to enable the feature.

Syntax: [no] sflow enable

Syntax: [no] sflow forwarding

Displaying sFlow Information

To display sFlow configuration information and statistics, enter the following command at any level of the CLI:

```
FastIron SuperX Router(config)# show sflow
sFlow services are enabled.
sFlow agent IP address: 123.123.123.1
4 collector destinations configured:
Collector IP 192.168.4.204, UDP 6343
Collector IP 192.168.4.200, UDP 6333
Collector IP 192.168.4.202, UDP 6355
Collector IP 192.168.4.203, UDP 6565
Polling interval is 0 seconds.
Configured default sampling rate: 1 per 512 packets.
Actual default sampling rate: 1 per 512 packets.
10552 UDP packets exported
24127 sFlow samples collected.
sFlow ports: ethe 1/2 to 1/12 ethe 1/15 ethe 1/25 to 1/26 ethe 4/1 ethe 5/10 to
5/20 ethe 8/1 ethe 8/4
Module Sampling Rates
---------------------
Slot  1 configured rate=512, actual rate=512
Slot  3 configured rate=0, actual rate=0
Slot  4 configured rate=10000, actual rate=32768
Slot  5 configured rate=512, actual rate=512
Slot  7 configured rate=0, actual rate=0
Slot  8 configured rate=512, actual rate=512
Port Sampling Rates
-------------------
Port 8/4, configured rate=512, actual rate=512, Subsampling factor=1
Port 8/1, configured rate=512, actual rate=512, Subsampling factor=1
Port 5/20, configured rate=3000, actual rate=8192, Subsampling factor=16
Port 5/19, configured rate=512, actual rate=512, Subsampling factor=1
Port 5/18, configured rate=512, actual rate=512, Subsampling factor=1
Port 5/17, configured rate=1500, actual rate=2048, Subsampling factor=4
Port 5/16, configured rate=1500, actual rate=2048, Subsampling factor=4
Port 5/15, configured rate=1500, actual rate=2048, Subsampling factor=4
Port 5/14, configured rate=1500, actual rate=2048, Subsampling factor=4
Port 5/13, configured rate=512, actual rate=512, Subsampling factor=1
Port 5/12, configured rate=512, actual rate=512, Subsampling factor=1
Port 5/11, configured rate=512, actual rate=512, Subsampling factor=1
Port 5/10, configured rate=512, actual rate=512, Subsampling factor=1
Port 4/1, configured rate=10000, actual rate=32768, Subsampling factor=1
Port 1/26, configured rate=512, actual rate=512, Subsampling factor=1
Port 1/25, configured rate=512, actual rate=512, Subsampling factor=1
Port 1/15, configured rate=512, actual rate=512, Subsampling factor=1
Port 1/12, configured rate=512, actual rate=512, Subsampling factor=1
Port 1/11, configured rate=512, actual rate=512, Subsampling factor=1
Port 1/10, configured rate=512, actual rate=512, Subsampling factor=1
Port 1/9, configured rate=512, actual rate=512, Subsampling factor=1
Port 1/8, configured rate=512, actual rate=512, Subsampling factor=1
Port 1/7, configured rate=1000, actual rate=2048, Subsampling factor=4
Port 1/6, configured rate=512, actual rate=512, Subsampling factor=1
Port 1/5, configured rate=512, actual rate=512, Subsampling factor=1
Port 1/4, configured rate=512, actual rate=512, Subsampling factor=1
Port 1/3, configured rate=512, actual rate=512, Subsampling factor=1
Port 1/2, configured rate=1000, actual rate=2048, Subsampling factor=4
```

Syntax: show sflow

This command shows the following information.

Table B.4: sFlow Information

This Field...	Displays...
sFlow services	The feature state, which can be one of the following: • disabled • enabled
sFlow agent IP address	The IP address that sFlow is using in the agent_address field of packets sent to the collectors. See “Source Address” on page B-10.
Collector	The collector information. The following information is displayed for each collector: • IP address • UDP port If more than one collector is configured, the line above the collectors indicates how many have been configured.
Polling interval	The port counter polling interval.
Configured default sampling rate	The configured global sampling rate. If you changed the global sampling rate, the value you entered is shown here. The actual rate calculated by the software based on the value you entered is listed on the next line, ”Actual default sampling rate”.
Actual default sampling rate	The actual default sampling rate.
UDP packets exported	The number of sFlow export packets the Foundry device has sent. **Note**: Each UDP packet can contain multiple samples.
sFlow samples collected	The number of sampled packets that have been sent to the collector(s).
sFlow ports	The ports on which you enabled sFlow.
Module Sampling Rates	The configured and actual sampling rates for each module. If a module does not have any sFlow-enabled ports, the rates are listed as 0.
Port Sampling Rates	The configured and actual sampling rates for each sFlow-enabled port. The Subsampling factor indicates how many times the sampling rate of the port's module is multiplied to achieve the port's sampling rate. Because of the way the actual sampling rates are computed, the Subsampling factors are always whole numbers.

Clearing sFlow Statistics

To clear the UDP packet and sFlow sample counters in the **show sflow** display, enter the following command:

```
FESX424 Router(config)# clear statistics
```

Syntax: clear statistics

This command clears the values in the following fields of the **show sflow** display:

- UDP packets exported
- sFlow samples collected

NOTE: This command also clears the statistics counters used by other features.

Configuring a Utilization List for an Uplink Port

You can configure uplink utilization lists that display the percentage of a given uplink port's bandwidth that is used by a specific list of downlink ports. The percentages are based on 30-second intervals of RMON packet statistics for the ports. Both transmit and receive traffic is counted in each percentage.

NOTE: This feature is intended for ISP or collocation environments in which downlink ports are dedicated to various customers' traffic and are isolated from one another. If traffic regularly passes between the downlink ports, the information displayed by the utilization lists does not provide a clear depiction of traffic exchanged by the downlink ports and the uplink port.

Each uplink utilization list consists of the following:

- Utilization list number (1, 2, 3, or 4)
- One or more uplink ports
- One or more downlink ports

Each list displays the uplink port and the percentage of that port's bandwidth that was utilized by the downlink ports over the most recent 30-second interval.

You can configure up to four bandwidth utilization lists.

Command Syntax

To configure an uplink utilization list, enter commands such as the following. The commands in this example configure a link utilization list with port 1/1 as the uplink port and ports 1/2 and 1/3 as the downlink ports.

```
FastIron SuperX Router(config)# relative-utilization 1 uplink eth 1/1 downlink eth
1/2 to 1/3
FastIron SuperX Router(config)# write memory
```

Syntax: [no] relative-utilization <num> uplink ethernet [<slotnum>/]<portnum> [to [<slotnum>/]<portnum> | [<slotnum>/]<portnum>…]
downlink ethernet [<slotnum>/]<portnum> [to [<slotnum>/]<portnum> | [<slotnum>/]<portnum>…]

The <num> parameter specifies the list number. You can configure up to four lists. Specify a number from 1 – 4.

The **uplink ethernet** parameters and the port number(s) you specify after the parameters indicate the uplink port(s).

The **downlink ethernet** parameters and the port number(s) you specify after the parameters indicate the downlink port(s).

Displaying Utilization Percentages for an Uplink

After you configure an uplink utilization list, you can display the list to observe the percentage of the uplink's bandwidth that each of the downlink ports used during the most recent 30-second port statistics interval. The number of packets sent and received between the two ports is listed, as well as the ratio of each individual downlink port's packets relative to the total number of packets on the uplink.

To display an uplink utilization list, enter a command such as the following at any level of the CLI:

```
FastIron SuperX Router(config)# show relative-utilization 1
```

```
uplink: ethe 1
30-sec total uplink packet count = 3011
packet count ratio (%)
  1/ 2:60   1/ 3:40
```

In this example, ports 1/2 and 1/3 are sending traffic to port 1/1. Port 1/2 and port 1/3 are isolated (not shared by multiple clients) and typically do not exchange traffic with other ports except for the uplink port, 1/1.

Syntax: show relative-utilization <num>

The <num> parameter specifies the list number.

NOTE: The example above represents a pure configuration in which traffic is exchanged only by ports 1/2 and 1/1, and by ports 1/3 and 1/1. For this reason, the percentages for the two downlink ports equal 100%. In some cases, the percentages do not always equal 100%. This is true in cases where the ports exchange some traffic with other ports in the system or when the downlink ports are configured together in a port-based VLAN.

In the following example, ports 1/2 and 1/3 are in the same port-based VLAN.

```
FastIron SuperX Router(config)# show relative-utilization 1
uplink: ethe 1
30-sec total uplink packet count = 3011
packet count ratio (%)
  1/ 2:100   1/ 3:100
```

Here is another example showing different data for the same link utilization list. In this example, port 1/2 is connected to a hub and is sending traffic to port 1/1. Port 1/3 is unconnected.

```
FastIron SuperX Router(config)# show relative-utilization 1
uplink: ethe 1
30-sec total uplink packet count = 2996
packet count ratio (%)
  1 /2:100   1/ 3:---
```

Appendix C
Policies and Filters

This appendix describes the various types of Foundry policies and filters. For each type of policy or filter, the CLI command syntax for configuring the policy or filter are provided. This appendix also refers you to specific configuration procedures.

NOTE: This appendix does not describe Access Control Lists (ACLs), which are additional methods for filtering packets. See "Rule-Based IP Access Control Lists (ACLs)" on page 12-1.

Table C.1: Chapter Contents

Foundry devices provide a robust array of policies and filters. You can configure policies and filters to do the following:

- Change Quality-of-Service priorities for individual ports, VLANs, and static MAC entries.
- Configure protocol-based VLANs, IP sub-net VLANs, and IPX network VLANs within standard 802.1d port-based VLANs.
- Learn or drop RIP routes on incoming traffic, based on network address or the RIP neighbor's IP address.
- Control learning and advertisement of RIP routes, based on network address or the RIP neighbor's IP

address.

- Control learning and advertisement of IPX RIP routes.
- Permit or deny access to IPX servers.
- Control learning and advertisement of routes learned from BGP4 neighbors. You can filter based on network address information, AS-path information, and community names.
- Redistribute routes among RIP, OSPF, and BGP4.
- Filter on specific MAC addresses, on Layer 2 multicast packets, and on Layer 2 broadcast packets.

NOTE: Foundry recommends that you use ACLs to handle L4 prioritization on all other platforms.

Scope

Some policies and filters are configured and apply globally, while others are configured globally but apply to individual ports. The following table lists the scope for each type of policy and filter.

Table C.2: Scopes of Policies and Filters

Policy or Filter Type	Scope
QoS policy	Configured and applied to one of the following: • Ports • VLANs • Static MAC entries
Address-lock filter	Configured and applied on individual ports.
Route filters • RIP route filters • IPX RIP route filters • IPX SAP service filters	Configured globally and applied to individual ports
RIP neighbor filters	Configured and applied globally
BGP4 filters • BGP4 address • BGP4 AS-path • BGP4 community	Configured and applied globally and in route maps
Route redistribution filters • RIP • OSPF • BGP4	Configured and applied globally

Default Filter Actions

By default, no policies or filters are defined on Foundry devices. The following table lists the default action when no policy or filter is configured and the default action after you configure a policy or filter. For some types of

policies and filters, the default action changes once you configure a policy or filter, regardless of the policy or filter's contents.

Table C.3: Default Policy and Filter Actions

Policy or Filter Type	Default action when no policies or filters are configured	Default action after a policy or filter is configured
QoS policy	Queue all packets in normal or 0 priority queue	Queue all packets in normal or 0 priority queue unless explicitly configured for a higher queue
Address-lock filter	Permit (forward) all packets	Permit only those packets whose source MAC addresses have been learned on the port; drop all others
Route filters • RIP neighbor filters • BGP4 address filters • BGP4 AS-path filters • BGP4 community filters	Permit (learn and advertise) all routes or services	Deny (do not learn or advertise) all routes or services
Route filters • RIP route filters	Permit (learn and advertise) all routes or services	Permit (learn and advertise) all routes or services
Route redistribution filter • RIP • OSPF • BGP4	Redistribute all routes if redistribution is enabled.	Once redistribution is enabled, redistribute routes of the specified type unless explicitly denied by filter **Note**: For RIP and OSPF, you must explicitly enable redistribution. Redistribution is enabled by default in BGP4.

Policy and Filter Precedence

QoS

You can apply QoS policies to individual ports, VLANs, static MAC address, Layer 4 sessions, and AppleTalk sockets. If a port is a member of two or more of these items and has different priorities, the priorities are merged. However, the resulting priority is never lower than the highest priority.

Precedence Among Filters on Different Layers

Generally, the Foundry device applies only the type of filter that applies to the traffic. For example, if a packet is a Layer 2 switched packet, then the device evaluates the packet against the port's MAC filters. If a packet is a routed IP packet, the device evaluates the packet against the port's IP access policies.

Foundry recommends that you do not use filters at different layers on the same port. For example, do not use MAC filters and IP access policies on the same port.

NOTE: You cannot use Layer 2 filters to filter for Layer 4 information. To filter for Layer 4 information, use IP access policies (filters).

NOTE: If you do choose to apply filters for multiple layers to the same port, note that Layer 2 MAC filters can affect the Layer 3 IP traffic that a port permits or denies on multinetted interfaces. A multinetted interface has multiple IP sub-net interfaces on the same port. MAC filters can filter on the Ethertype field. This field includes Layer 3 protocol information and identifies packets as IP packets, ARP packets, and so on.

If you configure a MAC filter, then leave the default action as “deny any”, all packets from one of the IP sub-net addresses to another address on the same multinetted interface that do not match the filter are denied. This includes packet types such as IP and ARP. The result is that you have a Layer 2 filter but Layer 3 traffic is dropped. To avoid this, make sure you configure a filter to “permit any” traffic, thus changing the default action to permit for packets that are not denied by the other MAC filters.

Precedence Among Filters on the Same Layer

For most types of filters, a Foundry device applies filters based on the order in which you list them in a port’s inbound or outbound filter list. For example, if you apply three filters, 3, 2, and 1024 to port 1/1’s outbound filter list, the filters are applied in the following order: 3, 2, 1024.

You must configure the policies or filters before you can add them to a policy or filter group.

When you configure a policy or filter group, you must add all the policies or filters at the same time. You cannot edit policy or filter groups. To change a group, you must delete it, then add a new one.

Foundry Policies

On a Foundry device, a policy is a set of rules that defines how the device handles packets. The following table lists the types of policies you can configure on a Foundry device.

Table C.4: Foundry Policies

Policy Type	Supported on...		See page...
	Router	Switch	
Quality-of-Service (QoS) Policies	X	X	C-5
Layer 3 Policies			C-5
Protocol-based VLANs – either forward or drop Layer 3 traffic based on protocol (or, for IP sub-net VLANs and IPX network VLANs, sub-net or network address)	X	X	C-5

Table C.5: Policies

Policy Type	See page...
Quality-of-Service (QoS) Policies	B-7
Layer 3 Policies	B-9
Protocol-based VLANs – either forward or drop Layer 3 traffic based on protocol (or, for IP sub-net VLANs and IPX network VLANs, sub-net or network address)	B-9
IP access policies – either forward or drop IP packets	B-10
Layer 4 Policies	B-38

Table C.5: Policies(Continued)

Policy Type	See page...
TCP/UDP access policies – either forward or drop packets based on TCP or UDP port	B-18

Quality-of-Service Policies

Foundry devices support Quality-of-Service (QoS) through implementation of 802.1q prioritization. You can configure QoS policies for packets associated with the following items:

- Ports
- VLANs
- Static MAC entries

The FESX, FSX, and FWSX provide eight QoS queues: 0 (normal) – 7 (highest priority).

The default queue for all packets is normal (or 0). You can change a QoS policy by placing a port, VLAN, or static MAC entry into a higher queue. See the chapter “Quality of Service” on page 18-1 for more information about the Foundry QoS algorithms.

Actions

QoS policies place packets in the specified queue for forwarding.

Scope

You can apply QoS policies to individual ports, VLANs, and static MAC address. If a port is a member of two or more of these items and has different priorities, the priorities are merged. However, the resulting priority is never lower than the highest priority.

Syntax

Use the following CLI commands to configure QoS policies.

Table C.6: QoS Policies

QoS Scope	CLI syntax
Individual port	(config-if-1/1)# priority <0-7>
Static MAC address[a]	(config)# static-mac-address <mac-addr> ethernet [<slotnum>/]<portnum> [priority <0-7>] [host-type | router-type]

a. You can configure static MAC addresses on Layer 2 Switches but not on Layer 3 Switches.

Layer 3 Policies

Layer 3 policies are rules that control transmission and receipt of packets based on Layer 3 routing protocol information in the packets. You can configure the following types of Layer 3 policies:

- Protocol-based VLANs

Protocol-Based VLANs

Within an 802.1d port-based VLAN, you can configure protocol-based VLANs that define Layer 3 broadcast domains for specific protocols. By configuring a port as a member of a protocol VLAN, you establish a forwarding policy for that port.

For example, if you have a port-based VLAN that contains ports 1 – 12, you can configure some or all of the ports in the VLAN as an AppleTalk protocol VLAN. AppleTalk broadcast traffic received on one of the ports in the AppleTalk VLAN is broadcast to the other ports in the AppleTalk VLAN, but not to ports outside the AppleTalk VLAN.

When a port in protocol-based VLAN receives a packet, the device examines the Layer 3 information in the packet to determine whether the packet type is the same as the protocol type of the VLAN.

- If the packet is the same type as the protocol of the VLAN, the device forwards the packet.
- If the packet is another protocol type, the device drops the packet.

For example, when a port in an AppleTalk VLAN receives an AppleTalk packet, the port forwards the packet. The same port drops IPX packets, unless the port also is a member of an IPX VLAN.

IP sub-net and IPX network VLANs are similar, except for these VLAN types the device examines the IP sub-net or IPX network address.

- If the IP sub-net or IPX network address matches the address of the IP sub-net VLAN or IPX network VLAN, the device forwards the packet.
- If the sub-net or network address does not match the VLAN, the device drops the packet.

See the chapter “Configuring Virtual LANs (VLANs)” on page 11-1 for configuration rules and examples.

Actions

A Foundry device forwards a packet if its Layer 3 protocol information matches the protocol VLAN’s protocol type, IP sub-net, or IPX network; otherwise, the policy drops the packet.

Scope

The forwarding policy of a port-based VLAN applies only to that VLAN.

Syntax

Use the following CLI commands to configure VLAN policies.

Table C.7: VLAN Policies

Scope	CLI syntax
VLAN type	FESX424 Router(config)# vlan <vlan-id> by port FESX424 Router(config-vlan-1)# [untagged] ethernet <portnum > [to | ethernet [<slotnum>/]<portnum>]

NOTE: The **untagged** command applies only if you are removing 802.1q tagging from the ports in the VLAN. 802.1q tagging allows a port to be a member of multiple port-based VLANs. Ports in a port-based VLAN are tagged by default. The default tag is 8100 and is a global parameter.

Foundry Filters

A filter is a set of comparison values and an action. If a packet matches the set of values in the filter, the Foundry device takes the action specified in the filter. Foundry devices provide filters for Layer 2, Layer 3, and Layer 4. A filter looks at the appropriate fields in a packet to compare information related to one of the layers. For example, MAC filters look at the source and destination MAC address and, optionally, at the encapsulation information. IPX filters look at the source and destination network and socket information but do not look at the MAC information.

The following table lists the various types of filters you can configure on Foundry FastIron X-Series devices.

Table C.8: Foundry Filters

Filter Type	Supported on...			See page.. .
	FESX	FSX	FWSX	
Layer 2 Filters				C-7
MAC filters	X	X	X	C-7
Address-lock filters	X	X	X	C-8
Layer 3 Filters				C-9
RIP route filters	X	X		C-10
RIP neighbor filters	X	X		C-11
BGP address filters	X	X		C-12
BGP AS-path filters	X	X		C-13
BGP community filters	X	X		C-14
RIP redistribution filters	X	X		C-15
OSPF redistribution filters	X	X		C-16
BGP redistribution filters	X	X		C-16

Layer 2 Filters

Layer 2 filters control a Foundry device's receipt of packets based on MAC address information. Foundry devices provide the following types of Layer 2 filters:

- MAC address filters
- Address-lock filters

MAC Filters

MAC filters forward or drop incoming packets based on the following information:

- Source MAC address
- Destination MAC address
- Encapsulation type and EtherType (optional)

A packet whose Layer 2 information matches the filter is either permitted (forwarded) or denied (dropped). You define a MAC filter on the global level, then apply it to an interface. The filter applies only to incoming traffic on the interface.

NOTE: MAC filters do not block management access to the Foundry device. For example, if you apply a filter to block a specific host, the filter blocks switch traffic from the host but does not prevent the host from establishing a management connection to the device through Telnet. To block management access, use an Access Control List (ACL). See "Software-Based IP Access Control Lists (ACLs)" on page 5-1.

Action

MAC filters forward (permit) or drop (deny) packets.

Scope

You configure MAC filters globally, then apply them to individual ports. The filters do not take effect until applied to specific ports. MAC filters apply only to incoming packets.

Syntax

Use the following CLI commands to configure MAC filters.

Table C.9: MAC Filters

CLI syntax
(config)# mac filter <filter-num> permit | deny any | <H.H.H> any | <H.H.H> etype | llc | snap <operator> <frame-type>
(config-if-1/1)# mac-filter-group <filter-list>

Address-Lock Filters

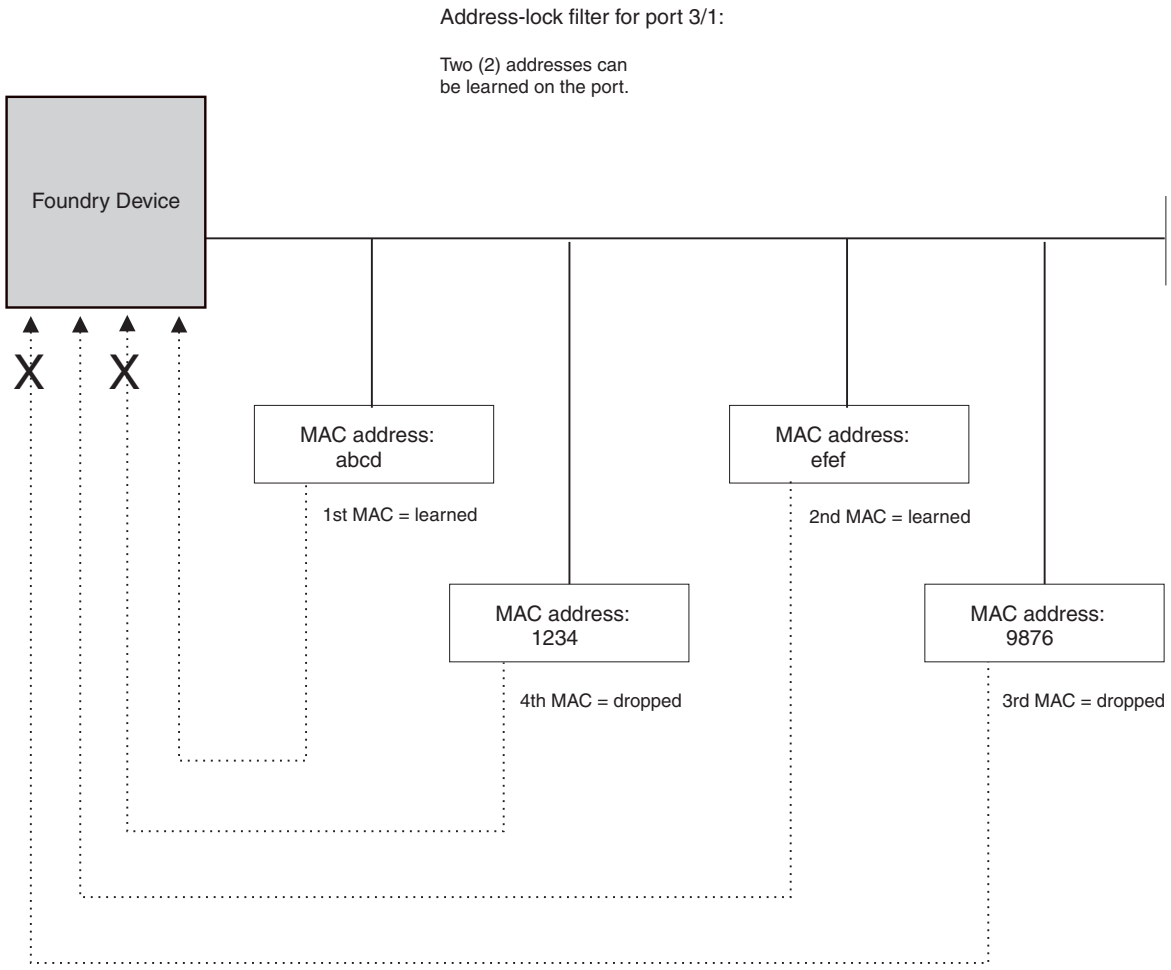
Address-lock filters limit the number of MAC addresses that can be learned on a port. The port forwards only those packets that contain one of the source MAC addresses learned by the port. The port drops other packets. In addition, the device generates an SNMP trap for other packets received by the port.

Figure B.6 shows an example of an address-lock filter. In this example, the Foundry device is configured to learn only two MAC addresses on port 3/1. After the device learns two addresses, port 3/1 can forward only a packet whose source address is one of the two learned addresses. The port drops all other packets. This applies even to MAC broadcasts. If one of the packets learned on the port is not addressed to the MAC broadcast address, the port cannot forward MAC broadcasts.

The device learns MAC addresses from the source-MAC-address field of inbound packets received on the port.

NOTE: The FastIron Edge Switch does not support address-lock filters on static trunk ports or ports on which link-aggregation is enabled.

Figure C.1 Address-lock filter



Actions

Forward (permit) only those packets with a MAC address that the port has learned. Deny all other packets.

Scope

You configure a lock address filter globally, but you also specify the port as part of the filter.

Syntax

Use the following CLI commands to configure address-lock filters.

Table C.10: Address-Lock Filters

CLI syntax
FESX424 Router(config)# lock-address ethernet [<slotnum>/]<portnum> addr-count <num>

Layer 3 Filters

Layer 3 filters control a Foundry device's transmission and receipt of packets based on routing protocol information in the packets. Foundry devices provide the following types of Layer 3 filters:

- RIP route filters

- RIP neighbor filters
- BGP route address filters
- BGP route AS-path filters
- BGP route community filters
- RIP redistribution filters
- OSPF redistribution filters
- BGP redistribution filters

IP Filters

IP filters control the IP packets that the Foundry device sends and receives and the routes that the device learns or advertises. IP forwarding filters (IP Access policies) control transmission and receipt of IP packets, while RIP route and neighbor filters control the routes that the device learns or advertises. Route filters filter on specific network addresses while neighbor filters filter on the IP addresses of the RIP neighbors.

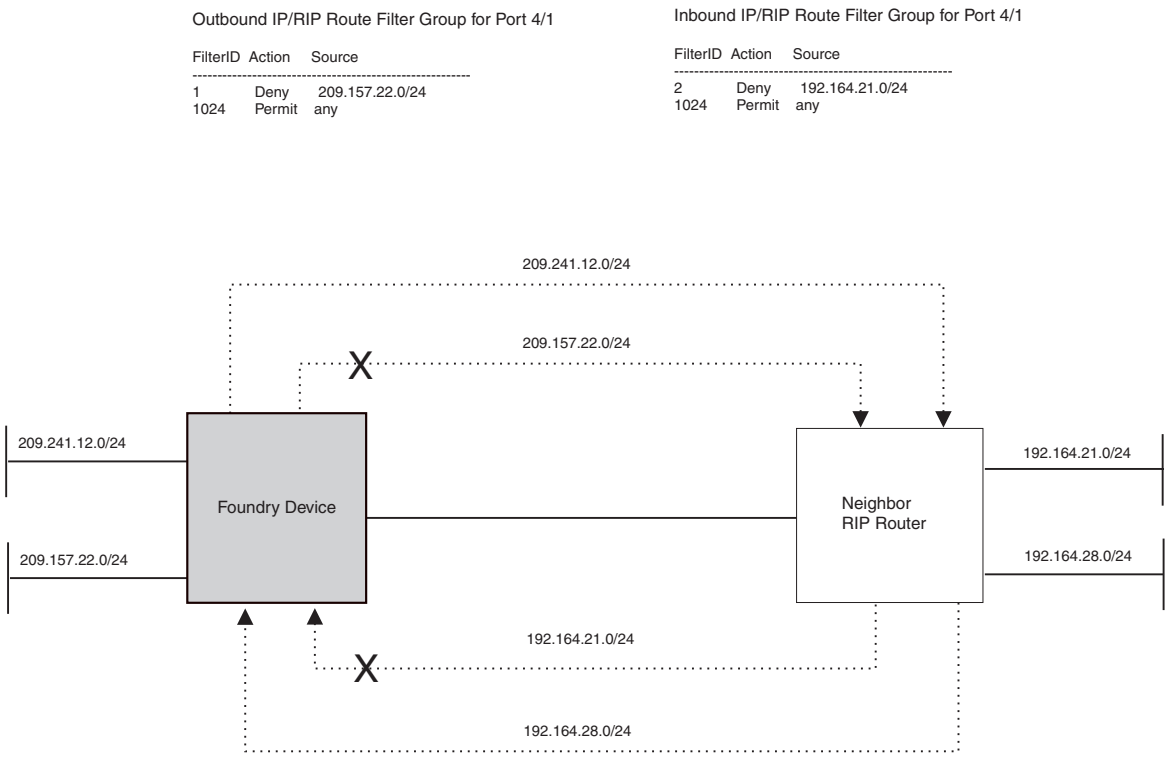
IP Forwarding Filters

IP forwarding filters determine whether to forward or drop an IP packet. IP forwarding filters on a Foundry Layer 2 Switch or Layer 3 Switch are called “IP access policies”. See “IP Access Policies” on page B-10.

RIP Route Filters

RIP route filters control the routes that a Foundry device learns and advertises. Figure B.7 shows an example of a port with RIP route filters. The port has filters for the inbound direction and the outbound direction. Notice that the same filter can be used for both directions. The inbound filters control the routes that the device learns; denied routes are not learned by the device. Outbound filters control the routes that the device advertises; denied routes are not advertised to RIP neighbors.

Figure C.2 RIP route filters



Actions

- A RIP route filter applied to outbound traffic on a port permits or denies advertisement of routes.

- A RIP route filter applied to inbound traffic on a port permits or denies learning of the route. When the device learns an RIP route, the route is added to the RIP route table.

Scope

You configure RIP route filters globally, then apply them to specific ports.

Syntax

Use the following CLI commands to configure RIP route filters.

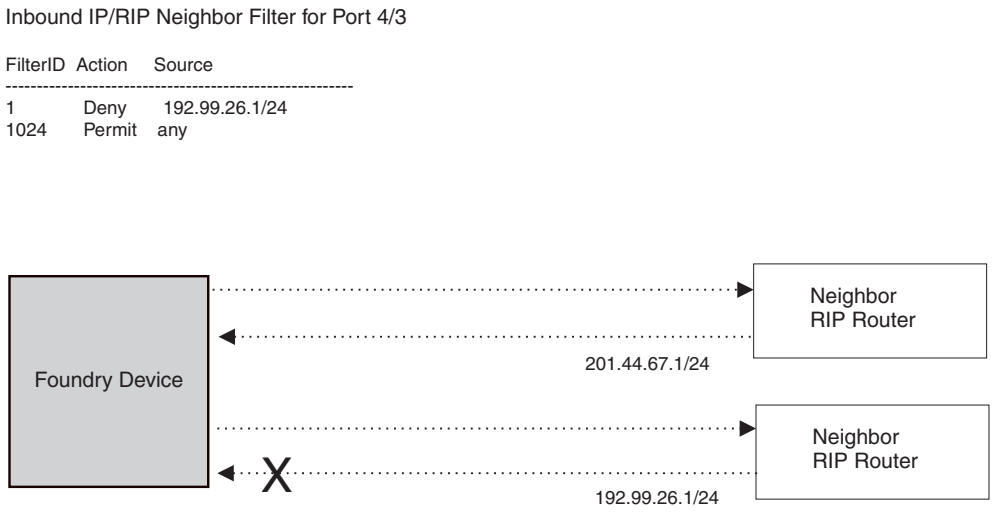
Table C.11: RIP Route Filters

CLI syntax
(config-rip-router)# filter <filter-num> permit | deny <source-ip-address> | any <source-mask> | any
(config-if-1/1)# ip rip filter-group in | out <filter-list>

RIP Neighbor Filters

RIP neighbor filters specify the RIP neighbors the Foundry device can receive updates from or send updates to. You identify the neighbor by specifying its IP address in the filter. Figure B.8 shows an example of an RIP neighbor filter. In this example, the Foundry device is configured to drop all RIP advertisements from the RIP neighbor 192.99.26.1/24. Since this is an outbound filter, the filter does not affect advertisements received by the Foundry device from 192.99.26.1/24. The Foundry device can still learn RIP routes from this neighbor.

Figure C.3 RIP neighbor filters



Actions

- A RIP neighbor filter applied to outbound traffic on a port permits or denies advertisement of routes.
- A RIP neighbor filter applied to inbound traffic on a port permits or denies learning of the routes advertised by the neighbor. When the device learns an RIP route, the route is added to the RIP route table.

Scope

You configure RIP neighbor filters globally. They are automatically applied to all RIP ports as soon as you configure them.

Syntax

Use the following CLI commands to configure RIP neighbor filters.

Table C.12: RIP Neighbor Filters

CLI syntax
FESX424 Router(config-rip-router)# neighbor <filter-num> permit | deny <source-IP-address> | any

BGP4 Filters

Border Gateway Protocol version 4 (BGP4) filters control the routes that a Foundry device learns from BGP4 neighbors and advertises to BGP4 neighbors. You can configure filters to filter route information based on network address, AS-path, or community name.

BGP4 Address Filters

BGP4 address filters control whether the Foundry device learns or drops BGP4 route information based on the route's network address.

Actions

- A BGP4 address filter applied to inbound packets permits (learns) or denies (drops) the specified network address in BGP4 updates received from a BGP4 neighbor.
- A BGP4 address filter applied to outbound packets permits (advertises) or denies (drops) the specified network address in BGP4 updates the Foundry device sends to a BGP4 neighbor.

Scope

You define BGP4 address filters globally, then apply them as part of a BGP4 neighbor's distribute list or as part of a match statement in a route map.

Syntax

Use the following CLI commands to configure BGP4 address filters.

Table C.13: BGP4 Address Filters

CLI syntax
FESX424 Router(config-bgp-router)# address-filter <num> permit | deny <ip-addr> <ip-mask> | any <ip-addr> <ip-mask> | any
FESX424 Router(config-bgp-router)# neighbor <router-id> remote-as <as-number> [advertisement-interval <num>] [distribute-list in | out <num,num,...>] [ebgp-multihop] [filter-list in | out <num,num,...>] [maximum-prefix <num>] [next-hop-self] [remote-as <as-number>] [route-map <map-name>] [send-community] [weight <num>]
FESX424 Router(config-bgp-routemap RMAP_NAME)# match as-path-filters | community-filters | address-filters <num,num,...> [metric <num>] [next-hop <ip-addr>] [route-type internal | external-type1 | external-type2] [tag <tag-value>]

NOTE: The **neighbor** command adds a BGP neighbor. The **distribute-list** parameter specifies a list of address filters and whether the list is applied to inbound or outbound BGP updates.

NOTE: The **match** command compares the information you configure for the command's parameters against BGP routes. You use this command when configuring a route map. If the comparison matches a route, set statements in the route map specify the action to take. See "Defining Route Maps" on page 14-49.

BGP4 AS-Path Filters

BGP4 AS-path filters control whether the Foundry device learns or drops BGP4 route information based on the route's AS-path. The ***AS-path*** is the list of BGP4 autonomous systems (ASs) through which the route information has traveled to reach the Foundry device.

Actions

- A BGP4 AS-path filter applied to inbound packets permits (learns) or denies (drops) routes for networks with the specified AS-path in BGP4 updates received from a BGP4 neighbor.
- A BGP4 AS-path filter applied to outbound packets permits (advertises) or denies (drops) routes for networks with the specified AS-path in BGP4 updates sent to a BGP4 neighbor.

Scope

You define BGP4 AS-path filters globally, then apply them as part of a BGP4 neighbor's distribute list or as part of a match statement in a route map.

Syntax

Use the following CLI commands to configure BGP4 AS-path filters.

Table C.14: BGP4 AS-Path Filters

CLI syntax
FESX424 Router(config-bgp-router)# as-path-filter <num> permit | deny <as-path>
FESX424 Router(config-bgp-router)# neighbor <router-id> remote-as <as-number> [advertisement-interval <num>] [distribute-list in | out <num,num,...>] [ebgp-multihop] [filter-list in | out <num,num,...>] [maximum-prefix <num>] [next-hop-self] [remote-as <as-number>] [route-map <map-name>] [send-community] [weight <num>]
FESX424 Router(config-bgp-routemap RMAP_NAME)# match as-path-filters | community-filters | address-filters <num,num,...> [metric <num>] [next-hop <ip-addr>] [route-type internal | external-type1 | external-type2] [tag <tag-value>]

NOTE: The <as-path> value can be a regular expression. See "Using Regular Expressions" on page 14-44.

NOTE: The **neighbor** command adds a BGP neighbor. The **filter-list** parameter specifies a list of AS-path filters and whether the list is applied to inbound or outbound BGP updates.

NOTE: The **match** command compares the information you configure for the command's parameters against BGP routes. You use this command when configuring a route map. If the comparison matches a route, set statements in the route map specify the action to take. See "Defining Route Maps" on page 14-49.

BGP4 Community Filters

BGP4 community filters control whether the Foundry device learns or drops BGP4 route information based on the route's community membership.

Actions

- A BGP4 community filter applied to inbound packets permits (learns) or denies (drops) routes for networks with the specified community membership in BGP4 updates received from a BGP4 neighbor.
- A BGP4 AS-path filter applied to outbound packets permits (advertises) or denies (drops) routes for networks with the specified community membership in BGP4 updates sent to a BGP4 neighbor.

Scope

You define BGP4 community filters globally, then apply them as part of a BGP4 neighbor's distribute list or as part of a match statement in a route map.

Syntax

Use the following CLI commands to configure BGP4 community filters.

Table C.15: BGP4 Community Filters

CLI syntax
FESX424 Router(config-bgp-router)# community-filter <filter-num> permit | deny <num> | internet | no-advertise | no-export
FESX424 Router(config-bgp-routemap RMAP_NAME)# match as-path-filters | community-filters | address-filters <num,num,...> [metric <num>] [next-hop <ip-addr>] [route-type internal | external-type1 | external-type2] [tag <tag-value>]

NOTE: The **match** command compares the information you configure for the command's parameters against BGP routes. You use this command when configuring a route map. If the comparison matches a route, set statements in the route map specify the action to take. See "Defining Route Maps" on page 14-49.

Redistribution Filters

Redistribution filters control the exchange of routes between routing protocols. RIP, OSPF, and BGP4 support redistribution of one another's routes. In addition, they all allow exchange of static routes.

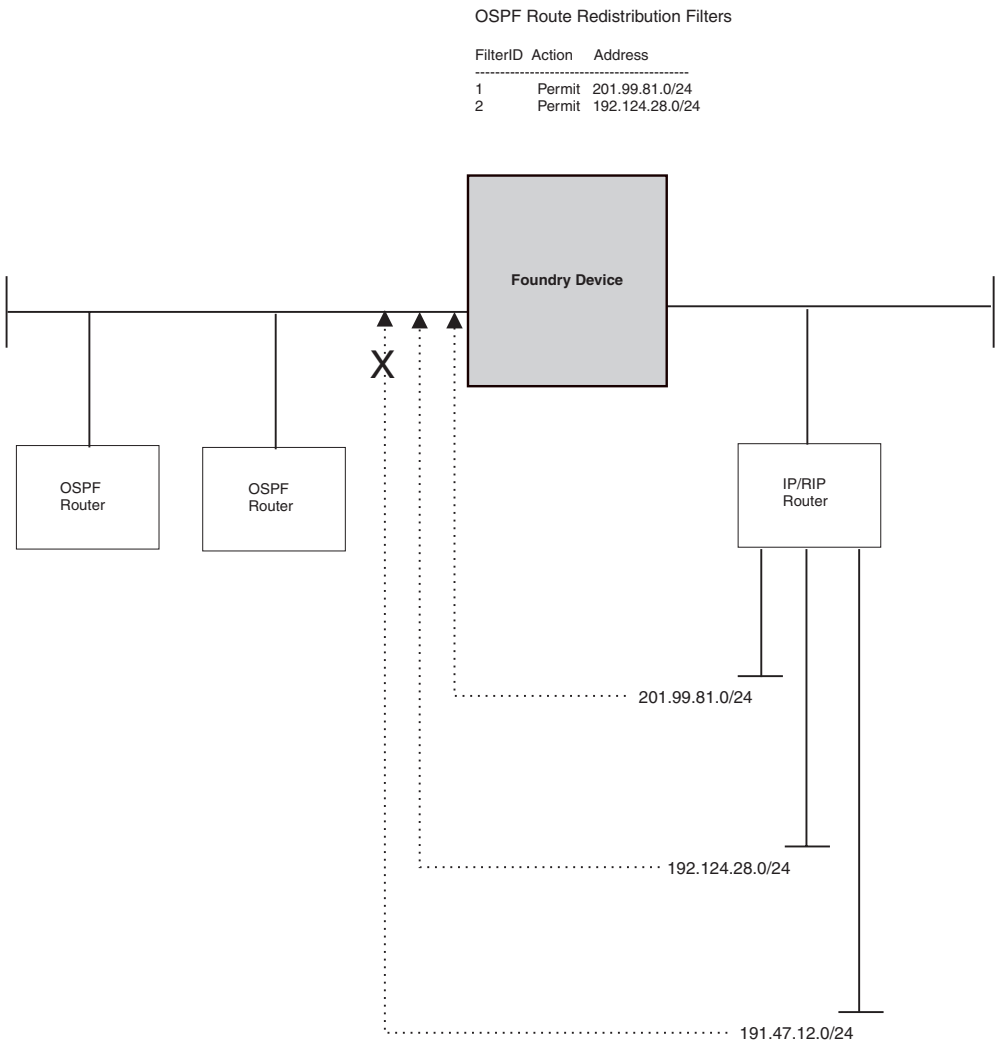
You configure RIP and OSPF redistribution filters to permit or deny routes for specific network addresses. Optionally, you can also filter on and modify the route metric. To configure redistribution, you configure redistribution filters in the protocol that will receive the routes. Redistribution is disabled by default in RIP and OSPF and enabled by default in BGP4.

BGP4 redistribution filters can filter based on a route's metric, weight, and also on the results of comparison of the route information with a route map. A ***route map*** is a named set of match conditions and parameter settings that a Foundry Layer 3 Switch can use to modify route attributes and to control redistribution of routes. For more information, see "Defining Route Maps" on page 14-49.

BGP4 allows you to include the redistribution filters as part of a route map. A route map examines and modifies route information exchanged between BGP4 and RIP or OSPF. See "Configuring BGP4" on page 14-1 for more information.

Figure B.10 shows an example of a redistribution filter. In this example, redistribution filters in OSPF are configured to redistribute two RIP routes into OSPF. Notice that unlike some other filter examples in this appendix, a filter for permitting all routes (to change the default action) is not configured. The default redistribution action is permit, even after you configure a redistribution filter. To maintain tight control over redistribution, define a "deny any" redistribution filter as the last filter (the one with the highest ID) and deny permit filters for specific routes.

Figure C.4 OSPF redistribution filters



RIP Redistribution Filters

RIP redistribution filters control redistribution of routes from other protocols into RIP. A Foundry device running RIP can redistribute static routes, OSPF routes, and BGP4 routes (if BGP4 is supported on the device) into RIP.

Optionally, you can specify a metric that the route must match or you can set the metric on redistributed routes. By setting the metric, you can cause the router to prefer RIP routes or redistributed routes to the specified network.

Actions

RIP redistribution filters permit (redistribute) or deny (do not redistribute) OSPF or BGP4 routes into RIP.

Scope

You configure RIP redistribution filters globally. They are automatically applied as soon as you configure them.

Syntax

Use the following CLI commands to configure RIP redistribution filters.

Table C.16: RIP Redistribution Filters

CLI syntax
FESX424 Router(config-rip-router)# permit | deny redistribute <filter-num> all | bgp | ospf | static address <ip-addr> <ip-mask> [match-metric <value> | set-metric <value>]

OSPF Redistribution Filters

OSPF redistribution filters control redistribution of routes from other protocols into OSPF. A Foundry device running OSPF can redistribute static routes, RIP routes, and BG4P routes (if BGP4 is supported on the device) into OSPF.

Optionally, you can specify a metric that the route must match or you can set the metric on redistributed routes. By setting the metric, you can cause the router to prefer OSPF routes or redistributed routes to the specified network.

Actions

OSPF redistribution filters permit (redistribute) or deny (don't redistribute) RIP or BGP4 routes into OSPF.

Scope

You configure and apply OSPF redistribution filters globally.

Syntax

Use the following CLI commands to configure OSPF redistribution filters.

Table C.17: OSPF Redistribution Filters

CLI syntax
FESX424 Router(config-ospf-router)# deny | permit redistribute <filter-num> all | bgp | rip | static address <ip-addr> [match-metric <value> | set-metric <value>]

BGP4 Redistribution Filters

BGP4 redistribution filters control redistribution of routes from other protocols into BGP4. A Foundry device running BGP4 can redistribute static routes, RIP routes, and OSPF routes into BGP4.

Optionally, you can modify a route's metric and weight and use a route map to change additional attributes of the route.

Actions

BGP4 redistribution filters permit (redistribute) or deny (don't redistribute) RIP or OSPF routes into RIP.

Scope

You configure and apply BGP4 redistribution filters globally.

Syntax

Use the following CLI commands to configure BGP4 redistribution filters.

Table C.18: BGP4 Redistribution Filters

CLI syntax
FESX424 Router(config-bgp-router)# redistribute rip | ospf | static [match internal | external1 | external2] [metric <num>] [route-map <name>] [weight <num>]

NOTE: The optional **match internal | external1 | external2** argument applies only to OSPF.

Appendix D
Software Features and Specifications

This appendix lists the following information:

- Feature support
- IEEE compliance
- RFC support
- ISO/IEC specification support
- Internet draft support

NOTE: For a list of features supported on a specific product, see the data sheet for that product.

Feature Highlights

The FESX, and FSX support many of the applicable system-level, Layer 2, and Layer 3 features supported on the BigIron Chassis devices. The FWSX supports system-level and Layer 2 features only. It does not support base Layer 3 and full Layer 3 features. The features that are available depend on the type of software image the device is running. You can run one of the following types of software images on these devices:

- Layer 2 (supported on all models)
- Base Layer 3 (supported on the FESX and FSX)
- Full Layer 3 (supported on FESX and FSX premium models only)

Table D.1 lists the software that is loaded into the device's primary and secondary flash areas at the factory. All the flash images are included on the CD-ROM shipped with the device.

Table D.1: Default Software Loads

Model	Software Images	
	Primary Flash	**Secondary Flash**
FESX424, FESX424HF FESX448 FSX	Layer 2	Base Layer 3
FESX424-PREM FESX448-PREM FSX-PREM	Full Layer 3	Layer 2
FWSX424 FWSX448	Layer 2	Layer 2

Supported Features

Table D.2 lists the feature highlights in the FSX, FESX, and FWSX software.

Table D.2: List of Supported Features

Category, Description, and Configuration Notes	Supported on		
	FSX	**FESX**	**FWSX**
Management Features			
Access Control Lists (ACLs) for controlling management access	X	X	X
IronView Network Manager (optional standalone and HP OpenView GUI)	X	X	X
Serial and Telnet access to industry-standard Command Line Interface (CLI)	X	X	X
SNMP V1, V2, V3	X	X	X
Web-based GUI	X	X	X
Security Features			
802.1X port security • Dynamic assignment for ACL, MAC filter, and VLAN	X	X	X

Table D.2: List of Supported Features (Continued)

Category, Description, and Configuration Notes	Supported on		
	FSX	FESX	FWSX
Access Control Lists (ACLs) for filtering transit traffic (applies to IP unicast traffic only) • Support for inbound ACLs only. These devices do not support outbound ACLs.	X	X	X
Address locking	X	X	X
Authentication, Authorization and Accounting (AAA) • RADIUS, TACACS/TACACS+	X	X	X
Denial of Service (DoS) protection • SYN Attacks and Smurf Attacks	X	X	X
Layer 2 MAC filtering • Filtering on source and destination MAC addresses	X	X	X
Local passwords	X	X	X
MAC port security	X	X	X
Secure Shell (SSH) version 1.5	X	X	X
User accounts	X	X	X
System Level Features			
802.3ad link aggregation (dynamic trunk groups) • Foundry ports follow the same configuration rules for dynamically created trunk groups as they do for statically configured trunk groups.	X	X	X
Auto MDI/MDIX	X	X	X
Broadcast, multicast, and unknown-unicast rate limiting	X	X	X
DiffServ support	X	X	X
Foundry Discovery Protocol (FDP) / Cisco Discovery Protocol (CDP)	X	X	X
Jumbo frames • Supported in Gigabit products only (FESX and FWSX) • Up to 9216 bytes on FSX, FESX, and FWSX	X	X	X
Mini jumbo frames • FES support only, starting with release 03.2.00 • Up to 2048 bytes			
Multiple Syslog server logging • Up to six Syslog servers	X	X	X

Table D.2: List of Supported Features (Continued)

Category, Description, and Configuration Notes	Supported on		
	FSX	FESX	FWSX
OSPF Version 2 MIB • RFC 1850 • FESX support starts in release 02.0.00	X	X	
Port mirroring and monitoring • Mirroring of both inbound and outbound traffic on individual ports is supported.	X	X	X
Priority mapping using ACLs • ToS-QoS mapping for *routed packets* using ACLs is supported. ToS-QoS mapping for *switched packets* is not supported.	X	X	X
Rate limiting • Port-based *inbound* rate limiting is supported. Port-based *outbound* rate limiting is not supported. • FESX support starts in release 01.1.00 • Fixed rate limiting is not supported on 10-Gigabit Ethernet ports.	X	X	X
sFlow • RFC 3176 • FESX, FWSX, and FSX support sFlow sampling of *inbound traffic* only. These devices do not sample *outbound packets*.	X	X	X
Static MAC entries with option to set priority	X	X	X
Trunk groups • FESX, FWSX, and FSX devices support up to 4-port trunk groups (trunk groups on these devices can have 2, 3, or 4 ports)	X	X	X
Layer 2 Features			
802.1d Spanning Tree Support • Enhanced IronSpan support includes Fast Port Span and Single-instance Span • PVST/PVST+ compatibility • Rapid Spanning Tree support allows for sub-second convergence	X	X	X
802.1p Quality of Service (QoS) • Strict Priority (SP) • Weighted Round Robin (WRR) • Support of 8 priority queues	X	X	X
802.1W Rapid Spanning Tree • Final IEEE standard	X	X	X

Table D.2: List of Supported Features (Continued)

Category, Description, and Configuration Notes	Supported on		
	FSX	FESX	FWSX
Dynamic Host Configuration Protocol (DHCP) Assist	X	X	X
IGMPv2 snooping (Layer 2 Multicast)	X	X	X
Metro Ring Protocol 1 (MRP 1) • These devices can be MRP masters or MRP members (for different rings). • The RHP received counter on non-master MRP nodes increment. This is different on other devices that support MRP 1.	X	X	X
Topology groups	X	X	X
Uni-directional Link Detection (UDLD) (Link keepalive)	X	X	X
Virtual Cable Testing (VCT) technology • FESX support starts in release 01.1.00	X	X	X
Virtual Switch Redundancy Protocol (VSRP)	X	X	X
VLAN Support:	X	X	X
• 802.1Q with tagging	X	X	X
• 802.1Q-in-Q Super Aggregated VLANs (SAVs) FESX support starts in release 01.1.00	X	X	X
• Dual-mode VLANs	X	X	X
• GVRP	X	X	X
• Private VLANS			
• Protocol VLANs (IPv4 and dynamic IPv6)	X	X	X
• Layer 3 Subnet VLANs (IP subnet network)	X	X	
• Super Aggregated VLANs	X	X	X
• Virtual routing interfaces	X	X	X
• VLAN groups	X	X	X
Wire-speed Layer 2 Switching	X	X	X
Base Layer 3 Features			
BGP • FESX support starts in release 02.1.01. • FSX support starts in release 02.2.00.	X	X	
RIP V1 and V2	X	X	
Routing for directly connected IP subnets	X	X	
• Static IP	X	X	

Table D.2: List of Supported Features (Continued)

Category, Description, and Configuration Notes	Supported on		
	FSX	FESX	FWSX
• Virtual Interfaces	X	X	
Full Layer 3 Features • FSX support starts in release 02.2.00.			
BGP4 • FESX support starts in release 02.1.01	X	X	
IGMP V1 and V2 • FESX support starts in release 02.0.00.	X	X	
IP • FESX support starts in release 02.0.00.	X	X	
IP multicast (DVMRP, PIM-SM, PIM-DM) • FESX support starts in release 02.0.00. • Layer 3 Switches support up to 1024 PIM groups and 1024 DVMRP groups by default.	X	X	
OSPF • FESX support starts in release 02.0.00.	X	X	
RIP V1 and V2 • FESX support starts in release 02.0.00.	X	X	
Route-only support • FSX devices support disabling Layer 2 Switching at the CLI Interface level as well as the Global CONFIG level. • FESX support starts in release 02.0.00. Release 02.0.00 supports disabling Layer 2 Switching on a global basis only. Release 02.1.01 supports disabling Layer 2 Switching on an individual interface and on a global basis.	X	X	
VRRP and VRRPE • FESX support starts in release 02.0.00.	X	X	

NOTE: Full Layer 3 features are supported on Premium devices only.

Unsupported Features

Table D.3 lists the features that are not supported on the FSX, FESX, and FWSX. If required, these features are available on other Foundry devices.

Table D.3: List of Unsupported Features

System Level Features not Supported	Not Supported on		
	FSX	FESX	FWSX
ACL logging • The FSX supports ACL logging for packets that are sent to the CPU for processing.		X	X
ACL statistics	X	X	X
Broadcast and multicast filters	X	X	X
Jumbo frames (on all models except FES12GCF)	X		
NetFlow	X	X	X
Outbound ACLs	X	X	X
Outbound rate limiting	X	X	X
Protected link groups	X	X	X
Server trunk groups for Layer 3 traffic • Server trunking of switched AppleTalk traffic also not supported	X	X	X
Layer 2 Features not Supported			
SuperSpan	X	X	X
VLAN-based priority	X	X	X
Layer 3 Features not Supported			
AppleTalk	X	X	X
Base Layer 3 features			X
BGP			X
Foundry Standby Router Protocol (FSRP)	X	X	X
IPX	X	X	X
IS-IS	X	X	X
Multiprotocol Border Gateway Protocol (MBGP)	X	X	X
Multiprotocol Label Switching (MPLS)	X	X	X
Multiprotocol Source Discovery Protocol (MSDP)	X	X	X
Network Address Translation (NAT)	X	X	X
Policy-Based Routing (PBR)	X	X	X

IEEE Compliance

Foundry devices support the following standards.

Table D.4: IEEE Compliance

Standard	Description	FESX	FSX	FWSX
802.1d	Bridging	X	X	X
802.1D	1998	X	X	X
802.1p/q	VLAN Tagging and Priority	X	X	X
802.1w	Rapid Spanning Tree	X	X	X
802.1x	Port-based Authentication, Dynamic VLAN, ACL, and MAC Filter Group Assignment	X	X	X
802.3	10Base-T	X	X	X
802.3	Ethernet Like MIB	X	X	X
802.3ab	1000Base-T		X	
802.3ad	Link Aggregation (Dynamic and Static) and Trunk Groups	X	X	X
802.3ae	10 Gigabit Ethernet	X	X	X
802.3af	Power over Ethernet	X	X	
802.3u	100Base-TX, 1000Base-FX	X	X	X
802.3z	1000Base-SX, 1000Base-LX, 1000Base-T	X	X	X
802.3x	Flow Control	X	X	X
	Ethernet Interface MIB	X	X	X
	PVSTP/+	X	X	X
	Repeater MIB	X	X	X
	SNMP MIB II	X	X	X
	SNMP V1, V2c, and V3	X	X	X

RFC Support

The following table lists the RFCs supported by Foundry devices.

NOTE: Some devices support only a subset of the RFCs. For example, Layer 2 Switches do not support router-specific RFCs. For a list of features supported on your device, see the data sheet or the software release notes for the version of software running on your device.

Table D.5: Foundry RFC Support

RFC Number	Protocol or Standard	FESX	FSX	FWSX
768	User Datagram Protocol (UDP)	X	X	
783	Trivial File Transfer Protocol (TFTP)	X	X	X
791	Internet Protocol (IP)	X	X	
792	Internet Control Message Protocol (ICMP)	X	X	
793	Transmission Control Protocol (TCP)	X	X	
826	Ethernet Address Resolution Protocol (ARP)	X	X	
854, 855, and 857	Telnet	X	X	X
894	IP over Ethernet frames	X	X	
903	Reverse ARP (RARP)	X	X	
906	Bootstrap loading using TFTP	X	X	
919	Broadcast Internet datagrams	X		
920	Domain requirements	X		
922	Broadcast Internet datagrams in the presence of subnets	X		
950	Internet standard subnetting procedure	X		
951	Bootstrap Protocol (BootP)	X	X	X
1027	Proxy ARP	X	X	
1042	IP datagrams over IEEE 802 networks (for Ethernet)	X		
1058	Route Information Protocol (RIP) version 1	X	X	
1075	Distance Vector Multicast Routing Protocol	X		
1112	Internet Gateway Management Protocol (IGMP) version 1	X	X	
1122 and 1123	Requirements for Internet hosts (routers)	X	X	X
1141	Incremental updating of the Internet checksum	X		
1155	Structure and Identification of Management Information (SMI)	X	X	X
1157	Simple Network Management Protocol (SNMP) version 1	X	X	X
1191	Path MTU Discovery	X	X	

Table D.5: Foundry RFC Support (Continued)

RFC Number	Protocol or Standard	FESX	FSX	FWSX
1212	Concise MIB Definitions	X	X	
1213	MIB II Definitions	X	X	X
1215	SNMP generic traps	X	X	X
1256	ICMP Router Discovery Protocol (IRDP)	X	X	X
1267	Border Gateway Protocol version 3	X	X	
1269	Definitions of Managed Objects for the Border Gateway Protocol: Version 3	X	X	
1321	The MD5 Message-Digest Algorithm	X	X	
1340	Assigned numbers (where applicable)	X	X	
1354	IP Forwarding Table MIB	X	X	X
1377	The PPP OSI Network Layer Control Protocol (OSINLCP)	X		
1398	Ethernet-Like MIB	X	X	
1492	An Access Control Protocol, Sometimes Called TACACS	X	X	
1493	Bridge MIB (excluding filtering of objects)	X	X	X
1516	Repeater MIB	X		X
1519	Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy	X	X	
1541 and 1542	Dynamic Host Configuration Protocol (DHCP)	X	X	X
1573	SNMP MIB II	X	X	X
1583	Open Shortest Path First (OSPF)	X	X	
1587	OSPF Not-So-Stubby Areas (NSSAs)	X	X	
1591	Domain Name System Structure and Delegation	X	X	
1643	Ethernet Interface MIB	X	X	X
1657	Definitions of Managed Objects for the Fourth Version of the Border Gateway Protocol (BGP4) using SMIv2	X	X	
1661	The Point-to-Point Protocol (PPP)	X	X	
1723	RIP version 2	X	X	
1724	RIP version 2 MIB	X	X	X
1742	AppleTalk Management Information Base II	X	X	
1745	OSPF Interactions	X	X	
1757	Remote Monitoring (RMON) groups 1, 2, 3, 9	X	X	X
1765	OSPF Database Overflow	X	X	
1771	Border Gateway Protocol version 4 (BGP4)	X	X	

Table D.5: Foundry RFC Support (Continued)

RFC Number	Protocol or Standard	FESX	FSX	FWSX
1812	Requirements for IP version 4 routers	X	X	
1850	OSPF version 2 MIB	X	X	
1905	Protocol Operations for version 2 of the Simple Network Management Protocol (SNMPv2)	X	X	
1906	Transport Mappings for version 2 of the Simple Network Management Protocol (SNMPv2)	X	X	
1965	Autonomous System Configurations for BGP4	X	X	
1966	BGP Route Reflection	X	X	
1997	BGP Communities Attributes	X	X	
2003	IP Tunneling	X		
2011	SNMPv2 Management Information Base for the Internet Protocol using SMIv2	X	X	
2012	SNMPv2 Management Information Base for the Transmission Control Protocol using SMIv2	X	X	
2013	SNMPv2 Management Information Base for the User Datagram Protocol using SMIv2	X	X	
2030	Simple Network Time Protocol (SNTP) version 4	X	X	
2068	HTTP	X	X	X
2096	IP Forwarding MIB	X	X	
2131	BootP/DHCP Relay	X		X
2138	Remote Authentication Dial In User Server (RADIUS)	X	X	X
2139	RADIUS Accounting	X	X	
2154	OSPF with Digital Signatures (Password, MD-5)	X	X	
2178	Open Shortest Path First (OSPF)	X	X	
2205	Resource ReSerVation Protocol (RSVP) -- version 1 Functional Specification	X	X	
2233	The Interfaces Group MIB using SMIv2	X	X	
2236	Internet Gateway Management Protocol (IGMP) version 2	X	X	
2239	802.3 Medium Attachment Units (MAUs) using SMIv2	X	X	X
2283	Multiprotocol Extensions for BGP4	X	X	
2328	OSPF version 2 **Note**: AS External LSA reduction is supported.	X	X	
2336	IGMP version 2	X	X	
2338	Virtual Router Redundancy Protocol (VRRP)	X	X	

Table D.5: Foundry RFC Support (Continued)

RFC Number	Protocol or Standard	FESX	FSX	FWSX
2362	IP Multicast PIM Sparse	X	X	
2370	The OSPF Opaque LSA Option	X	X	
2385	TCP MD5 Signature Option (for BGP4)	X	X	
2439	BGP Route Flap Dampening	X	X	
2482	Language Tagging in Unicode Plain Text	X	X	
2570	Introduction to version 3 of the Internet-standard Network Management Framework	X	X	X
2571	An Architecture of Describing SNMP Management Frameworks	X	X	X
2572	Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)	X	X	X
2573	SNMP version 3 Applications	X		X
2574	User-based Security (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)	X	X	X
2575	View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)	X	X	X
2576	Coexistence between Version 1, Version 2, and Version 3 of the Internet-standard Network Management Framework	X	X	
2578	Structure of Management Information Version 2 (SMIv2)	X	X	
2579	Textual Conventions for SMIv2	X	X	
2580	Conformance Statements for SMIv2	X	X	
2665	Ethernet Like MIB (incorporates RFC 1398)	X	X	
2674	Definitions of Managed Objects for Bridges with Traffic Classes, Multicast Filtering and Virtual LAN Extensions	X	X	
2787	Definitions of Managed Objects for the Virtual Router Redundancy Protocol	X	X	
2796	BGP Route Reflection	X	X	
2842	BGP Capability Advertisement	X	X	
2865	Remote Authentication Dial In User Service (RADIUS)	X	X	
2866	RADIUS Accounting	X	X	
2869	RADIUS Extensions	X	X	
2918	Route Refresh Capability for BGP4	X	X	
2932	IPv4 Multicast Routing MIB	X	X	
2933	Internet Group Management Protocol MIB	X	X	
2934	Protocol Independent Multicast MIB for IPv4	X	X	

Table D.5: Foundry RFC Support (Continued)

RFC Number	Protocol or Standard	FESX	FSX	FWSX
3176	InMon Corporation's sFlow: A Method for Monitoring Traffic in Switched and Routed Networks	X	X	X
3411	Simple Network Management Protocol (SNMP) Management Frameworks	X	X	X
3412	Message Processing and Dispatching for the Simple Network Management Protocol (SNMP)	X	X	X
3414	User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)	X	X	X
3415	View-based Access Control Model (VACM) for the Simple Network Management Protocol (SNMP)	X	X	X
3418	Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)	X	X	X
	AAA	X	X	X
	Bi-level access mode (standard and EXEC level)	X	X	X
	DVMRP V3-07	X	X	
	HTTP and HTTPS	X	X	X
	IGMP Snooping (versions 1, 2, and 3)	X	X	X
	IGMP version 3	X	X	
	Integrated standard-based Command Line Interface (CLI)	X	X	X
	IronView Network Manager (INM) web-based graphical user interface	X	X	X
	MRP	X	X	
	PIM-DM V1	X	X	
	PIM-SSM	X	X	
	Protection for Denial of Service attacks, such as TCP SYN or Smurf Attacks	X	X	X
	RMON HP OpenView for Sun Solaris, HP-UX, IBM's AIX, and Windows NT	X	X	X
	Secure Copy (SCP)	X	X	X
	SSH V 1.5	X	X	X
	TACACS/TACACS+	X	X	X
	TELNET and SSH V1	X	X	X
	UDLD	X	X	X
	Username/Password (challenge and response)	X	X	X

Internet Drafts

In addition to the RFCs listed in “RFC Support” on page D-9, the Layer 3 Switches support the following Internet drafts:

- ietf-idmr-dvmrp version 3.05, obsoletes RFC 1075
- draft-ietf-pim-dm-05 (V1)
- draft-ietf-pim-v2-dm-03 (V2)
- draft-katz-yeung-ospf-traffic-03.txt
- TACACS+ Protocol version 1.78

NOTE: Foundry supports the portions of this draft that describe the Extended IP reachablity TLV (TLV type 135) and the extended Intermediate System (IS) reachability TLV (TLV type 22) to provide support for wide metrics.

Appendix E
Cautions and Warnings

The cautions and warnings that appear in this manual are listed below in English, German, French, and Spanish.

Cautions

A caution calls your attention to a possible hazard that can damage equipment.

"Vorsicht" weist auf eine mögliche Beschädigung des Geräts hin. Sie finden die folgenden Vorsichtshinweise in diesem Handbuch.

Une mise en garde attire votre attention sur un risque possible d'endommagement de l'équipement. Ci-dessous, vous trouverez les mises en garde utilisées dans ce manuel.

Un mensaje de precaución le advierte sobre un posible peligro que pueda dañar el equipo. Las siguientes son precauciones utilizadas en este manual.

CAUTION:	Carefully follow the mechanical guides on each side of the power supply slot and make sure the power supply is properly inserted in the guides. Never insert the power supply upside down.
VORSICHT:	Beachten Sie mechanischen Führungen an jeder Seite des Netzteils, das ordnungegemäß in die Führungen gesteckt werden muss. Das Netzteil darf niemals umgedreht eingesteckt werden.
MISE EN GARDE:	Suivez attentivement les repères mécaniques de chaque côté du slot du bloc d'alimentation et assurez-vous que le bloc d'alimentation est bien inséré dans les repères. N'insérez jamais le bloc d'alimentation à l'envers.
PRECAUCIÓN:	Siga cuidadosamente las guías mecánicas de cada lado de la ranura del suministro de energía y verifique que el suministro de energía está insertado correctamente en las guías. No inserte nunca el suministro de energía de manera invertida.

CAUTION:	Remove the power cord from a power supply before you install it in or remove it from the device. Otherwise, the power supply or the device could be damaged as a result. (The device can be running while a power supply is being installed or removed, but the power supply itself should not be connected to a power source.)
VORSICHT:	Nehmen Sie vor dem Anschließen oder Abtrennen des Geräts das Stromkabel vom Netzteil ab. Ansonsten könnten das Netzteil oder das Gerät beschädigt werden. (Das Gerät kann

während des Anschließens oder Annehmens des Netzteils laufen. Nur das Netzteil sollte nicht an eine Stromquelle angeschlossen sein.)

MISE EN GARDE: Enlevez le cordon d'alimentation d'un bloc d'alimentation avant de l'installer ou de l'enlever du dispositif. Sinon, le bloc d'alimentation ou le dispositif risque d'être endommagé. (Le dispositif peut être en train de fonctionner lorsque vous installez ou enlevez un bloc d'alimentation, mais le bloc d'alimentation lui-même ne doit pas être connecté à une source d'alimentation.)

PRECAUCIÓN: Retire el cordón de corriente del suministro de corriente antes de instalarlo o retírarlo del instrumento. De no hacerse así, el suministro de corriente o el instrumento podrían resultar dañados. (El instrumento puede estar encendido mientras se instala o retira un suministro de corriente, pero el suministro de corriente en sí no deberá conectado a la corriente).

CAUTION: Do not install the device in an environment where the operating ambient temperature might exceed 40o C (104o F).

VORSICHT: Das Gerät darf nicht in einer Umgebung mit einer Umgebungsbetriebstemperatur von über 40° C (104° F) installiert werden.

MISE EN GARDE: N'installez pas le dispositif dans un environnement où la température d'exploitation ambiante risque de dépasser 40° C (104° F).

PRECAUCIÓN: No instale el instrumento en un entorno en el que la temperatura ambiente de operación pueda exceder los 40oC (104oF).

CAUTION: Make sure the air flow around the front, sides, and back of the device is not restricted.

VORSICHT: Stellen Sie sicher, dass an der Vorderseite, den Seiten und an der Rückseite der Luftstrom nicht behindert wird.

MISE EN GARDE: Vérifiez que rien ne restreint la circulation d'air devant, derrière et sur les côtés du dispositif et qu'elle peut se faire librement.

PRECAUCIÓN: Asegúrese de que el flujo de aire en las inmediaciones de las partes anterior, laterales y posterior del instrumento no esté restringido.

CAUTION: Use a separate branch circuit for each AC power cord, which provides redundancy in case one of the circuits fails.

VORSICHT: Es empfiehlt sich die Installation eines separaten Stromkreiszweiges für jede Wechselstrom-Elektroschnur als Redundanz im Fall des Ausfalls eines Stromkreises.

MISE EN GARDE: Utilisez un circuit de dérivation différent pour chaque cordon d'alimentation C.A. Ainsi, il y aura un circuit redondant en cas de panne d'un des circuits.

PRECAUCIÓN: Use un circuito derivado separado para cada cordón de alimentación de CA, con lo que se proporcionará redundancia en caso de que uno de los circuitos falle.

CAUTION: Ensure that the device does not overload the power circuits, wiring, and over-current protection. To determine the possibility of overloading the supply circuits, add the ampere (amp) ratings of all devices installed on the same circuit as the device. Compare this total with the rating limit for the circuit. The maximum ampere ratings are usually printed on the devices near the input power connectors.

VORSICHT: Stromkreise, Verdrahtung und Überlastschutz dürfen nicht durch das Gerät überbelastet werden. Addieren Sie die Nennstromleistung (in Ampere) aller Geräte, die am selben Stromkreis wie das Gerät installiert sind. Somit können Sie feststellen, ob die Gefahr einer Überbelastung der Versorgungsstromkreise vorliegt. Vergleichen Sie diese Summe mit der

Nennstromgrenze des Stromkreises. Die Höchstnennströme (in Ampere) stehen normalerweise auf der Geräterückseite neben den Eingangsstromanschlüssen.

MISE EN GARDE: Assurez-vous que le dispositif ne risque pas de surcharger les circuits d'alimentation, le câblage et la protection de surintensité. Pour déterminer le risque de surcharge des circuits d'alimentation, additionnez l'intensité nominale (ampères) de tous les dispositifs installés sur le même circuit que le dispositif en question. Comparez alors ce total avec la limite de charge du circuit. L'intensité nominale maximum en ampères est généralement imprimée sur chaque dispositif près des connecteurs d'entrée d'alimentation.

PRECAUCIÓN: Verifique que el instrumento no sobrecargue los circuitos de corriente, el cableado y la protección para sobrecargas. Para determinar la posibilidad de sobrecarga en los circuitos de suministros, añada las capacidades nominales de corriente (amp) de todos los instrumentos instalados en el mismo circuito que el instrumento. Compare esta suma con el límite nominal para el circuito. Las capacidades nominales de corriente máximas están generalmente impresas en los instrumentos, cerca de los conectores de corriente de entrada.

CAUTION: All devices with DC power supplies are intended for installation in restricted access areas only. A restricted access area is where access can be gained only by service personnel through the use of a special tool, lock and key, or other means of security, and is controlled by the authority responsible for the location.

VORSICHT: Alle Geräte mit DC-Netzteil sind nur für die Installation in Bereichen mit beschränktem Zugang gedacht. Ein Bereich mit beschränktem Zugang ist ein Bereich, zu dem nur Wartungspersonal mit Spezialwerkzeug, Schlüssel oder anderen Sicherheitsvorrichtungen Zugang hat. Dieser Zugang wird von für den Bereich zuständigen Personen überwacht.

MISE EN GARDE: Tous les dispositifs avec bloc d'alimentation C.C. sont conçus pour l'installation dans des zones à accès réglementé uniquement. Une zone à accès réglementé est une zone dont l'accès n'est possible qu'au personnel de service utilisant un verrou, une clé ou un outil spécial, ou d'autres moyens de sécurité, et qui est contrôlée par les autorités responsables du site.

PRECAUCIÓN: Todos los instrumentos con suministros de corriente continua han sido diseñados únicamente para instalación en áreas restringidas. Se entiende como área de acceso restringido un lugar al que solo puede acceder personal de servicio mediante el uso de una herramienta especial, llave y cerrojo u otro medio de seguridad similar, y que esté controlado por la autoridad responsable de esa ubicación.

CAUTION: For the DC input circuit to an FES, FESX, or FWSX (DC power supply part number RPS5DC and RPS-X424-DC), make sure there is a 10-amp listed circuit breaker, minimum -48VDC, double pole, on the input to the terminal block. The input wiring for connection to the product should be Listed copper wire, 14 AWG, marked VW-1, and rated 90 degrees Celsius.

VVORSICHT: Für den Eingangs-Gleichstromkreis an ein FES- oder FESX-Netzteil (Gleichstromnetzteile mit der Teilenr. RPS5DC und RPS-X424-DC) muss gewährleistet warden, dass ein zweipoliger 10 A-Leistungsschalter (min. -48VDC) am Eingang zur Reihenklemme installiert wird. Beim Eingangsdraht für den Anschluss am Produkt muss es sich um einen zulässigen Kupferdraht (14 AWG gekennzeichnet mit VW-1), der für mindestens 90° C ausgelegt ist, handeln.

MISE EN GARDE: Pour le circuit d'alimentation C.C. d'un FES ou FESX (références du bloc d'alimentation C.C. RPS5DC et RPS-X424-DC), assurez-vous de la présence d'un disjoncteur de 10 ampères, minimum –48 V C.C., double coupure, sur l'entrée vers le bloc d'alimentation. Les câbles

d'alimentation pour le produit doivent être en fils de cuivre, 14 AWG (American Wire Gauge), marqués VW-1 et classés 90 degrés Celsius.

PRECAUCIÓN: Para el circuito de entrada de CC a un modelo FES o FESX (suministro de corriente continua con No. de referencia RPS5DC y RPS-X424-DC), verifique que haya un cortacircuitos catalogado para 10 amperios, mínimo de –48 VCC, bipolar, en la entrada al bloque terminal. El cableado de entrada para la conexión al producto deberá ser catalogado de cobre, 14 AWG, marcado VW-1, y nominal para 90 grados Celsius.

CAUTION: Make sure you insert the power supply right-side up. It is possible to insert the supply upside down, although the supply will not engage with the power backplane when upside down. The power supply is right-side up when the power connector is on the left and the fan vent is on the right.

VORSICHT: Sicher Sie sicher, dass Sie das Netzteil mit der richtigen Seite nach oben weisend einstecken. Man kann die Karte auch umgekehrt einstecken. Allerdings rastet das umgekehrte Netzteil nicht in die Netzstrom-Rückwandplatine ein. Die rechte Seite des Netzteils weist nach oben, wenn sich der Stromanschlussstecker links und der Ventilatorschlitz rechts befindet.

MISE EN GARDE: Assurez-vous d'insérer le bloc d'alimentation dans le bon sens. Il est possible de l'insérer " la tête en bas ", mais le bloc d'alimentation ne s'enclenchera pas dans la face arrière d'alimentation s'il est inséré à l'envers. Le bloc d'alimentation est dans le bon sens lorsque le connecteur se trouve sur le côté gauche et le ventilateur sur la droite.

PRECAUCIÓN: Verifique que inserta el suministro de corriente con la cara correcta hacia arriba. Es posible insertar el suministro hacia abajo, pese a que este no se conectará con el enchufe posterior de esta forma. El suministro de potencia estará con la cara correcta hacia arriba cuando el conector de corriente quede a la izquierda y la abertura del ventilador queda a la derecha.

CAUTION: Use the erase startup-config command only for new systems. If you enter this command on a system you have already configured, the command erases the configuration. If you accidentally do erase the configuration on a configured system, enter the write memory command to save the running configuration to the startup-config file.

VORSICHT: Verwenden Sie den Befehl "Erase startup-config" (Löschen Startup-Konfig) nur für neue Systeme. Wenn Sie diesen Befehl in ein bereits konfiguriertes System eingeben, löscht der Befehl die Konfiguration. Falls Sie aus Versehen die Konfiguration eines bereits konfigurierten Systems löschen, geben Sie den Befehl "Write Memory" (Speicher schreiben) ein, um die laufende Konfiguration in der Startup-Konfig-Datei zu speichern.

MISE EN GARDE: N'utilisez la commande erase startup-config que pour les nouveaux systèmes. Si vous entrez cette commande sur un système que vous avez déjà configuré, elle efface la configuration. Si vous effacez la configuration par accident sur un système configuré, entrez la commande write memory pour enregistrer la configuration actuelle dans le fichier startup-config.

PRECAUCIÓN: Use el comando erase startup-config (borrar configuración de inicio) para sistemas nuevos solamente. Si usted introduce este comando en un sistema que ya ha configurado, el comando borrará la configuración. Si usted borra accidentalmente la configuración en un sistema ya configurado, introduzca el comando write memory (escribir memoria) para guardar la configuración en ejecución en el archivo startup-config.

CAUTION: Never leave tools inside the chassis.

VORSICHT: Lassen Sie keine Werkzeuge im Chassis zurück.

MISE EN GARDE: Ne laissez jamais d'outils à l'intérieur du châssis.

PRECAUCIÓN: No deje nunca herramientas en el interior del chasis.

CAUTION:	Do not remove the management module while the SuperX chassis is powered on and running. If you attempt to remove this module while the chassis is powered on and running, all traffic being handled by the system will stop.
VORSICHT:	Wenn das SuperX-Gehäuse eingeschaltet ist und sich im Betrieb befindet, darf das Management-Modul nicht entfernt werden. Wenn dieses Modul bei eingeschaltetem und laufendem Gehäuse entfernt wird, kommt der gesamte, vom System gehandhabte Verkehr zum Stillstand.
MISE EN GARDE:	N'enlevez pas le module de gestion pendant que le châssis SuperX est allumé et en cours de fonctionnement. Si vous essayez d'enlever ce module pendant que le châssis est allumé et en cours de fonctionnement, tout le trafic traité par le système sera interrompu.
PRECAUCIÓN:	No desmonte el módulo de conducción mientras el chasis SuperX esté encendido y en funcionamiento. De hacerlo así, todo el tráfico que esté siendo administrado por el sistema se detendrá.

CAUTION:	If you do not install a module in a slot, you must keep the slot panel in place. If you run the chassis with an uncovered slot, the system will overheat.
VORSICHT:	Falls kein Modul im Steckplatz installiert wird, muss die Steckplatztafel angebracht werden. Wenn ein Steckplatz nicht abgedeckt wird, läuft das System heiß.
MISE EN GARDE:	Si vous n'installez pas de module dans un slot, vous devez laisser le panneau du slot en place. Si vous faites fonctionner le châssis avec un slot découvert, le système surchauffera.
PRECAUCIÓN:	Si no instala un módulo en la ranura, deberá mantener el panel de ranuras en su lugar. Si pone en funcionamiento el chasis con una ranura descubierta, el sistema sufrirá sobrecalentamiento.

CAUTION:	The SX-POE-AC-PWR power supply is designed exclusively for use with the FastIron SuperX POE devices. The power supply produces extensive power to support 802.3af applications. Installing the power supply in a device other than the FastIron SuperX POE will cause extensive damage to your equipment.
VORSICHT:	Das SX-POE-AC-PWR Stromnetz hat für die FastIron SuperX POE Geräte ausschießlich aufgezeichnet. Dieses Stromnetz erzeugt umfassend Starkstrom zur Bestätigun von 802.3af Anwendungen. Ihr Anlage beschädigt wird, wenn das Stromnetz in Geräte anders als FastIron SuperX POE einbauen wird.
MISE EN GARDE:	Le bloc d'alimentation SX-POE-AC-PWR est conçu exclusivement pour être utilisé avec les dispositifs FastIron SuperX POE. Le bloc d'alimentation produit une alimentation très importante pour prendre en charge les applications 802.3af. Si vous l'installez dans un dispositif autre que les FastIron SuperX POE, il endommagera gravement votre équipement.
PRECAUCIÓN	El suministro de corriente alterna del SX-POE-AC-PWR está diseñado exclusivamente para uso con los dispositivos FastIron SuperX POE. El suministro de corriente produce suficiente energía para abastecer a las aplicaciones 802.3af. Si se instala el suministro de corriente en un dispositivo que no sea el FastIron SuperX POE, se producirán daños de consideración al equipo.

Warnings

A warning calls your attention to a possible hazard that can cause injury or death. The following are the warnings used in this manual.

"Achtung" weist auf eine mögliche Gefährdung hin, die zu Verletzungen oder Tod führen können. Sie finden die folgenden Warnhinweise in diesem Handbuch:

Un avertissement attire votre attention sur un risque possible de blessure ou de décès. Ci-dessous, vous trouverez les avertissements utilisés dans ce manuel.

Una advertencia le llama la atención sobre cualquier posible peligro que pueda ocasionar daños personales o la muerte. A continuación se dan las advertencias utilizadas en este manual.

WARNING:	The procedures in this manual are for qualified service personnel.
ACHTUNG:	Die Verfahren in diesem Handbuch sind nur für qualifiziertes Wartungspersonal gedacht.
AVERTISSEMENT:	Les procédures décrites dans ce manuel doivent être effectuées par le personnel de service qualifié uniquement.
ADVERTENCIA:	Los procedimientos de este manual se han hecho para personal de servicio cualificado.

WARNING:	All fiber optic interfaces use Class 1 lasers.
ACHTUNG:	Alle Glasfaser-Schnittstellen verwenden Laser der Klasse 1.
AVERTISSEMENT:	Toutes les interfaces en fibres optiques utilisent des lasers de classe 1.
ADVERTENCIA:	Todas las interfaces de fibra óptica utilizan láser de clase 1.

WARNING:	Make sure the rack or cabinet housing the device is adequately secured to prevent it from becoming unstable or falling over.
ACHTUNG:	Stellen Sie sicher, dass das Gestell oder der Schrank für die Unterbringung des Geräts auf angemessene Weise gesichert ist, so dass das Gestell oder der Schrank nicht wackeln oder umfallen kann.
AVERTISSEMENT:	Vérifiez que le bâti ou le support abritant le dispositif est bien fixé afin qu'il ne devienne pas instable ou qu'il ne risque pas de tomber.
ADVERTENCIA:	Verifique que el bastidor o armario que alberga el instrumento está asegurado correctamente para evitar que pueda hacerse inestable o que caiga.

WARNING:	Mount the devices you install in a rack or cabinet as low as possible. Place the heaviest device at the bottom and progressively place lighter devices above.
ACHTUNG:	Montieren Sie die Geräte im Gestell oder Schrank so tief wie möglich. Platzieren Sie das schwerste Gerät ganz unten, während leichtere Geräte je nach Gewicht (je schwerer desto tiefer) darüber untergebracht werden.
AVERTISSEMENT:	Montez les dispositifs que vous installez dans un bâti ou support aussi bas que possible. Placez le dispositif le plus lourd en bas et le plus léger en haut, en plaçant tous les dispositifs progressivement de bas en haut du plus lourd au plus léger.
ADVERTENCIA:	Monte los instrumentos que instale en un bastidor o armario lo más bajos posible. Ponga el instrumento más pesado en la parte inferior y los instrumentos progresivamente más livianos más arriba.

WARNING: Disconnect the power cord from all power sources to completely remove power from the device.

ACHTUNG: Ziehen Sie das Stromkabel aus allen Stromquellen, um sicherzustellen, dass dem Gerät kein Strom zugeführt wird.

AVERTISSEMENT: Débranchez le cordon d'alimentation de toutes les sources d'alimentation pour couper complètement l'alimentation du dispositif.

ADVERTENCIA: Para desconectar completamente la corriente del instrumento, desconecte el cordón de corriente de todas las fuentes de corriente.

WARNING: Make sure that the power source circuits are properly grounded, then use the power cord supplied with the device to connect it to the power source.

ACHTUNG: Stellen Sie sicher, dass die Stromkreise ordnungsgemäß geerdet sind. Benutzen Sie dann das mit dem Gerät gelieferte Stromkabel, um es an die Srromquelle anzuschließen.

AVERTISSEMENT: Vérifiez que les circuits de sources d'alimentation sont bien mis à la terre, puis utilisez le cordon d'alimentation fourni avec le dispositif pour le connecter à la source d'alimentation.

ADVERTENCIA: Verifique que circuitos de la fuente de corriente están conectados a tierra correctamente; luego use el cordón de potencia suministrado con el instrumento para conectarlo a la fuente de corriente.

WARNING: Do not use the handles on the power supply units to lift or carry a Foundry device.

ACHTUNG: Die Griffe an den Netzteilen dürfen nicht zum Anheben oder Tragen eines Foundrygeräte verwendet werden.

AVERTISSEMENT: N'utilisez pas les poignées des unités de bloc d'alimentation pour soulever ou porter un dispositif en châssis.

ADVERTENCIA: No use las asas de las unidades de suministro de corriente para alzar o transportar un instrumento de Foundry.

WARNING: If the installation requires a different power cord than the one supplied with the device, make sure you use a power cord displaying the mark of the safety agency that defines the regulations for power cords in your country. The mark is your assurance that the power cord can be used safely with the device.

ACHTUNG: Falls für die Installation ein anderes Stromkabel erforderlich ist (wenn das mit dem Gerät gelieferte Kabel nicht passt), müssen Sie sicherstellen, dass Sie ein Stromkabel mit dem Siegel einer Sicherheitsbehörde verwenden, die für die Zertifizierung von Stromkabeln in Ihrem Land zuständig ist. Das Siegel ist Ihre Garantie, dass das Stromkabel sicher mit Ihrem Gerät verwendet werden kann.

AVERTISSEMENT: Si l'installation nécessite un cordon d'alimentation autre que celui fourni avec le dispositif, assurez-vous d'utiliser un cordon d'alimentation portant la marque de l'organisation responsable de la sécurité qui définit les normes et régulations pour les cordons d'alimentation dans votre pays. Cette marque vous assure que vous pouvez utiliser le cordon d'alimentation avec le dispositif en toute sécurité.

ADVERTENCIA: Si la instalación requiere un cordón de corriente distinto al que se ha suministrado con el instrumento, verifique que usa un cordón de corriente que venga con la marca de la agencia de seguridad que defina las regulaciones para cordones de corriente en su país. Esta marca será su garantía de que el cordón de corriente puede ser utilizado con seguridad con el instrumento.

WARNING: Power supplies are hot swappable. However, Foundry Networks recommends that you disconnect the power supply from AC power before installing or removing the supply. The device can be running while a power supply is being installed or removed, but the power supply itself should not be connected to a power source. Otherwise, you could be injured or the power supply or other parts of the device could be damaged.

ACHTUNG: Netzteile können unter Strom stehend ausgetauscht werden. Allerdings empfiehlt Foundry Networks, dass Sie das Netzteil vom Netzstrom abtrennen, bevor Sie das Netzteil anschließen oder abtrennen. Das Gerät kann während des Anschließens oder Abnehmens des Netzteils laufen. Nur das Netzteil sollte nicht an eine Stromquelle angeschlossen sein. Ansonsten können Sie verletzt oder das Netzteil bzw. andere Geräteteile beschädigt werden.

AVERTISSEMENT: Les blocs d'alimentation peuvent être changés à chaud. Cependant, Foundry Networks vous conseille de débrancher le bloc d'alimentation de l'alimentation C.A. avant d'installer ou d'enlever le bloc d'alimentation. Le dispositif peut être en cours de fonctionnement pendant que vous installez ou enlevez un bloc d'alimentation, mais le bloc d'alimentation lui-même ne doit pas être connecté à une source d'alimentation. Sinon, vous risquez d'être blessé ou le bloc d'alimentation ou d'autres pièces du dispositif risquent d'être endommagés.

ADVERTENCIA: Los suministros de corriente pueden intercambiarse sin necesidad de ajustes. No obstante, Foundry Networks recomienda que desconecte el suministro de corriente de la toma de corriente alterna antes de instalar o retirar el suministro. El instrumento puede estar activado cuando se esté instalando o retirando un suministro de corriente, pero el suministro de corriente en sí no deberá estar conectado a la fuente de corriente. De no hacerlo así, podría sufrir daños personales o el suministro de corriente u otras piezas podrían resultar dañadas.

WARNING: Before beginning the installation, see the precautions in "Power Precautions" on page 2-4.

ACHTUNG: Vor der Installation siehe Vorsichtsmaßnahmen unter " Power Precautions " (Vorsichtsmaßnahmen in Bezug auf elektrische Ablagen) auf den Seiten 2 - 4.

AVERTISSEMENT: Avant de commencer l'installation, consultez les précautions décrites dans " Power Precautions " (Précautions quant à l'alimentation), pages 2-4.

ADVERTENCIA: Antes de comenzar la instalación, consulte las precauciones en la sección " Power Precautions" (Precauciones sobre corriente) que se encuentra en las páginas 2-4.

WARNING: For safety reasons, the ESD wrist strap should contain a series 1 meg ohm resistor.

ACHTUNG: Aus Sicherheitsgründen sollte ein EGB-Armband zum Schutz von elektronischen gefährdeten Bauelementen mit einem 1 Megaohm-Reihenwiderstand ausgestattet sein.

AVERTISSEMENT: Pour des raisons de sécurité, la dragonne ESD doit contenir une résistance de série 1 méga ohm.

ADVERTENCIA: Por razones de seguridad, la correa de muñeca ESD deberá contener un resistor en serie de 1 mega ohmio.

WARNING: A fully populated chassis is heavy. TWO OR MORE PEOPLE ARE REQUIRED WHEN LIFTING, HANDLING, OR MOUNTING THESE DEVICES.

ACHTUNG: Ein voll bestücktes Gehäuse ist schwer. ZUM ANHEBEN, HANDHABEN ODER MONTIEREN DIESER GERÄTE SIND MINDESTENS ZWEI PERSONEN ERFORDERLICH.

AVERTISSEMENT: Les châssis sont lourds quand ils sont entièrement remplis. POUR SOULEVER, MANIPULER OU MONTER CES DISPOSITIFS, DEUX PERSONNES MINIMUM SONT NÉCESSAIRES.

ADVERTENCIA: Un chasis muy concurrido es muy pesado. SE REQUIEREN DOS O MÁS PERSONAS CUANDO SE VAYA A ALZAR, MANEJAR O MONTAR ESTE DISPOSITIVO.

WARNING: Make sure to choose the appropriate circuit device depending on the number of AC power supplies installed in the chassis. The maximum current draw for the system is one AC power supply.

ACHTUNG: Je nach Anzahl der Wechselstrom-Netzteile im Gehäuse muss das passende Stromgerät ausgewählt werden. Es darf nicht mehr als der von einem Wechselstrom-Netzteil bereitgestellte Strom entnommen werden.

AVERTISSEMENT: Assurez-vous de choisir le dispositif de circuit approprié selon le nombre de blocs d'alimentation C.A. installés dans le châssis. L'appel de courant maximum pour le système est d'un bloc d'alimentation C.A.

ADVERTENCIA: Asegúrese de elegir el dispositivo de circuito apropiado dependiendo del número de suministros de CA instalados en el chasis. La llamada de corriente máxima para el sistema es de un suministro de CA.

WARNING: Be careful not to accidently insert your fingers into the fan tray while removing it from the chassis. The fan may still be spinning at a high speed.

ACHTUNG: Die Finger dürfen nicht versehentlich in das Ventilatorblech gesteckt werden, wenn dieses vom Gehäuse abgenommen wird. Der Ventilator kann sich unter Umständen noch mit hoher Geschwindigkeit drehen.

AVERTISSEMENT: Faites attention de ne pas accidentellement insérer vos doigts dans le boîtier du ventilateur lorsque vous l'enlevez du châssis. Il est possible que le ventilateur tourne encore à grande vitesse.

ADVERTENCIA: Procure no insertar los dedos accidentalmente en la bandeja del ventilador cuando esté desmontando el chasis. El ventilador podría estar girando a gran velocidad.