Foundry Switch and Router Command Line Interface Reference

FOUNDRY NETWORKS
www.foundrynetworks.com

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

May 2003

Contents

Chapter 1
Getting Started

Introduction

This reference describes the Command Line Interface (CLI) for Foundry Layer 2 Switches and Layer 3 Switches.

For step-by-step instructions on how to install key features of the system, see the *Foundry Switch and Router Installation and Basic Configuration Guide*.

NOTE: Not all commands shown in this reference are supported on both Layer 2 Switch and Layer 3 Switch platforms. Stackable devices and Chassis devices may also have differing commands.

NOTE: This reference lists all the commands that appear at each command level for users with super-user access. If you are logged on with port-configuration access or read-only access, some of these commands will not be displayed and will not be available.

NOTE: This reference does not describe the Layer 2 Switch and Layer 3 Switch diagnostic commands or the ServerIron Transparent Cache Switching (TCS), Server Load Balancing (SLB), or Firewall Load Balancing (FWLB) commands.

For information about the ServerIron TCS, SLB, and FWLB commands, see the *Foundry ServerIron Installation and Configuration Guide*.

For information about the diagnostics commands, see the *Foundry Diagnostic Guide*.

Audience

This manual is designed for system administrators with a working knowledge of Layer 2 and Layer 3 switching and routing.

If you are using a Foundry Layer 3 Switch, you should be familiar with the following protocols if applicable to your network – IP, RIP, OSPF, IS-IS, BGP4, MBGP, MPLS, IGMP, PIM, DVMRP, IPX, AppleTalk, FSRP, 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.

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 Switch and Router Installation and Basic Configuration Guide* – provides hardware and software installation information, and configuration information for system-level features.
- *Foundry Security Guide* – provides procedures for securing management access to Foundry devices and for protecting against Denial of Service (DoS) attacks.
- *Foundry Enterprise Configuration and Management Guide* – provides configuration information for enterprise routing protocols including IP, RIP, IP multicast, OSPF, BGP4, VRRP and VRRPE.
- *Foundry NetIron Service Provider Configuration and Management Guide* – provides configuration information for IS-IS and MPLS.
- *Foundry Diagnostic Guide* – provides descriptions of diagnostic commands that can help you diagnose and solve issues on Layer 2 Switches and Layer 3 Switches.

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.

What's New In This Edition?

This edition describes the following software release:

- 07.6.03
- 07.6.02

This release applies to the following products:

- NetIron Internet Backbone router
- BigIron with M2 (Management II) or higher modules
- BigIron with Velocity Management Module version I (VM1)
- FastIron II, FastIron II Plus, and FastIron III with M2 or higher modules
- FastIron 4802
- FastIron Edge Switch Series

For a list of the enhancements, see the "Getting Started" chapter in the *Foundry Switch and Router Installation and Basic Configuration Guide*.

NOTE: If you want documentation specifically for a 07.1.x release, see the January, 2001 edition of the manuals and the release notes for the release you are using. For the 07.2.06 release or a 07.3.x release, see the June, 2001 edition and the release notes.

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

The latest product information and technical tips are always available to our customers from the Foundry Networks web site. You can access the web site at the following URL:

- 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
Using the Command Line Interface

The CLI is a text-based interface for configuring and monitoring Foundry Layer 2 Switches and Layer 3 Switches. You can access the CLI can through either a direct serial connection to the device or through a Telnet session.

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 trace routes.
- 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 Access Control Lists (ACLs), a RADIUS server, or a TACACS/TACACS+ server for authentication. See the *Foundry Security Guide*.

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 the point in the command string.

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.

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 2.1: 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.

Table 2.1: CLI Line-Editing Commands (Continued)

Ctrl-Key Combination	Description
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.

EXEC Commands

There are two different levels of EXEC commands, the ***User Level*** and the ***Privileged Level***. The User level commands are at the top of the CLI hierarchy. These are the first commands that you have access to when connected to the device through the CLI. At this level, you can view basic system information and verify connectivity but cannot make any changes to the device configuration. To make changes to the configuration, you must move to other levels of the CLI hierarchy. This is accomplished by the User EXEC level command **enable** at initial log-on. This command takes you to the Privileged EXEC level, from which you can reach the configuration command levels.

Privileged Level

The Privileged EXEC level commands primarily enable you to transfer and store software images and configuration files between the network and the system, and review the configuration.

You reach this level by entering **enable** [<password>] or **enable** <username> <password> at the User EXEC level.

CONFIG Commands

CONFIG commands modify the configuration of a Foundry Layer 2 Switch or Layer 3 Switch. This reference describes the following CONFIG CLI levels.

Global Level

The global CONFIG level allows you to globally apply or modify parameters for ports on the device. You reach this level by entering **configure terminal** at the privileged EXEC level.

Redundancy Level (Layer 3 Switches only)

This redundancy level allows you to configure redundancy parameters for redundant management modules. You reach this level by entering the **redundancy** command at the global CONFIG level.

Interface Level

The interface level allows you to assign or modify specific port parameters on a port-by-port basis. You reach this level by entering **interface ethernet** <portnum>, **interface loopback** <num>, **interface ve** <num>, or **interface pos** <portnum> at the global CONFIG level.

Trunk Level

The trunk level allows you to change parameters for statically-configured trunk groups. You reach this level by entering a **trunk** command with the appropriate port parameters.

Router RIP Level (Layer 3 Switches only)

The RIP level allows you to configure parameters for the RIP routing protocol. You reach this level by entering the **router rip** command at the global CONFIG level.

Router OSPF Level (Layer 3 Switches only)

The OSPF level allows you to configure parameters for the OSPF routing protocol. You reach this level by entering the **router ospf** command at the global CONFIG level.

Router IS-IS Level (Layer 3 Switches only)

The IS-IS level allows you to configure parameters for the IS-IS routing protocol. You reach this level by entering the **router isis** command at the global CONFIG level.

BGP Level (Layer 3 Switches only)

The BGP level allows you to configure Layer 3 Switches for Border Gateway Protocol version 4 (BGP4). You reach this level by entering the **router bgp** command at the global CONFIG level.

IP Tunnel Level (Layer 3 Switches only)

The IP tunnel level allows you to define parameters for IP-in-IP tunnels to pass data through non-DVMRP and non-PIM IP multicast routers.

You reach this level by entering the **ip tunnel**… command at the interface CONFIG level.

Router MSDP Level (Layer 3 Switches only)

The MSDP level allows you to configure details for the Multicast Source Discovery Protocol (MSDP). You reach this level by entering the **router msdp** command at the global CONFIG level.

Router DVMRP Level (Layer 3 Switches only)

The DVMRP level allows you to configure details for the DVMRP multicast protocol. You reach this level by entering the **router dvmrp** command at the global CONFIG level.

Router PIM Level (Layer 3 Switches only)

The PIM level allows you to configure parameters for the Protocol Independent Multicast (PIM) routing protocol. You reach this level by entering the **router pim** command at the global CONFIG level.

Broadcast Filter Level

The broadcast filter level allows you to assign broadcast filters to specific ports. You reach this level by entering **broadcast filter**... at the global CONFIG level.

Multicast Filter Level

The multicast filter level allows you to assign multicast filters to specific ports. You reach this level by entering **multicast filter**... at the global CONFIG level.

Route Map Level (Layer 3 Switches only)

The Route Map level allows you to configure parameters for a BGP4 route map. You reach this level by entering the **route-map** <name> command at the global CONFIG level.

Router VRRP Level (Layer 3 Switches only)

The VRRP level allows you to configure parameters for the Virtual Router Redundancy Protocol (VRRP). You reach this level by entering the **router vrrp** command at the global CONFIG level, then entering the **ip vrrp vrid** <num> command at the interface configuration level.

Router VRRPE Level (Layer 3 Switches only)

The VRRPE level allows you to configure parameters for VRRP Extended. You reach this level by entering the **router vrrp-extended** command at the global CONFIG level, then entering the **ip vrrp-extended vrid** <num> command at the interface configuration level.

MPLS Level

The MPLS level allows you to configure Multiprotocol Label Switching on Foundry NetIron Internet Routers. You reach this level by entering the **router mpls** command at the Global CONFIG level.

MPLS Interface Level

The MPLS interface level allows you to configure parameters for an individual MPLS-enabled interface. You reach this level by entering the **mpls-interface pos** <interface number> command at the MPLS level.

MPLS Policy Level

The MPLS policy level allows you to configure global MPLS parameters. You reach this level by entering the **policy** command at the MPLS level.

MPLS Path Level

The MPLS path level allows you to configure paths that specify a route across an MPLS domain. You reach this level by entering the **path** <path name> command at the MPLS level.

MPLS Signalled LSP Level

The MPLS signalled LSP level allows you to configure dynamically established LSPs. You reach this level by entering the **lsp** <name> command at the MPLS level.

MPLS Secondary Path Level

The MPLS secondary path level allows you to configure redundant paths for an LSP. You reach this level by entering the **secondary-path** <path name> command at the MPLS signalled LSP level.

MPLS RSVP Level

The MPLS RSVP level allows you to configure RSVP parameters for use in signalling LSPs. You reach this level by entering the **rsvp** command at the MPLS level.

MPLS Static LSP Level

The MPLS static LSP level allows you to configure static LSPs. You reach this level by entering the **static-lsp** <name> command at the MPLS level.

MPLS VLL Level

The MPLS VLL level allows you to configure MPLS Virtual Leased Line. You reach this level by entering the **vll** <name> command at the MPLS level.

MPLS LDP Level

The MPLS LDP level allows you to configure Label Distribution Protocol (LDP). You reach this level by entering the **ldp** command at the MPLS level.

MPLS VPLS Level

The MPLS VPLS level allows you to configure MPLS Virtual Private LAN Services. You reach this level by entering the **vpls** <name> command at the MPLS level.

VLAN Level

Policy-based VLANs allow you to assign VLANs on a protocol (IP, IPX, Decnet, AppleTalk, NetBIOS, Others), sub-net (IP sub-net and IPX network), AppleTalk cable, port, or 802.1q tagged basis.

You reach this level by entering the **vlan** <vlan-id> **by port** command at the Global CONFIG Level.

Metro Ring Level

Metro rings provide Layer 2 connectivity and fast failover in ring topologies.

You reach this level by entering the **metro-ring** <ring-id> command at the Global CONFIG Level.

VSRP Level

The VSRP level allows you to configure parameters for the Virtual Switch Redundancy Protocol (VSRP). You reach this level by entering the **vsrp vrid** <num> command at the VLAN configuration level, then entering the **vsrp vrid** <num> command at the VLAN configuration level.

Topology Group Level

A topology group enables you to control the Layer 2 protocol configuration and Layer 2 state of a set of ports in multiple VLANs based on the configuration and states of those ports in a single master VLAN. One instance of the Layer 2 protocol controls all the VLANs.

You reach this level by entering the **topology-group** <group-id> command at the Global CONFIG Level.

STP Group Level

STP groups enable you to manage multiple port-based VLANs using the same spanning tree.

You reach this level by entering the **stp-group** <num> command at the Global CONFIG Level.

802.1X Port Security Level

The 802.1X port security level allows you to configure the 802.1X port security. You reach this level by entering the **dot1x-enable** command at the at the Global level.

MAC Port Security Level

The MAC port security level allows you to configure the port security feature. You reach this level by entering the **port security** command at the at the Global or Interface levels.

GVRP Level

GARP VLAN Registration Protocol (GVRP) is a Generic Attribute Registration Protocol (GARP) application that provides VLAN registration service by means of dynamic configuration (registration) and distribution of VLAN membership information.

You reach the GVRP level by entering the **gvrp-enable** command at the Global CONFIG Level.

Real Server Level (Layer 3 Switches only)

The Real Server level allows you to configure router-based health check parameters for a Layer 3 Switch to assist with Globally-distributed Server Load Balancing (SLB). See the "Route Health Injection" chapter of the *Foundry Enterprise Configuration and Management Guide*. You reach this level by entering the **server real**... command at the global CONFIG level.

Application Port Level (Layer 3 Switches only)

The Application Port level allows you to configure health check parameters for a TCP HTTP port. The commands at this level apply only when you are configuring a Layer 3 Switch to assist ServerIrons, third-party SLBs, or web servers with globally-distributed SLB. See the "Route Health Injection" chapter of the *Foundry Enterprise Configuration and Management Guide*. You reach this level by entering the **server port http** | <tcp/udp-portnum> command at the global CONFIG level.

NOTE: If you enter **server port ?**, numerous well-known port names are listed. The current software release supports only HTTP ports.

Accessing the CLI

The CLI can be accessed through both serial and Telnet connections. For initial log on, you must use a serial connection. Once an IP address is assigned, you can access the CLI through Telnet.

NOTE: When accessing the CLI through Telnet, you are prompted for a password. By default, the password required is the password you enter for general access at initial setup. You also have the option of assigning a separate password for Telnet access with the **enable telnet password** <password> command, found at the Global Level.

NOTE: At initial log on, all you need to do is type **enable** at the prompt, then press Return. You only need to enter a password after a permanent password is entered at the Global CONFIG Level of the CLI.

Once connectivity to the device is established, you will see one of the following prompts:

```
FI4802-PREM>
BigIron>
FastIronII>
ServerIron>
FastIron>
NetIron>
TurboIron>
```

At this prompt (>), you are at the user EXEC level of the CLI command structure.

NOTE: If you install switch code on a router, the command prompt begins with "`SW-`" to indicate the software change. This is true even if you change the system name.

To reach the Global CONFIG Level, the uppermost level of the CONFIG commands, enter the following commands:

```
BigIron> enable                 User Level-EXEC commands
BigIron# configure terminal     Privileged Level-EXEC commands
BigIron(config)#                Global Level-CONFIG commands
```

You can then reach all other levels of the CONFIG command structure from this point.

NOTE: Layer 2 Switch configuration is done using the following CLI levels: global, interface, VLAN port and protocol.

NOTE: Layer 3 Switch configuration is done using the following CLI levels: global, interface, IP tunnel, VLAN, and router levels.

NOTE: The CLI prompt will change at each level of the CONFIG command structure, to easily identify the current level:

`BigIron>`	User Level EXEC Command
`BigIron#`	Privileged Level EXEC Command
`BigIron(config)#`	Global Level CONFIG Command
`BigIron(config-if-5/1)#`	Interface Level CONFIG Command
`BigIron(config-lbif-1)#`	Loopback Interface CONFIG Command
`BigIron(config-ve-1)#`	Virtual Interface CONFIG Command
`BigIron(config-trunk-4/1-4/8)#`	Trunk group CONFIG Command
`BigIron(config-if-tunnel)#`	IP Tunnel Level CONFIG Command
`BigIron(config-bcast-filter-id-1)#`	Broadcast Filter Level CONFIG Command
`BigIron(config-mcast-filter-id-1)#`	Multicast Filter Level CONFIG Command
`BigIron(config-bgp-router)#`	BGP Level CONFIG Command
`BigIron(config-dvmrp-router)#`	DVMRP Level CONFIG Command
`BigIron(config-ospf-router)#`	OSPF Level CONFIG Command
`BigIron(config-isis-router)#`	`IS-IS` Level CONFIG Command
`BigIron(config-pim-router)#`	PIM Level CONFIG Command
`BigIron(config-msdp-router)#`	MSDP Level CONFIG Command
`BigIron(config-redundancy)#`	Redundant Management Module CONFIG Command
`BigIron(config-rip-router)#`	RIP Level CONFIG Command
`BigIron(config-rs-realservername)#`	Real Server Level CONFIG Command
`BigIron(config-port-80)#`	Application Port CONFIG Command
`BigIron(config-bgp-routemap Map_Name)#`	Route Map Level CONFIG Command
`BigIron(config-vlan-1)#`	VLAN Port-based Level CONFIG Command
`BigIron(config-vlan-atalk-proto)#`	VLAN Protocol Level CONFIG Command
`BigIron(config-stp-group-1)#`	STP Group CONFIG Command
`BigIron(config-gvrp)#`	GVRP CONFIG Command
`NetIron(config-mpls)#`	MPLS Level CONFIG Command
`NetIron(config-mpls-interface)#`	MPLS Interface Level CONFIG Command
`NetIron(config-mpls-policy)#`	MPLS Policy Level CONFIG Command
`NetIron(config-mpls-path)#`	MPLS Path Level CONFIG Command
`NetIron(config-mpls-lsp)#`	MPLS Signalled LSP Level CONFIG Command
`NetIron(config-mpls-lsp-sec-path)#`	MPLS Secondary Path Level CONFIG Command
`NetIron(config-mpls-rsvp)#`	MPLS RSVP Level CONFIG Command
`NetIron(config-mpls-static-lsp)#`	MPLS Static LSP Level CONFIG Command

NOTE: The CLI prompt at the interface level includes the port speed. The speed is one of the following:

- e100 – The interface is a 10/100 port.
- e1000 – The interface is a Gigabit port.

For simplicity, the port speeds sometimes are not shown in example Interface level prompts in this manual.

Navigating Among Command Levels

To reach other CLI command levels, you need to enter certain commands. At each level there is a launch command that allows you to move either up or down to the next level.

CLI Command Structure

Many CLI commands may require textual or numeral input as part of the command. These fields are either required or optional depending on how the information is bracketed. For clarity, a few CLI command examples are explained below.

EXAMPLE:

Syntax: deny redistribute <value> all | bgp | rip | static address <ip-addr> <ip-mask>
[match-metric <value> | set-metric <value>]

When an item is bracketed with "< >" symbols, the information requested is a variable and required.

When an item is not enclosed by "< >" or "[ ]" symbols, the item is a required keyword.

When an item is bracketed with "[ ]" symbols, the information requested is optional.

When two or more options are separated by a vertical bar, " | ", you must enter one of the options as part of the command.

EXAMPLE:

Syntax: priority normal | high *means enter either* **priority normal** *or* **priority high**

For example, the command syntax above requires that either **normal** or **high** be entered as part of the command.

To get a quick display of available options at a CLI level or for the next option in a command string, enter a question mark (?) at the prompt or press TAB.

EXAMPLE:

To view all available commands at the user EXEC level, enter the following or press TAB at the User EXEC CLI level:

```
BigIron> ? <return>
enable
exit
fastboot
ping
show
stop-trace-route
traceroute
```

You also can use the question mark (?) with an individual command, to see all available options or to check context.

EXAMPLE:

To view possible **copy** command options, enter the following:

```
BigIron# copy ?
  flash
  running-config
  startup-config
  tftp
BigIron# copy flash ?
  tftp
```

Searching and Filtering Output

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 2-11 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.

```
BigIron# 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.

```
BigIron# 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.

```
BigIron# 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 is displayed 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. You can also search and filter output from this prompt. For example:

```
BigIron# ?
  append              Append one file to another
  appletalk-ping      Ping AppleTalk node
  atm                 ATM commands
  attrib              Change flash card file attribute
  boot                Boot system from bootp/tftp server/flash image
  cd                  Change flash card working slot or current directory
  chdir               Change flash card working slot or current directory
  clear               Clear table/statistics/keys
  clock               Set clock
  configure           Enter configuration mode
  copy                Copy between flash, flash card, tftp, config/code
  debug               Enable debugging functions (see also 'undebug')
  delete              Delete flash card files
  dir                 List flash card files
  disable             Disable a module before removing it
  enable              Enable a disabled module
  erase               Erase image/configuration from flash
  exit                Exit Privileged mode
  fastboot            Select fast-reload option
  format              Format flash card
  gignpa              Gigabit processor commands
  hd                  Display hex dump of flash card file
  kill                Kill active CLI session
--More--, next page: Space, next line: Return key, quit: Control-c
```

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...
  sync-standby        Synchronize active and standby module
  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 2.2: 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*

Table 2.2: Special Characters for Regular Expressions (Continued)

Character	Operation
+	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+
?	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_

Table 2.2: Special Characters for Regular Expressions (Continued)

Character	Operation
[ ]	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.
|	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 “*****”.

```
BigIron# show ip route bgp | include \*
```

Syntax Shortcuts

A command or parameter can be abbreviated as long as enough text is entered to distinguish it from other commands at that level. For example, given the possible commands **copy tftp**… and **config tftp**…, possible shortcuts are **cop tftp** and **con tftp** respectively. In this case, **co** does not properly distinguish the two commands.

Saving Configuration Changes

You can make configuration changes while the device is running. The type of configuration change determines whether or not it becomes effective immediately or requires a save to flash (**write memory**) and reset of the system (**reload**), before it becomes active.

This approach in adopting configuration changes:

- Allows you to make configuration changes to the operating or running configuration of the device to address a short-term requirement or validate a configuration without overwriting the permanent configuration file, the startup configuration, that is saved in the system flash, and;
- Ensures that dependent or related configuration changes are all cut in at the same time.

In all cases, if you want to make the changes permanent, you need to save the changes to flash using the **write memory** command. When you save the configuration changes to flash, this will become the configuration that is initiated and run at system boot.

NOTE: Most configuration changes are dynamic and thus do not require a software reload. If a command requires a software reload to take effect, the documentation states this.

Chapter 3
Command List

This chapter lists all the commands in the CLI. The commands are listed in two ways:

- All commands are listed together in a single alphabetic list. See “Complete Command List” on page 3-1.
- Commands are listed separately for each CLI level (for example, global CONFIG level, BGP4 level, and so on). See “Commands Listed by CLI Level” on page 3-35.

In each list, the page numbers in this reference that describe the commands are listed.

NOTE: This reference does not describe the Layer 2 Switch or Layer 3 Switch diagnostic commands or the ServerIron Transparent Cache Switching (TCS), Server Load Balancing (SLB), or Firewall Load Balancing (FWLB) commands.

For information about the ServerIron TCS, SLB, and FWLB commands, see the *Foundry ServerIron Installation and Configuration Guide*.

For information about the diagnostics commands, see the *Foundry Diagnostic Guide*.

Complete Command List

The following table lists all the CLI commands on Foundry Layer 2 Switches and Layer 3 Switches.

Table 3.1: Complete Layer 2/3 Command List

Table 3.1: Complete Layer 2/3 Command List (Continued)

acl-denied-icmp-msg	6-10
activate	20-1, 21-1, 35-1
active-management	8-1
address-filter	13-1
add-vlan	33-1
admin-group	24-1, 23-1
advertise backup	20-1, 21-1, 35-1
age	40-1
aggregate-address	13-2
aggregated-vlan	6-11
all-client	6-11
always-compare-med	13-2
append	5-1
appletalk address	7-1
appletalk arp-age	6-11
appletalk arp-retransmit-count	6-12
appletalk arp-retransmit-interval	6-12
appletalk cable-range	7-1
appletalk deny	7-1
appletalk deny additional-zones	7-2
appletalk glean-packets	6-12
appletalk permit	7-2
appletalk qos socket	6-12
appletalk routing	7-2
appletalk rtmp-update-interval	6-13
appletalk zip-query-interval	6-13
appletalk zone-name	7-3
appletalk-cable-vlan	33-1
appletalk-ping	4-1
aps authenticate	7-3
aps force	7-3
aps group	7-4
aps lockout	7-4

Table 3.1: Complete Layer 2/3 Command List (Continued)

aps manual	7-5
aps protect	7-5
aps revert	7-5
aps timers	7-6
aps working	7-6
area	11-1
area <num> | <ip-addr> virtual-link <ip-addr>	11-2
area range	11-4
area-password	12-1
arp	6-13
as-path-filter	13-3
as-path-ignore	13-3
atalk-proto	33-2
atm boot interactive	6-14
atm boot tftp	5-2
atm copy slot1 | slot2 flash	5-2
atm copy tftp flash	5-3
atm-hcsadd	7-6
atm pvc	7-6
atm-scram	7-8
atm vc-per-vp	7-9
attrib	5-3
auto-acl-rebind	6-14
auto-cost reference-bandwidth	11-4
auto-gig	7-10
auto-summary	13-3
autosave	40-1
backup	20-1, 21-1, 35-1
backup-hello-interval	20-1, 21-1, 35-2
bandwidth	7-10
banner exec	6-14
banner incoming	6-15
banner motd	6-15

Table 3.1: Complete Layer 2/3 Command List (Continued)

bgp-redistribute-internal	13-4
block-applicant	38-1
block-learning	38-1
boot system bootp	5-4
boot system flash primary	5-4
boot system flash secondary	5-4
boot system slot1 | slot2	5-5, 6-16
boot system tftp	5-5
bootp-relay-max-hops	6-17
broadcast filter	6-17
broadcast limit	6-18, 7-10
bsr-candidate	17-1
cam-partition	6-19
cd	5-5
cdp enable	7-11
cdp run	6-19
chassis name	6-19
chassis poll-time	6-20
chassis trap-log	6-20
chdir	5-5
clear acl-on-arp	5-6
clear appletalk arp	5-6
clear appletalk cache	5-6
clear appletalk route	5-7
clear appletalk traffic	5-7
clear arp	5-7
clear atm map	5-8
clear dot1x statistics	5-8
clear dvmrp cache	5-8
clear dvmrp flow	5-8
clear dvmrp route	5-8
clear fdp counters	5-9
clear fdp table	5-9

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

clock summer-time	6-20
clock timezone	6-20
cluster-id	13-4
community-filter	13-5
compare-routerid	13-5
confederation	13-6
configure terminal	5-19
config-trunk-ind	9-1
confirm-port-up	6-21
console	6-22
copy <from-card> <to-card>	5-19
copy flash flash	5-19
copy flash slot1 | slot2	5-19
copy flash tftp	5-20
copy running slot1 | slot2	5-20
copy running-config tftp	5-21
copy slot1 | slot2 flash	5-21
copy slot1 | slot2 running	5-21
copy slot1 | slot2 start	5-22
copy slot1 | slot2 tftp	5-22
copy start slot1 | slot2	5-22
copy startup-config tftp	5-23
copy tftp flash	5-23
copy tftp running-config	5-23
copy tftp slot1 | slot2	5-23
copy tftp startup-config	5-24
cos	26-1, 29-1
crc	7-11
crypto key	6-22
crypto random-number-seed	6-23
csnp-interval	12-1
cspf	26-1
dampening	13-6

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

frame-relay	7-15
from	26-2
gig-default	6-30, 7-16
global-protocol-vlan	6-30
global-stp	6-31
graft-retransmit-timer	16-2, 17-2
group-router-interface	33-4
gvrp-base-vlan-id	6-31
gvrp-enable	6-31
gvrp-max-leaveall-timer	6-31
hd	5-29
hello padding	7-16, 12-4
hello-interval	12-3, 20-2, 21-2, 31-1, 35-3
hello-interval target	31-2
hello-multiplier	12-4
hello-time	34-2
hello-timer	17-2
hello-timeout	31-2
hello-timeout target	31-2
hold-down-interval	35-4
hop-limit	26-2
hostname	6-32, 12-5
hw-drop-acl-denied-packet	6-32
hw-module	6-32
inactivity-timer	17-2
include-all	26-3
include-any	26-3
include-port	35-4
initial-ttl	35-4
in-segment	29-2
insert	25-2
install	26-3, 29-2
interface	6-33

Table 3.1: Complete Layer 2/3 Command List (Continued)

interface group-ve	6-34
interface link-hold-down	6-35
ip access-group	7-17
ip access-group flow-mode	7-18
ip access-group frag deny	7-18
Note: IronCore only	
ip access-group frag deny	7-19
Note: JetCore only	
ip access-group frag inspect	7-19
ip access-group max-l4-cam	7-19
ip access-group redirect-deny-to-interf	7-20
ip access-group ve-traffic	7-20
ip access-list	6-35
ip access-list disable-log-to-cpu	6-37
ip access-list frag-rate-on-interface	6-37
ip access-list frag-rate-on-system	6-38
ip access-list logging-age	6-38
ip access-policy	6-39
ip access-policy-group	7-21
ip address	6-41, 7-21, 21-2, 35-5
ip-address	20-2, 21-2, 35-5
ip arp-age	6-41, 7-22
ip as-path	6-42
ip bootp-gateway	7-23
ip broadcast-zero	6-42
ip community-list	6-43
ip default-gateway	6-43
ip default-network	6-44
ip directed-broadcast	6-44, 7-23
ip dns domain-name	6-44
ip dns server-address	6-45
ip dont-advertise	7-23
ip dont-use-acl	6-45

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

ip load-sharing	6-55
ip load-sharing by-host	6-55
ip load-sharing route-by-host	6-56
ip metric	7-30
ip mroute	6-57
ip mtu	7-31
ip multicast	6-57
ip multicast age-interval	6-58
ip multicast boundary	7-31
ip multicast filter	6-58
ip multicast query-interval	6-59
ip multicast-perf	6-59
ip multicast-routing	6-59
ip nat inside	7-32
ip nat inside destination list	6-60
ip nat inside destination static	6-60
ip nat inside source list	6-61
ip nat inside source static	6-62
ip nat outside	7-32
ip nat pool	6-62
ip nat translation	6-63
ip net-aggregate	6-63
ip ospf area	7-32
ip ospf auth-change-wait-time	7-33
ip ospf authentication-key	7-33
ip ospf cost	7-34
ip ospf database-filter	7-34
ip ospf dead-interval	7-35
ip ospf hello-interval	7-35
ip ospf md5-authentication	7-35
ip ospf network	7-36
ip ospf passive	7-37
ip ospf priority	7-38

Table 3.1: Complete Layer 2/3 Command List (Continued)

ip ospf retransmit-interval	7-38
ip ospf transmit-delay	7-38
ip pim	7-38
ip pim ttl	7-40
ip pimsm-snooping	6-64
ip pim-sparse	7-39
ip policy frag-match-dst	7-41
ip policy frag-match-src	7-41
ip policy frag-match-src-dest	7-41
ip policy prefer-direct-route	6-65
ip policy route-map	6-65, 7-42
ip prefix-list	6-65
ip proxy-arp	6-66
ip radius source-interface	6-66
ip rarp	6-66
ip rebind-acl	6-67
ip redirect	7-42
ip rip	7-42
ip rip filter-group	7-43
ip rip learn-default	7-43
ip rip poison-reverse	7-43
ip route	6-67
ip route-cache flow	7-43
ip router isis	7-44
ip router-id	6-68
ip show-portname	6-69
ip show-subnet-length	6-69
ip source-route	6-69
ip ssh authentication-retries	6-69
ip ssh idle-time	6-70
ip ssh key-size	6-70
ip ssh password-authentication	6-70
ip ssh permit-empty-passwd	6-71

Table 3.1: Complete Layer 2/3 Command List (Continued)

ip ssh port	6-71
ip ssh pub-key-file	6-71
ip ssh rsa-authentication	6-72
ip ssh scp	6-72
ip ssh source-interface	6-72
ip ssh timeout	6-72
ip strict-acl-tcp	6-73
ip strict-acl-udp	6-73
ip tacacs source-interface	6-74
ip tcp burst-normal	6-74, 7-44
ip tcp keepalive	6-75
ip telnet source-interface	6-76
ip tftp source-interface	6-76
ip ttl	6-77
ip tunnel	7-45
ip use-acl-on-arp	7-45
ip vrrp	7-46
ip vrrp auth-type	7-46
ip vrrp-extended	7-46
ip vrrp-extended auth-type	7-46
ip vsrp auth-type	7-46
ipg10	7-47
ipg100	7-47
ipg1000	7-48
ip-multicast-disable	7-31
ip-policy	7-40
ip-proto	33-4
ip-subnet	33-5
ipv6-proto	33-5
ipx forward-filter	6-77
ipx forward-filter-group	7-48
ipx gns-reply-disable	7-49
ipx gns-round-robin	6-77

Table 3.1: Complete Layer 2/3 Command List (Continued)

ipx max-rip-entries	6-77
ipx max-sap-entries	6-78
ipx netbios-allow	6-78, 7-49
ipx network	7-49
ipx output-gns-filter	7-49
ipx rip-filter	6-78
ipx rip-filter-group	6-78, 7-50
ipx rip-max-packetsize	7-50
ipx rip-multiplier	7-50
ipx sap-access-list	6-79
ipx sap-filter	6-79
ipx sap-filter-group	6-79, 7-51
ipx sap-interval	7-51
ipx sap-max-packetsize	7-51
ipx sap-multiplier	7-51
ipx update-time	7-52
ipx-network	33-6
ipx-proto	33-6
isis circuit-type	7-52
isis metric	7-52
isis passive	7-53
isis password	7-53
isis priority	7-53
is-type	12-5
join-timer	38-3
jumbo	6-80
jumbo1920	6-81
keepalive	7-54
kill	5-30
l2-session-switch	7-56
ldp	22-1
ldp-enable	22-2
learn-default	10-3

Table 3.1: Complete Layer 2/3 Command List (Continued)

link-aggregate active | passive | off	7-55
link-aggregate configure	7-56
link-keepalive ethernet <portnum>	6-81
link-keepalive interval	6-81
link-keepalive retries	6-82
load-interval	7-56
local-as	13-9
locate	5-31
lock-address ethernet	6-82
log	11-10
log-adjacency-changes	12-6
logging	6-82
loop	7-57
loose	25-2
lsp	22-2
lsp-gen-interval	12-6
lsp-refresh-interval	12-6
mac deny-src-mac-filter-group	7-58
mac filter	6-84
mac filter log-enable	6-87
mac filter-group	7-58
mac filter-group log-enable	7-59
mac-age-time	6-84
management-vlan	33-7
master	34-2
master-vlan	36-1, 37-1
match	19-1
max-attribute-entries	13-9
maximum	40-2
maximum-number-of-lsa	11-11
maximum-paths	12-6, 13-10
max-lsp-lifetime	12-7
max-neighbors	13-10

Table 3.1: Complete Layer 2/3 Command List (Continued)

max-routes	11-12, 13-10
maxreq	39-2
mcast-hw-replic-disable	6-87
mcast-hw-replic-oar	6-87
md	5-31
med-missing-as-worst	13-11
member-group	36-2, 37-2
member-vlan	36-2, 37-2
message-interval	17-3
metric	14-2, 26-4, 29-2
metric-style	12-7
metric-type	11-12
metro-ring	33-7
mirror-port	6-88
mkdir	5-31
module	6-89
monitor	7-59, 9-3
more	5-32
mpls-interface	22-2
mrinfo	5-33
msdp-peer	15-1
mtraceroute	5-33
mtu	7-59
multicast filter	6-89
multicast limit	6-90, 7-60
multicast-flooding	33-7
multipath	13-11
mvlan-mode	7-60
name	34-2
nbr-timeout	16-2, 17-3
ncopy flash primary | secondary slot1 | slot2 <to-name>	5-34
ncopy flash primary | secondary tftp <ip-addr> <from-name>	5-34

Table 3.1: Complete Layer 2/3 Command List (Continued)

ncopy running slot1 | slot2 <to-name>	5-34
ncopy running-config tftp <ip-addr> <from-name>	5-35
ncopy slot1 | slot2 <from-name> flash primary | secondary	5-35
ncopy slot1 | slot2 <from-name> running	5-35
ncopy slot1 | slot2 <from-name> slot1 | slot2 [<to-name>]	5-36
ncopy slot1 | slot2 <from-name> start	5-36
ncopy slot1 | slot2 <from-name> tftp <ip-addr> [<to-name>]	5-36
ncopy start slot1 | slot2 <to-name>	5-37
ncopy startup-config tftp <ip-addr> <from-name>	5-37
ncopy tftp <ip-addr> <from-name> flash primary | secondary	5-37
ncopy tftp <ip-addr> <from-name> running-config	5-38
ncopy tftp <ip-addr> <from-name> slot1 | slot2 [<to-name>]	5-38
ncopy tftp <ip-addr> <from-name> startup-config	5-38
neighbor	10-3, 13-12
net <area-id>.<system-id>.<sel>	12-8
netbios-proto	33-8
network	13-15
next-hop-enable-default	13-16
next-hop-recursion	13-16
no	6-90
non-preempt-mode	20-2, 21-3, 35-5
offset-list	10-4
other-proto	33-9
out-segment	29-3
owner	20-2
page-display	5-39
password-change	6-90
path	22-2
perf-mode	6-90
permit redistribute	10-4, 11-12

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

show super-span	43-141
show tech-support	43-141
show telnet	43-142
show terminal	43-143
show topology-group	43-143
show trunk	43-144
show users	43-145
show version	43-145
show vlan-group	43-146
show vlans	43-147
show vm-map	43-147
show vm-state	43-148
show vsrp	43-149
show vsrp aware	43-150
show web-connection	43-151
show who	43-151
skip-page-display	5-45
snmp disable	6-116
snmp-client	6-116
snmp-server community	6-117
snmp-server contact	6-118
snmp-server enable ethernet	6-118
snmp-server enable traps	6-118
snmp-server enable traps holddown-time	6-119
snmp-server enable traps link-change	7-74
snmp-server enable vlan	6-119
snmp-server engineid	6-119
snmp-server group	6-120
snmp-server host	6-121
snmp-server location	6-121
snmp-server pw-check	6-122
snmp-server trap-source	6-122
snmp-server user	6-122

Table 3.1: Complete Layer 2/3 Command List (Continued)

Table 3.1: Complete Layer 2/3 Command List (Continued)

system-max	6-131
table-map	13-19
tacacs-server	6-132
tagged	33-16
tag-type	6-133
tcp keepalive	42-2
telnet	5-47
telnet access-group	6-134
telnet login-timeout	6-134
telnet server enable vlan	6-134
telnet server suppress-reject-message	6-134
telnet-client	6-135
telnet-server	6-135
telnet-timeout	6-135
temperature shutdown	5-47
temperature warning	5-48
terminal length	6-136
terminal monitor	5-48
tftp client enable vlan	6-136
tie-breaking	26-6
timeout quiet-period	39-3
timeout tx-period	39-3
timeout re-authperiod	39-3
timeout security-hold-time	39-3
timers	13-19
timers lsa-group-pacing	11-15
timers spf	11-15
to	26-6, 29-3
topology-group	6-136
traceroute	4-3
track-port	20-3, 21-3, 35-7
traffic-eng max-burst	26-6, 29-4
traffic-eng max-rate	26-6, 29-3

Table 3.1: Complete Layer 2/3 Command List (Continued)

traffic-eng mean-rate	26-6, 29-4
traffic-engineering ospf	24-3
trap	11-16
trigger-interval	16-4
trunk	6-137
trunk deploy	6-138
ttl-threshold	14-2
undelete	5-49
unknown-unicast limit	6-138, 7-76
untagged	30-2, 32-2, 33-17
update-time	10-5, 13-20
uplink-switch	33-17
username	6-139
use-vrrp-path	10-6
violation restrict	40-3
violation shutdown	40-3
virtual-interface-mac	6-139
vlan	6-140
vlan-dynamic-discovery	6-140
vlan max-vlans	6-141
vlan-group	6-140
vll	22-3
vll-peer	30-2
vm boot	5-49, 6-141
vm copy flash flash	5-50
vm copy tftp flash	5-50
vm vm-map	6-142
vm vm-map per-port-dma	6-142
vm vm-map port-dma	6-142
vpls	22-4
vpls-peer	32-2
vsrp vrid	33-17
web access-group	6-143

Table 3.1: Complete Layer 2/3 Command List (Continued)

web-client	6-143
web-management	6-143
web-management enable vlan	6-144
whois	5-50
write memory	6-144
write terminal	6-144

Commands Listed by CLI Level

The following sections contain tables that list the CLI commands within each level of the CLI.

EXEC Level

There are two different levels of EXEC commands, the ***User Level*** and the ***Privileged Level***. The User level commands are at the top of the CLI hierarchy. These are the first commands that you have access to when connected to the Layer 2 Switch or Layer 3 Switch through the CLI. At this level, you can view basic system information and verify connectivity but cannot make any changes to the Layer 2 Switch or Layer 3 Switch configuration. To make changes to the configuration, you must move to other levels of the CLI hierarchy. This is accomplished by the User EXEC level command enable at initial log-on. This command takes you to the Privileged EXEC level, from which you can reach the configuration command levels.

Table 3.2: User EXEC Commands

appletalk-ping	4-1
enable	4-1
enable <password>	4-1
enable <username> <password>	4-2
fastboot	4-2
ping	4-2
show	4-3
stop-traceroute	4-3
traceroute	4-3

Privileged Level

The Privileged EXEC level commands primarily enable you to transfer and store Layer 2 Switch or Layer 3 Switch software images and configuration files between the network and the system, and review the configuration.

You reach this level by entering **enable** [<password>] or **enable** <username> <password> at the User EXEC level.

Table 3.3: Privileged EXEC Commands

10gig copy	5-1
append	5-1
appletalk-ping	5-2
atm boot tftp	5-2
atm copy slot1 | slot2 flash	5-2
atm copy tftp flash	5-3
attrib	5-3
boot system bootp	5-4
boot system flash primary	5-4

Table 3.3: Privileged EXEC Commands (Continued)

Table 3.3: Privileged EXEC Commands (Continued)

Table 3.3: Privileged EXEC Commands (Continued)

Table 3.3: Privileged EXEC Commands (Continued)

Table 3.3: Privileged EXEC Commands (Continued)

CONFIG Commands

CONFIG commands modify the configuration of a Foundry Layer 2 Switch or Layer 3 Switch. This reference describes the following CONFIG CLI levels.

Global Level

The global CONFIG level allows you to globally apply or modify parameters for ports on the Layer 2 Switch or Layer 3 Switch. You reach this level by entering **configure terminal** at the privileged EXEC level.

Table 3.4: Global CONFIG Commands

Table 3.4: Global CONFIG Commands (Continued)

Table 3.4: Global CONFIG Commands (Continued)

Table 3.4: Global CONFIG Commands (Continued)

Table 3.4: Global CONFIG Commands (Continued)

Table 3.4: Global CONFIG Commands (Continued)

Table 3.4: Global CONFIG Commands (Continued)

Table 3.4: Global CONFIG Commands (Continued)

Table 3.4: Global CONFIG Commands (Continued)

Table 3.4: Global CONFIG Commands (Continued)

Redundancy Level

The Redundancy CONFIG level allows you to configure parameters on redundant management modules. You reach this level by entering **redundancy** at the global CONFIG level.

Table 3.5: Redundancy CONFIG Commands

Table 3.5: Redundancy CONFIG Commands (Continued)

exit	8-2
no	8-2
quit	8-2
show	8-2
sync-standby	8-2
write memory	8-3
write terminal	8-3

Interface Level

The interface level allows you to assign or modify specific port parameters on a port-by-port basis. You reach this level by entering **interface ethernet** <portnum>, **interface loopback** <num>, **interface ve** <num>, or **interface pos** <portnum> at the global CONFIG level.

Table 3.6: Interface Commands

appletalk address	7-1
appletalk cable-range	7-1
appletalk deny	7-1
appletalk deny additional-zones	7-2
appletalk permit	7-2
appletalk routing	7-2
appletalk zone-name	7-3
aps authenticate	7-3
aps force	7-3
aps group	7-4
aps lockout	7-4
aps manual	7-5
aps protect	7-5
aps revert	7-5
aps timers	7-6
aps working	7-6
atm-hcsadd	7-6
atm-pvc	7-6
atm-scram	7-8
atm vc-per-vp	7-9

Table 3.6: Interface Commands (Continued)

Table 3.6: Interface Commands (Continued)

Table 3.6: Interface Commands (Continued)

Table 3.6: Interface Commands (Continued)

Table 3.6: Interface Commands (Continued)

Table 3.6: Interface Commands (Continued)

write terminal	7-76

Trunk Level

The trunk level allows you to change parameters for statically-configured trunk groups. You reach this level by entering a **trunk** command with the appropriate port parameters.

Table 3.7: Trunk Commands

config-trunk-ind	9-1
disable	9-2
enable	9-2
end	9-3
exit	9-3
monitor	9-3
no	9-4
port-name	9-4
quit	9-4
show	9-4
write memory	9-4
write terminal	9-4

Router RIP Level (Layer 3 Switches only)

The RIP level allows you to configure parameters for the RIP routing protocol. You reach this level by entering the **router rip** command at the global CONFIG level.

Table 3.8: RIP Commands

default-metric	10-1
deny redistribute	10-1
end	10-2
exit	10-2
filter	10-2
filter-group	10-3
learn-default	10-3
neighbor	10-3
no	10-4
offset-list	10-4

Table 3.8: RIP Commands (Continued)

Router OSPF Level (Layer 3 Switches only)

The OSPF level allows you to configure parameters for the OSPF routing protocol. You reach this level by entering the **router ospf** command at the global CONFIG level.

Table 3.9: OSPF Commands

Table 3.9: OSPF Commands (Continued)

Router IS-IS Level (Layer 3 Switches only)

The IS-IS level allows you to configure parameters for the IS-IS routing protocol. You reach this level by entering the **router isis** command at the global CONFIG level.

Table 3.10: IS-IS Commands

Table 3.10: IS-IS Commands (Continued)

BGP Level (Layer 3 Switches only)

The BGP level allows you to configure Layer 3 Switches for Border Gateway Protocol version 4 (BGP4). You reach this level by entering the **router bgp** command at the global CONFIG level.

Table 3.11: BGP4 Commands

Table 3.11: BGP4 Commands (Continued)

fast-external-fallover	13-9
local-as	13-9
max-attribute-entries	13-9
maximum-paths	13-10
max-neighbors	13-10
max-routes	13-10
med-missing-as-worst	13-11
multipath	13-11
neighbor	13-12
network	13-15
next-hop-enable-default	13-16
next-hop-recursion	13-16
no	13-16
quit	13-16
readvertise	13-17
redistribute connected	13-17
redistribute ospf	13-17
redistribute rip	13-18
redistribute static	13-18
show	13-19
synchronization	13-19
table-map	13-19
timers	13-19
update-time	13-20
write memory	13-20
write terminal	13-20

IP Tunnel Level (Layer 3 Switches only)

The IP tunnel level allows you to define parameters for IP-in-IP tunnels to pass data through non-DVMRP and non-PIM IP multicast routers.

You reach this level by entering the **ip tunnel**… command at the interface CONFIG level.

Table 3.12: IP Tunnel Commands

encap-control	14-1

Table 3.12: IP Tunnel Commands (Continued)

end	14-1
exit	14-1
metric	14-2
no	14-2
quit	14-2
show	14-2
ttl-threshold	14-2
write memory	14-2
write terminal	14-3

MSDP Level (Layer 3 Switches only)

The MSDP level allows you to define parameters for MSDP.

You reach this level by entering the **router msdp** command at the Global CONFIG level.

Table 3.13: MSDP Commands

end	15-1
exit	15-1
msdp-peer	15-1
no	15-2
quit	15-2
sa-filter in	15-2
sa-filter originate	15-3
show	15-4
write memory	15-4
write terminal	15-5

Router DVMRP Level (Layer 3 Switches only)

The DVMRP level allows you to configure details for the DVMRP multicast protocol. You reach this level by entering the **router dvmrp** command at the global CONFIG level.

NOTE: The interface and IP tunnel parameters for DVMRP are configured at the Interface and IP Tunnel levels, respectively. See those sections of this reference for specific examples.

Table 3.14: DVMRP Commands

default-gateway	16-1
end	16-1

Table 3.14: DVMRP Commands (Continued)

exit	16-1
graft-retransmit-timer	16-2
nbr-timeout	16-2
no	16-2
probe-interval	16-2
prune-age	16-2
quit	16-3
report-interval	16-3
route-discard-timeout	16-3
route-expire-timeout	16-3
show	16-4
trigger-interval	16-4
write memory	16-4
write terminal	16-4

Router PIM Level (Layer 3 Switches only)

The PIM level allows you to configure parameters for the Protocol Independent Multicast (PIM) routing protocol. You reach this level by entering the **router pim** command at the global CONFIG level.

Table 3.15: PIM Commands

bsr-candidate	17-1
end	17-1
exit	17-2
graft-retransmit-timer	17-2
hello-timer	17-2
inactivity-timer	17-2
message-interval	17-3
nbr-timeout	17-3
no	17-3
prune-timer	17-3
quit	17-4
rp-address	17-4
rp-candidate	17-5
show	17-6

Table 3.15: PIM Commands (Continued)

spt-threshold	17-7
write memory	17-7
write terminal	17-7

Broadcast Filter Level

The broadcast filter level allows you to assign broadcast filters to specific ports. You reach this level by entering **broadcast filter**... at the global CONFIG level.

Table 3.16: Broadcast Filter Commands

end	18-1
exclude-ports	18-1
exit	18-2
no	18-2
quit	18-2
show	18-2
write memory	18-2
write terminal	18-3

Multicast Filter Level

The multicast filter level allows you to assign multicast filters to specific ports. You reach this level by entering **multicast filter**... at the global CONFIG level.

Table 3.17: Multicast Filter Commands

end	18-3
exclude-ports	18-3
exit	18-3
no	18-4
quit	18-4
show	18-4
write memory	18-4
write terminal	18-4

Real Server Level (Layer 3 Switches only)

The Real Server level allows you to configure router-based health check parameters for a Layer 3 Switch to assist with Globally-distributed Server Load Balancing (SLB). See the "Route Health Injection" chapter of the *Foundry Enterprise Configuration and Management Guide*. You reach this level by entering the **server real**... command at the global CONFIG level.

Table 3.18: Real Server Commands

end	41-1
exit	41-1
no	41-1
port <num> disable	41-1
port <num> keepalive	41-2
port <num> status_code	41-2
port <num> url	41-3
quit	41-3
show	41-3
write memory	41-3
write terminal	41-3

Router VRRP Level (Layer 3 Switches only)

The VRRP level allows you to configure parameters for the Virtual Router Redundancy Protocol (VRRP). You reach this level by entering the **router vrrp** command at the global CONFIG level, then entering the **ip vrrp vrid** <num> command at the interface configuration level.

Table 3.19: VRRP Commands

activate	20-1
advertise backup	20-1
backup	20-1
backup-hello-interval	20-1
dead-interval	20-1
end	20-2
exit	20-2
hello-interval	20-2
ip-address	20-2
no	20-2
non-preempt-mode	20-2
owner	20-2

Table 3.19: VRRP Commands (Continued)

quit	20-3
show	20-3
track-port	20-3
write memory	20-3
write terminal	20-3

Router VRRPE Level (Layer 3 Switches only)

The VRRPE level allows you to configure parameters for VRRP Extended (VRRP). You reach this level by entering the **router vrrp-extended** command at the global CONFIG level, then entering the **ip vrrp-extended vrid** <num> command at the interface configuration level.

Table 3.20: VRRPE Commands

activate	21-1
advertise backup	21-1
backup	21-1
backup-hello-interval	21-1
dead-interval	21-1
disable	21-2
enable	21-2
end	21-2
exit	21-2
hello-interval	21-2
ip address	21-2
ip-address	21-2
no	21-3
non-preempt-mode	21-3
quit	21-3
show	21-3
track-port	21-3
write memory	21-3
write terminal	21-3

MPLS Level

The MPLS level allows you to configure Multiprotocol Label Switching on Foundry NetIron Internet Routers. You reach this level by entering the **router mpls** command at the Global CONFIG level.

Table 3.21: MPLS commands

MPLS Interface Level

The MPLS interface level allows you to configure parameters for an individual MPLS-enabled interface. You reach this level by entering the **mpls-interface pos** <interface number> command at the MPLS level.

Table 3.22: MPLS interface commands

Table 3.22: MPLS interface commands (Continued)

write terminal	23-2

MPLS Policy Level

The MPLS policy level allows you to configure global MPLS parameters. You reach this level by entering the **policy** command at the MPLS level.

Table 3.23: MPLS policy commands

admin-group	24-1
end	24-1
exit	24-1
no	24-2
propagate-ttl	24-2
quit	24-2
retry-limit	24-2
retry-time	24-2
show	24-3
traffic-engineering ospf	24-3
write memory	24-3
write terminal	24-3

MPLS Path Level

The MPLS path level allows you to configure paths that specify a route across an MPLS domain. You reach this level by entering the **path** <path name> command at the MPLS level.

Table 3.24: MPLS path commands

delete	25-1
end	25-1
exit	25-1
insert	25-2
loose	25-2
no	25-2
quit	25-2
show	25-2
strict	25-2
write memory	25-3

Table 3.24: MPLS path commands (Continued)

write terminal	25-3

MPLS Signalled LSP Level

The MPLS signalled LSP level allows you to configure dynamically established LSPs. You reach this level by entering the **lsp** <name> command at the MPLS level.

Table 3.25: MPLS signalled LSP commands

cos	26-1
cspf	26-1
disable	26-1
enable	26-2
end	26-2
exclude-any	26-2
exit	26-2
from	26-2
hop-limit	26-2
include-all	26-3
include-any	26-3
install	26-3
metric	26-4
no	26-4
primary-path	26-4
priority	26-4
quit	26-5
record	26-5
secondary-path	26-5
shortcuts ospf	26-5
show	26-5
tie-breaking	26-6
to	26-6
traffic-eng max-rate	26-6
traffic-eng mean-rate	26-6
traffic-eng max-burst	26-6
write memory	26-7

Table 3.25: MPLS signalled LSP commands (Continued)

write terminal	26-7

MPLS Secondary Path Level

The MPLS secondary path level allows you to configure redundant paths for an LSP. You reach this level by entering the **secondary-path** <path name> command at the MPLS signalled LSP level.

Table 3.26: MPLS secondary path commands

cos	27-1
cspf	27-1
end	27-1
exclude-any	27-2
exit	27-2
hop-limit	27-2
include-all	27-2
include-any	27-3
no	27-3
priority	27-3
quit	27-3
record	27-3
show	27-4
standby	27-4
traffic-eng max-rate	27-4
traffic-eng mean-rate	27-4
traffic-eng max-burst	27-4
write memory	27-5
write terminal	27-5

MPLS RSVP Level

The MPLS RSVP level allows you to configure RSVP parameters for use in signalling LSPs. You reach this level by entering the **rsvp** command at the MPLS level.

Table 3.27: MPLS rsvp commands

end	28-1
exit	28-1
no	28-1

Table 3.27: MPLS rsvp commands (Continued)

quit	28-1
refresh-interval	28-2
refresh-multiple	28-2
show	28-2
write memory	28-2
write terminal	28-2

MPLS Static LSP Level

The MPLS static LSP level allows you to configure static LSPs. You reach this level by entering the **static-lsp** <name> command at the MPLS level.

Table 3.28: MPLS static LSP commands

cos	29-1
disable	29-1
enable	29-1
end	29-1
exit	29-2
in-segment	29-2
install	29-2
metric	29-2
no	29-3
out-segment	29-3
quit	29-3
show	29-3
to	29-3
traffic-eng max-rate	29-3
traffic-eng mean-rate	29-4
traffic-eng max-burst	29-4
write memory	29-4
write terminal	29-4

MPLS VLL Level

The MPLS VLL level allows you to configure MPLS Virtual Leased Line. You reach this level by entering the **vll** <name> command at the MPLS level.

Table 3.29: MPLS VLL commands

MPLS LDP Level

The MPLS LDP level allows you to configure Label Distribution Protocol (LDP). You reach this level by entering the **ldp** command at the MPLS level.

Table 3.30: MPLS LDP commands

MPLS VPLS Level

The MPLS VPLS level allows you to configure MPLS Virtual Private LAN Services. You reach this level by entering the **vpls** <name> command at the MPLS level.

Table 3.31: MPLS VPLS commands

VLAN Level

Policy-based VLANs allow you to assign VLANs on a protocol (IP, IPX, Decnet, AppleTalk, NetBIOS, Others), sub-net (IP sub-net and IPX network), AppleTalk cable, port, or 802.1q tagged basis.

You reach this level by entering the **vlan** <vlan-id> **by port** command at the Global CONFIG Level.

Table 3.32: VLAN Commands

Table 3.32: VLAN Commands (Continued)

Metro Ring Level

Metro rings provide Layer 2 connectivity and fast failover in ring topologies.

You reach this level by entering the **metro-ring** <ring-id> command at the Global CONFIG Level.

Table 3.33: Metro Ring Commands

Table 3.33: Metro Ring Commands (Continued)

end	34-1
exit	34-2
hello-time	34-2
master	34-2
name	34-2
no	34-3
preforwarding-time	34-3
quit	34-3
ring-interface	34-3
show	34-4
write memory	34-4
write terminal	34-4

VSRP Level

The VSRP level allows you to configure parameters for the Virtual Switch Redundancy Protocol (VSRP). You reach this level by entering the **vsrp vrid** <num> command at the VLAN configuration level, then entering the **vsrp vrid** <num> command at the VLAN configuration level.

Table 3.34: VSRP Commands

activate	35-1
advertise backup	35-1
backup	35-1
backup-hello-interval	35-2
dead-interval	35-2
disable	35-3
enable	35-3
end	35-3
exit	35-3
hello-interval	35-3
hold-down-interval	35-4
include-port	35-4
initial-ttl	35-4
ip address	35-5
ip-address	35-5

Table 3.34: VSRP Commands (Continued)

no	35-5
non-preempt-mode	35-5
quit	35-6
save-current-values	35-6
show	35-7
track-port	35-7
write memory	35-7
write terminal	35-7

Topology Group Level

A topology group enables you to control the Layer 2 protocol configuration and Layer 2 state of a set of ports in multiple VLANs based on the configuration and states of those ports in a single master VLAN. One instance of the Layer 2 protocol controls all the VLANs.

You reach this level by entering the **topology-group** <group-id> command at the Global CONFIG Level.

Table 3.35: Topology Group Commands

end	36-1
exit	36-1
master-vlan	36-1
member-group	36-2
member-vlan	36-2
no	36-3
quit	36-3
show	36-3
write memory	36-3
write terminal	36-3

STP Group Level

STP groups enable you to manage multiple port-based VLANs using the same spanning tree.

You reach this level by entering the **stp-group** <num> command at the Global CONFIG Level.

Table 3.36: STP Group Commands

end	37-1
exit	37-1
master-vlan	37-1

Table 3.36: STP Group Commands (Continued)

member-group	37-2
member-vlan	37-2
no	37-2
quit	37-2
show	37-3
write memory	37-3
write terminal	37-3

802.1X Port Security Level

The 802.1X port security level allows you to configure the 802.1X port security. You reach this level by entering the **dot1x-enable** command at the at the Global level.

Table 3.37: 802.1X Port Security commands

enable	39-1
end	39-1
exit	39-1
maxreq	39-2
no	39-2
quit	39-2
re-authentication	39-2
servertimeout	39-2
show	39-3
supptimeout	39-3
timeout quiet-period	39-3
timeout tx-period	39-3
timeout re-authperiod	39-3
timeout security-hold-time	39-3
write memory	39-4
write terminal	39-4

MAC Port Security Level

The MAC port security level allows you to configure the port security feature. You reach this level by entering the **port security** command at the at the Global or Interface levels.

Table 3.38: MAC Port Security commands

GVRP Level

GARP VLAN Registration Protocol (GVRP) is a Generic Attribute Registration Protocol (GARP) application that provides VLAN registration service by means of dynamic configuration (registration) and distribution of VLAN membership information.

You reach the GVRP level by entering the **gvrp-enable** command at the Global CONFIG Level.

Table 3.39: GVRP Commands

Table 3.39: GVRP Commands (Continued)

show	38-3
write memory	38-4
write terminal	38-4

Application Port Level (Layer 3 Switches only)

The Application Port level allows you to configure health check parameters for a TCP HTTP port. The commands at this level apply only when you are configuring a Layer 3 Switch to assist ServerIrons, third-party SLBs, or web servers with globally-distributed SLB. See the "Route Health Injection" chapter of the *Foundry Enterprise Configuration and Management Guide*. You reach this level by entering the **server port http** | <tcp/udp-portnum> command at the global CONFIG level.

NOTE: If you enter **server port ?**, numerous well-known port names are listed. The current software release supports only HTTP ports.

Table 3.40: Application Port Commands

end	42-1
exit	42-1
no	42-1
quit	42-1
show	42-2
tcp keepalive	42-2
write memory	42-2
write terminal	42-2

Route Map Level (Layer 3 Switch only)

The Route Map level allows you to configure parameters for a BGP4 route map. You reach this level by entering the **route-map** <name> command at the global CONFIG level.

Table 3.41: Route Map Commands

end	19-1
exit	19-1
match	19-1
no	19-2
quit	19-2
set	19-3
set mirror-interface	19-4
show	19-5

Table 3.41: Route Map Commands (Continued)

write memory	19-5
write terminal	19-5

Show Commands

The show commands display configuration information and statistics. You can enter these commands from any level of the CLI.

Table 3.42: Show Commands

show 802-1w	43-1
show 802-1w detail	43-4
show aaa	43-6
show access-list	43-7
show acl-on-arp	43-8
show appletalk arp	43-8
show appletalk cache	43-8
show appletalk globals	43-9
show appletalk interface	43-9
show appletalk route	43-10
show appletalk traffic	43-11
show appletalk zone	43-12
show arp	43-13
show atm map	43-14
show atm-state	43-14
show atm vc	43-15
show cam	43-15
show cam-partition	43-18
show chassis	43-20
show clock	43-21
show configuration	43-21
show default	43-21
show dot1x	43-24
show dot1x config	43-25
show dot1x statistics	43-25
show fdp entry	43-26

Table 3.42: Show Commands (Continued)

Table 3.42: Show Commands (Continued)

Table 3.42: Show Commands (Continued)

Table 3.42: Show Commands (Continued)

Table 3.42: Show Commands (Continued)

Table 3.42: Show Commands (Continued)

Table 3.42: Show Commands (Continued)

Chapter 4
User EXEC Commands

appletalk-ping

Verifies connectivity to an AppleTalk network and node.

EXAMPLE:

To verify connectivity to node 50 on network 100, enter the following:

```
BigIron> appletalk-ping 100.50
```

Syntax: appletalk-ping <network.node>

Possible values: N/A

Default value: N/A

enable

At initial startup, you enter this command to access the privileged EXEC level of the CLI. You access subsequent levels of the CLI using the proper launch commands.

You can assign a permanent password with the enable password… command at the global level of the CONFIG command structure. To reach the global level, enter configure terminal. Until a password is assigned, you have access only to the user EXEC level.

EXAMPLE:

```
BigIron> enable
```

Syntax: enable

Possible values: N/A

Default value: No system default

enable <password>

Once an Enable password is defined for the device, you must enter this command along with the defined password to access the Privileged EXEC level of the CLI.

EXAMPLE:

```
BigIron> en whatever
BigIron#
```

Syntax: enable <password>

Possible values: N/A

Default value: N/A

enable <username> <password>

If local access control, RADIUS authentication, or TACACS/TACACS+ authentication is configured on the device, you need to enter a user name and password to access the Privileged EXEC level.

EXAMPLE:

```
BigIron> en waldo whereis

BigIron#
```

Syntax: enable <username> <password>

Possible values: a valid username and password for the authentication method used by the device

Default value: N/A

fastboot

By default, this option is turned off, to provide a three-second pause to allow you to break into the boot prompt, if necessary. Use fastboot on to turn this option on and eliminate the three-second pause. To turn this feature off later, enter the command, **fastboot off**. Fastboot changes will be saved automatically but will not become active until after a system reset.

To execute an immediate reload of the boot code from the console without a three-second delay, enter the **fast reload** command. The fast reload command is available at the privileged EXEC level.

EXAMPLE:

```
BigIron> fastboot on
```

Syntax: fastboot [on | off]

Possible values: on or off

Default value: off

ping

Verifies connectivity to a Foundry Layer 2 Switch, Layer 3 Switch, or other device. The command performs an ICMP echo test to confirm connectivity to the specified device.

NOTE: If you address the ping to the IP broadcast address, the device lists the first four responses to the ping.

EXAMPLE:

```
BigIron> ping 192.22.2.33
```

Syntax: ping <ip addr> | <hostname> [source <ip addr>] [count <num>] [timeout <msec>] [ttl <num>] [size <byte>] [quiet] [numeric] [no-fragment] [verify] [data <1-to-4 byte hex>] [brief]

The only required parameter is the IP address or host name of the device.

NOTE: If the device is a Foundry Layer 2 Switch or Layer 3 Switch, you can use the host name only if you have already enabled the Domain Name Server (DNS) resolver feature on the device from which you are sending the ping. See the “Configuring IP” chapter of the *Foundry Enterprise Configuration and Management Guide*.

The **source** <ip addr> specifies an IP address to be used as the origin of the ping packets.

The **count** <num> parameter specifies how many ping packets the device sends. You can specify from 1 – 4294967296. The default is 1.

The **timeout** <msec> parameter specifies how many milliseconds the Foundry device waits for a reply from the pinged device. You can specify a timeout from 1 – 4294967296 milliseconds. The default is 5000 (5 seconds).

The **ttl** <num> parameter specifies the maximum number of hops. You can specify a TTL from 1 – 255. The default is 64.

The **size** <byte> parameter specifies the size of the ICMP data portion of the packet. This is the payload and does not include the header. You can specify from 0 – 4000. The default is 16.

The **no-fragment** parameter turns on the "don't fragment" bit in the IP header of the ping packet. This option is disabled by default.

The **quiet** parameter hides informational messages such as a summary of the ping parameters sent to the device and instead only displays messages indicating the success or failure of the ping. This option is disabled by default.

The **verify** parameter verifies that the data in the echo packet (the reply packet) is the same as the data in the echo request (the ping). By default the device does not verify the data.

The **data** <1 – 4 byte hex> parameter lets you specify a specific data pattern for the payload instead of the default data pattern, "abcd", in the packet's data payload. The pattern repeats itself throughout the ICMP message (payload) portion of the packet.

NOTE: For numeric parameter values, the CLI does not check that the value you enter is within the allowed range. Instead, if you do exceed the range for a numeric value, the software rounds the value to the nearest valid value.

The **brief** parameter causes ping test characters to be displayed. The following ping test characters are supported:

! Indicates that a reply was received.

. Indicates that the network server timed out while waiting for a reply.

U Indicates that a destination unreachable error PDU was received.

I Indicates that the user interrupted ping.

Possible values: see above

Default value: see above

show

Displays a variety of configuration and statistical information about the device. See "Show Commands" on page 43-1.

stop-traceroute

Stops an initiated trace on a Foundry device.

EXAMPLE:

```
BigIron> stop-traceroute
```

Syntax: stop-traceroute

Possible values: N/A

Default value: N/A

traceroute

Allows you to trace the path from the current Foundry device to a host address.

The CLI displays trace route information for each hop as soon as the information is received. Traceroute requests display all responses to a given TTL. In addition, if there are multiple equal-cost routes to the destination, the Foundry device displays up to three responses by default.

EXAMPLE:

```
BigIron> traceroute 192.33.4.7 minttl 5 maxttl 5 timeout 5
```

Syntax: traceroute <host-ip-addr> [maxttl <value>] [minttl <value>] [numeric] [timeout <value>] [source-ip <ip addr>]

Possible and default values:

minttl – minimum TTL (hops) value: Possible values are 1 – 255. Default value is 1 second.

maxttl – maximum TTL (hops) value: Possible values are 1 – 255. Default value is 30 seconds.

timeout – Possible values are 1 – 120. Default value is 2 seconds.

numeric – Lets you change the display to list the devices by their IP addresses instead of their names.

source-ip <ip addr> – Specifies an IP address to be used as the origin for the traceroute.

Chapter 5
Privileged EXEC Commands

10gig copy

Upgrades the Field-Programmable Gate Arrays (FPGAs) on a 10 Gigabit Ethernet module.

EXAMPLE:

```
BigIron# 10gig copy tftp 10.10.10.10 rxbmgr.bin
BigIron# 10gig copy tftp 10.10.10.10 rxpp.bin
BigIron# 10gig copy tftp 10.10.10.10 txaccum.bin
BigIron# 10gig copy tftp 10.10.10.10 txpp.bin
BigIron# 10gig copy tftp 10.10.10.10 ageram.bin
```

Syntax: 10gig copy tftp | slot1 | slot2 flash <ip-addr> <filename> [module <slotnum>]

tftp | slot1 | slot2 – specifies the location of the FPGA file. The **tftp** parameter indicates that the file is on a TFTP server. The **slot1** and **slot2** parameters indicate that the file is on a PCMCIA flash card. Specify **slot1** if the file is on the flash card in PCMCIA slot 1. Specify **slot2** if the file is on the flash card in PCMCIA slot 2.

<ip-addr> – specifies the IP address of the TFTP server, if you specify **tftp**.

<filename> – specifies the FPGA file name.

NOTE: You can store and copy the FPGA files using any valid filename; however, Foundry recommends that you use the file names listed in the "Software Image Files" section of the release notes. The device uses information within the files to install them in the correct FPGAs. The **show flash** command lists the FPGAs.

Possible values: See above

Default value: N/A

append

Appends a file on a PCMCIA flash card to the end of another file.

EXAMPLE:

```
BigIron# append newacls.cfg startup-config.cfg
```

This command appends a file called "newacls.cfg" to the end of a file called "startup-config.cfg" file. This example assumes that both files are present on the PCMCIA slot and in the subdirectory level that currently have the management focus.

The following command appends a file in the current subdirectory to the end of a file in another subdirectory:

```
BigIron# append newacls.cfg \TEST\startup-config.cfg
```

Syntax: append [<from-card> <to-card>] [\<from-dir-path>\]<from-name> [\<to-dir-path>\]<to-name>

The <from-card> and <to-card> parameters specify the source and destination flash cards when you are appending a file on one flash card to a file located on another flash card.

The [\<from-dir-path>\]<from-name> parameter specifies the file you are adding to the end of another file. If the file is not located in the current subdirectory (the subdirectory that currently has the management focus), specify the subdirectory path in front of the file name.

The [\<to-dir-path>\]<to-name> parameter specifies the file to which you are appending the other file. If the file is not located in the current subdirectory, specify the subdirectory path in front of the file name.

Possible values: See above

Default value: N/A

appletalk-ping

Verifies connectivity to an AppleTalk network and node.

EXAMPLE:

To verify connectivity to node 50 on network 100, enter the following:

```
BigIron# appletalk-ping 100.50
```

Syntax: appletalk-ping <network.node>

Possible values: See above

Default value: N/A

atm boot tftp

Boots the ATM module from a TFTP server.

NOTE: This command applies only to the interactive boot method. See “atm boot interactive” on page 6-14.

EXAMPLE:

To boot the ATM module from a TFTP server, enter a command such as the following at the Privileged EXEC level of the CLI:

```
BigIron# atm boot tftp 209.157.22.26 A2R07205.bin
```

Syntax: atm boot tftp <ip-addr> <atm-image-file-name>

The <ip-addr> parameter specifies the IP address of the TFTP server.

The <atm-image-file-name> parameter lists the name of the image file you want the module to boot from the TFTP server.

Possible values: See above.

Default value: N/A

atm copy slot1 | slot2 flash

Copies an ATM image file from a flash card to an ATM module’s flash memory.

EXAMPLE:

To copy an ATM image file from a flash card onto all the ATM modules in the chassis, enter a command such as the following:

```
BigIron# atm copy slot1 flash A2R07205.bin primary
```

Syntax: atm copy slot1 | slot2 flash <atm-image-file-name> boot | primary | secondary [<slotnum>]

The command in this example copies an ATM image file named A2R07205.bin from the flash card in slot 1 to all the ATM modules in the chassis.

The **boot** parameter specifies that you are upgrading the boot code rather than the flash code.

The **primary** and **secondary** parameters identify either the primary or secondary flash on the module. For each command, the parameter specifies the destination of the copy operation.

The **slot** parameter specifies a chassis slot. This parameter is optional. If you specify a slot number, the upgrade affects only the module in the slot you specify. If you do not specify a slot, the upgrade affects all the ATM modules in the chassis.

- Slots in a four-slot chassis are numbered 1 – 4, from top to bottom.
- Slots in an eight-slot chassis are numbered 1 – 8, from left to right.
- Slots in a fifteen-slot chassis are numbered 1 – 15, from left to right.

To copy an ATM image file from a flash card onto a specific ATM module, enter a command such as the following:

```
BigIron# atm copy slot1 A2R07205.bin flash primary 4
```

The command in this example copies the specified image file onto the ATM module in chassis slot 4 only, but does not copy the file to other ATM modules in the chassis.

Possible values: See above.

Default value: N/A

atm copy tftp flash

Upgrades the software in the primary flash on all ATM modules in a Chassis device.

EXAMPLE:

```
BigIron# atm copy tftp flash 109.157.22.26 A2R07205.bin primary
```

Syntax: atm copy tftp flash <tftp-server-ip-addr> <atm-image-file-name> boot | primary | secondary [<slotnum>]

The **boot** parameter specifies that you are upgrading the boot code rather than the flash code.

The **primary** and **secondary** parameters identify either the primary or secondary flash on the module. For each command, the parameter specifies the destination of the copy operation.

The **slot** parameter specifies a chassis slot. This parameter is optional. If you specify a slot number, the upgrade affects only the module in the slot you specify. If you do not specify a slot, the upgrade affects all the ATM modules in the chassis.

- Slots in a four-slot chassis are numbered 1 – 4, from top to bottom.
- Slots in an eight-slot chassis are numbered 1 – 8, from left to right.
- Slots in a fifteen-slot chassis are numbered 1 – 15, from left to right.

To upgrade the software on the secondary flash on the ATM module in chassis slot 6 only, enter the following command:

```
BigIron# atm copy tftp flash 109.157.22.26 A2R07205.bin secondary 6
```

Possible values: See above.

Default value: N/A

attrib

Changes the read-write attribute of a file on a flash card in a Management IV module's PCMCIA slot.

The read-write attribute specifies whether a file on a flash card can be changed or deleted.

- Read-only – You can display or copy the file but you cannot replace (copy over) or delete the file.
- Read-write – You can replace (copy over) or delete the file. This is the default.

Use the following method to change the read-write attribute of a file.

EXAMPLE:

To protect a file from accidental changes by changing the read-write attribute from read-write to read-only, enter a command such as the following:


```
BigIron# attrib ro goodcfg.cfg
```

Syntax: attrib [slot1 | slot2] ro | rw <file-name>

To determine the read-write attribute of a file, use the **dir** command to list the directory information for the file. Files set to read-only are listed with "R" in front of the file name. See "dir" on page 5-25.

To change all files on a flash card to read-only, enter a command such as the following:

```
BigIron# attrib ro *.*
```

This command changes the read-write attribute for all files on the flash card that currently has the management focus to read-only.

Possible values: See above.

Default value: rw (read-write)

boot system bootp

Initiates a system boot from a BootP server. You can specify the preferred initial boot source and boot sequence in the startup-config file. If upon boot, the user-specified boot source and sequence fails, then by default, the Foundry Layer 2 Switch or Layer 3 Switch will attempt to load the software image from a different source. The following sources will be tried one at a time, in the order noted, until a software load is successful.

- flash primary
- flash secondary
- bootp

If the image does not load successfully from the above sources, you are prompted to enter alternative locations from which to load an image:

- boot system bootp
- boot system flash primary
- boot system flash secondary
- boot system tftp

EXAMPLE:

```
BigIron# boot sys bootp
```

Syntax: boot system bootp

Possible values: N/A

Default value: N/A

boot system flash primary

Initiates a system boot from the primary software image stored in flash.

EXAMPLE:

```
BigIron# boot sys fl pri
```

Syntax: boot system flash primary

Possible values: N/A

Default value: N/A

boot system flash secondary

Initiates a system boot from the secondary software image stored in flash.

EXAMPLE:

```
BigIron# boot sys fl sec
```

Syntax: boot system flash secondary

Possible values: N/A

Default value: N/A

boot system slot1 | slot2

Initiates a system boot from an image file on a PCMCIA flash card.

EXAMPLE:

To reboot the device using a software image file on the flash card, enter a command such as the following at the Privileged Exec level of the CLI:

```
BigIron# boot system slot1 B2R07100.bin
```

The command in this example reboots the device using the image file B2R07100.bin located on the PCMCIA flash card in slot 1. This example assumes the image file is in the root directory on the flash card. If the image file is in a subdirectory, specify the subdirectory path. For example, to boot using an image in a subdirectory called “B2R”, enter a command such as the following:

```
BigIron# boot system slot1 \B2R\B2R07100.bin
```

Syntax: boot system slot1 | slot2 [\<dir-path>\]<file-name>

The **slot1** | **slot2** parameter indicates the flash card slot.

The <file-name> parameter specifies the file name. If the file is in a subdirectory, specify the subdirectory path in front of the file name. If the file name you specify is not a full path name, the CLI assumes that the name (and path, if applicable) you enter are relative to the subdirectory that currently has the management focus.

Possible values: See above

Default value: N/A

boot system tftp

Initiates a system boot of the software image from a TFTP server.

EXAMPLE:

```
BigIron# boot sys tftp 192.22.33.44 current.img
```

Syntax: boot system tftp <ip-addr> <filename>

Possible values: N/A

Default value: N/A

NOTE: Before entering the TFTP boot command, you must first assign an IP address, IP mask and default gateway (if applicable) at the boot prompt as shown.

EXAMPLE:

```
boot> ip address 192.22.33.44 255.255.255.0
```

```
boot> ip default-gateway 192.22.33.1
```

You now can proceed with the **boot system tftp**… command.

cd

Another form of the **chdir** command. See “chdir” on page 5-5.

chdir

Switches the management focus from one flash card in a Management IV module’s PCMCIA slot to the other slot.

The effect of file management commands depends on the flash card that has the management focus. For example, if you enter a command to delete a file, the software deletes the specified file from the flash card that currently has the management focus.

EXAMPLE:

To switch the focus of the CLI from one flash card to the other, enter a command such as the following:

```
BigIron# cd slot2
BigIron#
```

Syntax: cd | chdir slot1 | slot2

Syntax: cd | chdir <dir-name>

When you enter the **cd** command, the software changes the management focus to the slot or subdirectory path you specify, then displays a new command prompt.

If a slot you specify does not contain a flash card, the software displays the message shown in the following example.

```
BigIron# cd slot2
The system can not find the drive specified
```

To switch the management focus to a different subdirectory, enter a command such as the following:

```
BigIron# cd PLOOK
Current directory of slot1 is: \PLOOK
```

This command changes the focus from the root directory level (\) to the subdirectory named “PLOOK”.

If you specify an invalid subdirectory path, the CLI displays a message such as the following:

```
BigIron# cd PLOOK
Path not found
```

If you are certain the path you specified exists, make sure you are at the correct level for reaching the path. For example, if you are already at the PLOOK level, the CLI cannot find the subdirectory “\PLOOK” because it is not a subdirectory from the level that currently has the management focus.

Possible values: N/A

Default value: N/A

clear acl-on-arp

To clear the filter count for all interfaces on the device, enter a command such as the following:

```
BigIron(config)# clear acl-on-arp
```

Syntax: clear acl-on-arp

The command resets the filter count on all interfaces in a device back to zero

Possible values: N/A

Default value: N/A

clear appletalk arp

Erases all data currently resident in the AppleTalk ARP table, as displayed by the **show appletalk arp** command.

EXAMPLE:

```
BigIron# clear appletalk arp
```

Syntax: clear appletalk arp

Possible values: N/A

Default value: N/A

clear appletalk cache

Erases all learned data from non-local networks that is currently resident in the AppleTalk cache (forwarding table), as displayed by the **show appletalk cache** command.

EXAMPLE:

To remove all non-local entries from the AppleTalk cache, enter the following:

```
BigIron# clear appletalk cache
```

NOTE: Local routes are indicated by zeros in a show appletalk cache display. All entries not marked with 0.0 or 0000.0000.0000 will be erased.

Syntax: clear appletalk cache

Possible values: N/A

Default value: N/A

clear appletalk route

Erases all learned routes and zones (non-local routes and zones) currently resident in the AppleTalk routing table, as displayed by the **show appletalk route** command.

EXAMPLE:

To remove all non-local entries from the AppleTalk routing table, enter the following:

```
BigIron# clear appletalk route
```

NOTE: Local routes are indicated by zeros as shown in the show appletalk route display. All entries not marked with 0.0 or 0000.0000.0000 will be erased.

Syntax: clear appletalk route

Possible values: N/A

Default value: N/A

clear appletalk traffic

Erases all RTMP, ZIP, AEP, DDP, and AARP statistics for the Layer 3 Switch. You can display a summary of the statistics to be erased by entering the **show appletalk traffic** command.

EXAMPLE:

```
BigIron# clear appletalk traffic
```

Syntax: clear appletalk traffic

Possible values: N/A

Default value: N/A

clear arp

Removes all data from the ARP cache.

EXAMPLE:

```
BigIron# clear arp
```

Syntax: clear arp [ethernet <num> | mac-address <xxxx.xxxx.xxxx> [<mask>] | <ip-addr> [<ip-mask>]]

Specify the MAC address mask as “f”s and “0”s, where “f”s are significant bits. Specify IP address masks in standard decimal mask format (for example, 255.255.0.0).

The following command clears all ARP entries for port 2 on the module in slot 3.

```
BigIron# clear arp ethernet 3/2
```

Possible values: N/A

Default value: N/A

clear atm map

Clears learned ATM point-to-multipoint mappings and refreshes the interfaces that use Inverse ARP.

NOTE: This command applies only to ATM point-to-multipoint interfaces that use Inverse ARP. This command does not affect point-to-point interfaces or statically configured mappings.

EXAMPLE:

```
BigIron# clear atm map
```

Syntax: clear atm map

Possible values: N/A

Default value: N/A

clear dot1x statistics

Clears the 802.1X statistics counters. You can clear the 802.1X statistics counters on all interfaces at once, on individual interfaces, or on a range of interfaces.

EXAMPLE:

To clear the 802.1X statistics counters on all interfaces on the device:

```
BigIron# clear dot1x statistics all
```

To clear the 802.1X statistics counters on interface e 3/11:

```
BigIron# clear dot1x statistics e 3/11
```

Syntax: clear dot1x statistics all

Syntax: clear dot1x statistics <portnum>

Syntax: clear dot1x statistics <portnum> to <portnum>

Possible values: Port number

Default value: N/A

clear dvmrp cache

Erases all DVMRP forwarding entries.

EXAMPLE:

```
BigIron# clear dvmrp cache
```

Syntax: clear dvmrp cache

Possible values: N/A

Default value: N/A

clear dvmrp flow

Erases all information in the DVMRP flow cache, specifically source, group and forwarding index information.

EXAMPLE:

```
BigIron# clear dvmrp flow
```

Syntax: clear dvmrp

Possible values: N/A

Default value: N/A

clear dvmrp route

Erases all DVMRP routing information that DVMRP exchanges with its peers.

EXAMPLE:

```
BigIron# clear dvmrp route
```

Syntax: clear dvmrp

Possible values: N/A

Default value: N/A

clear fdp counters

Clears the counters for Foundry Discovery Protocol (FDP) and Cisco Discovery Protocol (CDP) statistics.

EXAMPLE:

```
BigIron# clear fdp counters
```

Syntax: clear fdp counters

Possible values: N/A

Default value: N/A

clear fdp table

Clears the Foundry and Cisco neighbor information gathered from Foundry Discovery Protocol (FDP) and Cisco Discovery Protocol (CDP) packets.

EXAMPLE:

```
BigIron# clear fdp table
```

Syntax: clear fdp table

Possible values: N/A

Default value: N/A

clear gvrp statistics

Clears the GVRP statistics counters.

EXAMPLE:

```
BigIron# clear gvrp statistics all
```

This command clears the counters for all ports. To clear the counters for a specific port only, enter a command such as the following:

```
BigIron# clear gvrp statistics ethernet 2/1
```

Syntax: clear gvrp statistics all | ethernet <portnum>

Possible values: See above

Default value: N/A

clear ip acl-traffic

Clears ACL statistics for IronCore flow-based (CPU-based) ACLs.

EXAMPLE:

```
BigIron(config)# clear ip acl-traffic
```

Syntax: clear ip acl-traffic

Possible values: N/A

Default value: N/A

clear ip bgp neighbor

Closes a neighbor session and flushes all the routes exchanged by the Layer 3 Switch and the neighbor. You also can reset a neighbor session without closing it by resending the BGP route table (soft-outbound option).

See the "Closing or Resetting a Session With Neighbors" section of the "Configuring BGP" chapter in the *Foundry Enterprise Configuration and Management Guide* for more information.

EXAMPLE:

To close all neighbor sessions, enter the following command.

```
BigIron# clear ip bgp neighbor
```

Closes a neighbor session and flushes all the routes exchanged by the Layer 3 Switch and the neighbor.

EXAMPLE:

To resend routes to a neighbor without closing the neighbor session, enter a command such as the following:

```
BigIron# clear ip bgp neighbor 10.0.0.1 soft-outbound
```

Syntax: clear ip bgp neighbor all | <ip-addr> | <peer-group-name> | <as-num> [soft-outbound | soft [in | out]]

Syntax: clear ip bgp neighbor all | <ip-addr> | <peer-group-addr> [last-packet-with-error | notification-errors | traffic]

Syntax: clear ip bgp neighbor <ip-addr> [soft in prefix-filter]

The **all** | <ip-addr> | <peer-group-name> parameter indicates whether you are clearing BGP4 information for all neighbors, for an individual neighbor, or for a peer group. If you specify a neighbor's IP address, you are clearing information for only that neighbor. If you specify a peer group name, you are clearing information for all the neighbors within that peer group.

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.
  - 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.

The **soft-outbound** option causes the device to compile 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.

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.

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**).

The **last-packet-with-error** option clears the buffer containing the first 400 bytes of the last BGP4 packet that contained an error.

The **notification-errors** option clears the buffer containing the last NOTIFICATION message sent or received.

The **traffic** option clears the BGP4 message counter for the specified neighbor.

The **soft in prefix-filter** parameter sends an updated IP prefix list as an Outbound Route Filter (ORF) to the neighbor as part of its route refresh message to the neighbor. This parameter applies to the cooperative route filtering feature.

NOTE: If the Layer 3 Switch or the neighbor is not configured for cooperative filtering, the command sends a normal route refresh message.

Possible values: See above

Default value: N/A

clear ip bgp routes

Clears BGP4 routes from the IP route table and resets the routes.

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.

EXAMPLE:

```
BigIron# clear ip bgp routes
```

Syntax: clear ip bgp routes [<ip-addr>/<prefix-length>]

Possible values: See above

Default value: N/A

clear ip bgp traffic

Clears the BGP4 message counter for all neighbors.

EXAMPLE:

```
BigIron# clear ip bgp traffic
```

Syntax: clear ip bgp traffic

Possible values: N/A

Default value: N/A

clear ip cache

Removes all entries from the IP cache.

EXAMPLE:

```
BigIron# cl ip cache
```

Syntax: clear ip cache [ipaddr]

Possible values: N/A

Default value: N/A

clear ip dr-aggregate

Clears the default-route cache entries.

EXAMPLE:

```
BigIron# clear ip dr-aggregate
```

Syntax: clear ip dr-aggregate

Possible values: N/A

Default value: N/A

clear ip msdp peer

Clears MSDP peer information.

EXAMPLE:

```
BigIron# clear ip msdp peer 205.216.162.1
Remote connection closed
```

Syntax: clear ip msdp peer <ip-addr>

The command in this example clears the MSDP peer connection with MSDP router 205.216.162.1. The CLI displays a message to indicate when the connection has been successfully closed.

Possible values: N/A

Default value: N/A

clear ip msdp sa-cache

Clears entries from the MSDP Source Active cache.

EXAMPLE:

```
BigIron# clear ip msdp sa-cache
```

Syntax: clear ip msdp sa-cache [<source-addr> | <group-addr>]

The command in this example clears all the cache entries. Use the <source-addr> parameter to clear only the entries for a specified course. Use the <group-addr> parameter to clear only the entries for a specific group.

Possible values: N/A

Default value: N/A

clear ip msdp statistics

Clears MSDP statistics.

EXAMPLE:

```
BigIron# clear ip msdp statistics
```

Syntax: clear ip msdp statistics [<ip-addr>]

The command in this example clears statistics for all the peers. To clear statistics for only a specific peer, enter the peer's IP address.

Possible values: N/A

Default value: N/A

clear ip multicast

Clears learned IP multicast group reports on a Layer 2 Switch. This command applies to the IP multicast traffic reduction feature.

EXAMPLE:

To clear all the IGMP reports learned by the Layer 2 Switch, enter the following command at the Privileged EXEC level of the CLI:


```
FastIronII# clear ip multicast all
```

The following example shows IGMP report information listed by the **show ip multicast** command, followed by removal of the information by the **clear ip multicast all** command.

```
FastIronII# 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

FastIronII# clear ip multicast all

FastIronII# 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 reports for a specific IP multicast group, enter a command such as the following:

```
FastIronII# clear ip multicast group 239.255.162.5
```

The following example shows how to clear the IGMP reports for a specific group and retain reports for other groups.

```
FastIronII# 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

FastIronII# clear ip multicast group 239.255.162.5

FastIronII# 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 reports for all groups.

The **group** <group-id> parameter clears the reports for the specified group but does not clear the reports for other groups.

Possible values: N/A

Default value: N/A

clear ip nat

Clears entries from the Network Address Translation (NAT) table.

EXAMPLE:

To clear all dynamic entries from the NAT translation table, enter the following command:

```
BigIron# clear ip nat all
```

Syntax: clear ip nat all

To clear only the entries for a specific address entry, enter a command such as the following:

```
BigIron# clear ip nat inside 209.157.1.43 10.10.10.5
```


This command clears the inside NAT entry that maps private address 10.10.10.5 to Internet address 209.157.1.43. Here is the syntax for this form of the command.

Syntax: clear ip nat inside <global-ip> <private-ip>

If you use Port Address Translation, you can selectively clear entries based on the TCP or UDP port number assigned to an entry by the feature. For example, the following command clears one of the entries associated with Internet address 209.157.1.44 but does not clear other entries associated with the same address.

```
BigIron# clear ip nat inside 209.157.1.43 1081 10.10.10.5 80
```

The command above clears all inside NAT entries that match the specified global IP address, private IP address, and TCP or UDP ports.

Syntax: clear ip nat <protocol> inside <global-ip> <internet-tcp/udp-port> <private-ip> <private-tcp/udp-port>

The <protocol> parameter specifies the protocol type and can be tcp or udp.

Possible values: See above

Default value: N/A

clear ip route

Clears all IP routes from memory.

EXAMPLE:

```
BigIron# cl ip ro
```

Syntax: clear ip route [<ip-addr> <ip-mask>]

Possible values: The <ip-addr> <ip-mask> option clears the specified route from the IP route table, while leaving other routes in the table.

Default value: N/A

clear ip vrrp-stat

Clears VRRP or VRRPE statistics.

EXAMPLE:

```
BigIron# cl ip vrrp
```

Syntax: clear ip vrrp-stat

Possible values: N/A

Default value: N/A

clear ipx cache

Clears all entries in the IPX cache.

EXAMPLE:

```
BigIron# cl ipx ca
```

Syntax: clear ipx cache

Possible values: N/A

Default value: N/A

clear ipx route

Clears all IPX routes and servers from memory.

EXAMPLE:

```
BigIron# cl ipx rou
```

Syntax: clear ipx route

Possible values: N/A

Default value: N/A

clear isis

Clears IS-IS information.

EXAMPLE:

```
NetIron# clear isis all
```

This command clears all the following:

- Neighbors (closes the Layer 3 Switch's adjacencies with its IS-IS neighbors)
- Routes
- PDU statistics
- Error statistics

Syntax: clear isis all | counts | neighbor | route | traffic

The **all** parameter clears all the IS-IS information. Using this option is equivalent to entering separate commands with each of the other options.

The **counts** parameter clears the error statistics.

The **neighbor** parameter closes the Layer 3 Switch's adjacencies with its IS-IS neighbors and clears the neighbor statistics.

The **route** parameter clears the IS-IS route table.

The **traffic** parameter clears the PDU statistics.

NOTE: The **traffic** option also clears the values displayed in the **show isis interface** command's Control Messages Sent and Control Messages Received fields.

Possible values: N/A

Default value: N/A

clear link-aggregate

Clears the 802.3ad link-aggregation information negotiated using LACP.

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.

EXAMPLE:

```
BigIron# clear link-aggregate
```

Syntax: clear link-aggregate

Possible values: N/A

Default value: N/A

clear link-keepalive statistics

Clears Uni-Directional Link Detection (UDLD) statistics.

EXAMPLE:

```
BigIron# clear link-keepalive statistics
```

Syntax: clear link-keepalive statistics

Possible values: N/A

Default value: N/A

clear logging

Removes all entries from the SNMP event log.

EXAMPLE:

```
BigIron# cl logging
```

Syntax: clear logging

Possible values: N/A

Default value: N/A

clear mac vpls

Clears the entries stored in the VPLS MAC database belonging to a VPLS instance.

EXAMPLE:

```
NetIron# clear mac vpls name v1
```

Syntax: clear mac vpls name <name> | id <vpls-vcid> | ethernet <portnum> | label <label>

Possible values:

The **name** <name> parameter clears all entries associated with the named VPLS instance.

The **id** <vpls-vcid> parameter clears all entries associated with the specified VPLS ID.

The **ethernet** <portnum> parameter clears all entries on the specified port.

The **label** <label> parameter clears all entries associated with a local label

Default value: N/A

clear mac-address

Removes learned MAC address entries from the MAC address table.

EXAMPLE:

```
BigIron# clear mac-address ethernet 1/1
```

This command clears the learned MAC addresses for port 1/1. MAC addresses for other ports are not affected.

The following command clears all learned MAC addresses:

```
BigIron# clear mac-address
```

Syntax: clear mac-address [ethernet | pos <portnum>] | [vlan <vlan-id>] | [module <slotnum>]

Possible values: See above.

Default value: N/A

clear mpls rsvp statistics

Clears the RSVP statistics counters.

EXAMPLE:

```
BigIron# clear mpls rsvp
```

Syntax: clear mpls rsvp

Possible values: N/A

Default value: N/A

clear mpls statistics

Clears the MPLS statistics counters.

EXAMPLE:

```
BigIron# clear mpls statistics
```

Syntax: clear mpls statistics

Possible values: N/A

Default value: N/A

clear pim cache

Erases all forwarding entries from the PIM cache.

EXAMPLE:

```
BigIron# cl pim cache
```

Syntax: clear pim cache

Possible values: N/A

Default value: N/A

clear pim rp-map

A new command in 07.6.03, 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 the **rp-address** command.

EXAMPLE:

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:

```
BigIron(config)# clear pim rp-map
```

Syntax: clear pim rp-map

Possible values: N/A

Default value: N/A

clear public-key

Clears the public keys from the active configuration.

EXAMPLE:

```
BigIron# clear public-key
```

Syntax: clear public-key

Possible values: N/A

Default value: N/A

clear statistics

Resets statistics counters to zero. You can clear all statistics (the default) or rate counters only. In addition, you can clear statistics for all slots and ports (the default) or specify particular slots or ports.

EXAMPLE:

```
BigIron# clear statistics
```

Syntax: clear statistics [ethernet <portnum>]

Syntax: clear statistics [slot <slot-num>]

Syntax: clear statistics [rate-counters [ethernet <portnum> | slot <slot-num>]]

Possible values: N/A

Default value: N/A

clear statistics dos-attack

Resets counters for ICMP and TCP SYN packet burst thresholds.

EXAMPLE:

```
BigIron# clear statistics dos-attack
```

Syntax: clear statistics dos-attack

Possible values: N/A

Default value: N/A

clear statistics rate-counters

Clears Adaptive Rate Limiting statistics or Denial of Service protection statistics.

EXAMPLE:

```
BigIron# clear statistics rate-counters ethernet 1/1
```

This command clears the Adaptive Rate Limiting statistics that have been accumulated for port 1/1.

Syntax: clear statistics rate-counters [dos-attack | ethernet <portnum> | pos <portnum> | slot <slotnum>]

The **dos-attack** parameter clears statistics about ICMP and TCP SYN packets dropped because burst thresholds were exceeded. See the “Protecting Against Denial of Service Attacks” chapter in the *Foundry Security Guide*.

The other parameters clear Adaptive Rate Limiting statistics for the specified interface or module.

Possible values: N/A

Default value: N/A

clear web-connection

Clears all Web management interface sessions with the device. The sessions are immediately ended when you enter the command.

EXAMPLE:

```
BigIron# cl web
```

Syntax: clear web-connection

Possible values: N/A

Default value: N/A

clock

The system clock can be set for a Layer 2 Switch or Layer 3 Switch. This command allows you to set the time and date. The time zone must be set using the **clock timezone**... command at the global CONFIG level.

NOTE: Clock settings are not saved over power cycles; however, you can configure the system to reference an SNTP server at power up. This server will then automatically download the correct time reference for the network. For more details on this capability, reference the **sntp** command at the privileged EXEC level and the **sntp poll-interval** and **sntp server** commands at the global CONFIG level.

EXAMPLE:

```
BigIron# clock set 10:15:05 10-15-98
```

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

Possible values: N/A

Default value: N/A

configure terminal

Launches you into the global CONFIG level.

EXAMPLE:

```
BigIron# conf term
BigIron(config)#
```

Syntax: configure terminal

Possible values: N/A

Default value: N/A

copy <from-card> <to-card>

Copies files from one PCMCIA flash card on a management module to the other card.

NOTE: This command does the same thing as the **ncopy slot1** | **slot2** <from-name> **slot1** | **slot2** <to-name> command. See "ncopy slot1 | slot2 <from-name> slot1 | slot2 [<to-name>]" on page 5-36.

EXAMPLE:

To copy a file from one flash card to the other, enter the following command:

```
BigIron# copy slot1 slot2 sales.cfg
```

Syntax: copy <from-card> <to-card> [\<from-dir-path>\]<from-name> [[\<to-dir-path>\]<to-name>]

The command shown in the example above copies a file from the flash card in slot 1 to the flash card in slot 2. In this case, the software uses the same name for the original file and for the copy. Optionally, you can specify a different file name for the copy.

Possible values: See above.

Default value: N/A

copy flash flash

Copies a software image between the primary and secondary flash storage locations.

EXAMPLE:

Suppose you want to copy the software image stored in the primary flash into the secondary storage location. To do so, enter the following command.

```
BigIron# copy flash flash secondary
```

If you want to copy the image from the secondary flash to the primary flash, enter the following command.

```
BigIron# copy flash flash primary
```

In the **copy flash flash**…command, the first '**flash**' refers to the origin of the image and the second '**flash**' in the command points to the destination flash. Note that in the command above, when '**primary**' is entered, the system automatically knows that the origin flash is the secondary flash location.

Syntax: copy flash flash [primary | secondary]

Possible values: N/A

Default value: N/A

copy flash slot1 | slot2

Copies a file from flash memory to a PCMCIA flash card on the management module.

NOTE: This command does the same thing as the **ncopy flash primary** | **secondary slot1** | **slot2** <to-name> command. See “ncopy flash primary | secondary slot1 | slot2 <to-name>” on page 5-34.

EXAMPLE:

To copy a file from flash memory to a flash card, enter a command such as the following:

```
BigIron# copy flash slot2 BIS07000.bin primary
Flash Card Write (128 KBytes per dot) .......
Write to slot2 BIS07000.bin succeeded
```

The command in this example copies a software image file from the primary area in flash memory onto the flash card in slot 2.

If the copy does not succeed, the software lists messages to indicate the reason the copy did not work. For example, the following messages indicate that the copy did not work because the slot specified for the copy does not contain a flash card.

```
BigIron# copy flash slot2 m4s.car secondary
The system can not find the drive specified
Write to slot2 m4s.car failed
```

Syntax: copy flash slot1 | slot2 [\<to-dir-path>\]<to-name> primary | secondary

Possible values: See above.

Default value: N/A

copy flash tftp

Uploads a copy of the primary or secondary software image to a TFTP server.

NOTE: This command does the same thing as the **ncopy flash primary** | **secondary tftp** <ip-addr> <from-name> command. See “ncopy flash primary | secondary tftp <ip-addr> <from-name>” on page 5-34.

EXAMPLE:

```
BigIron# copy flash tftp 192.22.33.4 test.img secondary
```

Syntax: copy flash tftp <ip-addr> <filename> primary | secondary

Possible values: See above.

Default value: N/A

copy running slot1 | slot2

Copies the device’s running-config to a PCMCIA flash card. The running-config contains the device’s currently active configuration information. When you copy the running-config to a flash card, you are making a copy of the device’s current configuration, including any configuration changes you have not saved to the startup-config file.

NOTE: This command does the same thing as the **ncopy running slot1** | **slot2** <to-name> command. See “ncopy running slot1 | slot2 <to-name>” on page 5-34.

EXAMPLE:

To copy the device’s running configuration into a file on a flash card, enter a command such as the following:

```
BigIron# copy running slot1 runip.1
Write to slot1 run.sw succeeded
```

Syntax: copy running slot1 | slot2 [\<to-dir-path>\]<to-name>

Possible values: See above.

Default value: N/A

copy running-config tftp

Uploads a copy of the running configuration file from the Layer 2 Switch or Layer 3 Switch to a designated TFTP server.

NOTE: This command does the same thing as the **ncopy running-config tftp** <ip-addr> <from-name> command. See "ncopy running-config tftp <ip-addr> <from-name>" on page 5-35.

EXAMPLE:

```
BigIron# copy running-config tftp 192.22.3.44 newrun.cfg
```

Syntax: copy running-config tftp <ip-addr> <filename>

Possible values: See above.

Default value: N/A

copy slot1 | slot2 flash

Copies a file from a PCMCIA flash card to the primary area in flash memory.

NOTE: This command does the same thing as the **ncopy slot1** | **slot2** <from-name> **flash primary** | **secondary** command. See "ncopy slot1 | slot2 <from-name> flash primary | secondary" on page 5-35.

EXAMPLE:

To copy a file from a flash card to the primary area in flash memory, enter a command such as the following:

```
BigIron# copy slot1 flash B2P07000.bin primary
BigIron# Flash Erase ---------------------------------------
Flash Memory Write (8192 bytes per dot) .....................
.............................................................
......code flash copy done
```

Syntax: copy slot1 | slot2 flash [\<from-dir-path>\]<from-name> primary | secondary

Possible values: See above.

Default value: N/A

copy slot1 | slot2 running

Loads ACLs from a running-config file into the device's active configuration.

For example, if the device's configuration includes a large set of Access Control Lists (ACLs), you can configure the ACLs offline in a text file on a PC, then save the file to the flash card. To load the ACLs, you can insert the flash card in the Foundry device, then copy the file to the device's running configuration.

NOTE: This feature allows you to preconfigure and load large sets of ACLs. If you accidentally try to load a running-config file that contains other types of configuration information using this method, the software might display error messages. This occurs when the device's parser encounters lines in the file that do not correspond to valid configuration commands.

NOTE: This command does the same thing as the **ncopy slot1** | **slot2** <from-name> running command. See "ncopy slot1 | slot2 <from-name> running" on page 5-35.

EXAMPLE:

To copy a running-config file from a flash card, enter a command such as the following:

```
BigIron# copy slot2 running runip.2
```

Syntax: copy slot1 | slot2 running [\<from-dir-path>\]<from-name>

The command in this example changes the device's active configuration based on the information in the file.

Possible values: See above.

Default value: N/A

copy slot1 | slot2 start

Copies a startup-config file from a PCMCIA flash card to flash memory. By default, the device uses the startup-config in the primary area of flash memory to configure itself when you boot or reload the device.

The device cannot use a startup-config file on a flash card to configure itself. You cannot boot or reload from a flash card.

NOTE: This command does the same thing as the **ncopy slot1** | **slot2** <from-name> start command. See "ncopy slot1 | slot2 <from-name> start" on page 5-36.

EXAMPLE:

To copy a startup-config file from a flash card to flash memory, enter a command such as the following:

```
BigIron# copy slot1 start test2.cfg
..Write startup-config done.
```

Syntax: copy slot1 | slot2 start [\<from-dir-path>\]<from-name>

This command copies a configuration file named test2.cfg from the flash card in slot 2 into the device's flash memory. The next time you reboot or reload the device, it uses the configuration information in test2.cfg.

Possible values: See above.

Default value: N/A

copy slot1 | slot2 tftp

Copies a file from a PCMCIA flash card to a TFTP server.

NOTE: This command does the same thing as the **ncopy slot1** | **slot2** <from-name> **tftp** <ip-addr> [<to-name>] command. See "ncopy slot1 | slot2 <from-name> tftp <ip-addr> [<to-name>]" on page 5-36.

EXAMPLE:

To copy a file from a flash card to a TFTP server, enter a command such as the following:

```
BigIron# copy slot1 tftp 192.168.1.17 notes.txt
Uploading 254 bytes to tftp server ...
Upload to TFTP server done.
```

Syntax: copy slot1 | slot2 tftp <ip-addr> [\<from-dir-path>\]<from-name> [<to-name>]

Possible values: See above.

Default value: N/A

copy start slot1 | slot2

Copies the device's startup-config file from flash memory onto a PCMCIA flash card.

NOTE: This command does the same thing as the **ncopy start slot1** | **slot2** <to-name> command. See "ncopy start slot1 | slot2 <to-name>" on page 5-37.

EXAMPLE:

To copy the device's startup-config file from flash memory onto a flash card, enter a command such as the following:

```
BigIron# copy start slot1 mfgtest.cfg
Write to slot1 cfgtest.cfg succeeded
```


Syntax: copy start slot1 | slot2 [\<to-dir-path>\]<to-name>

Possible values: See above.

Default value: N/A

copy startup-config tftp

Uploads a copy of the startup configuration file from the Layer 2 Switch or Layer 3 Switch to a TFTP server.

NOTE: This command does the same thing as the **ncopy startup-config tftp** <ip-addr> <from-name> command. See "ncopy startup-config tftp <ip-addr> <from-name>" on page 5-37.

EXAMPLE:

```
BigIron# copy startup-config tftp 192.22.3.44 new.cfg
```

Syntax: copy startup-config tftp <ip-addr> <filename>

Possible values: See above.

Default value: N/A

copy tftp flash

Downloads a copy of a Foundry software image from a TFTP server into the system flash in the primary or secondary storage location.

NOTE: This command does the same thing as the **ncopy tftp** <ip-addr> <from-name> **flash primary** | **secondary** command. See "ncopy tftp <ip-addr> <from-name> flash primary | secondary" on page 5-37.

EXAMPLE:

```
BigIron# copy tftp flash 192.22.33.4 test.img primary
```

To download into the secondary storage location, enter the command listed below instead:

```
BigIron# copy tftp flash 192.22.33.4 test.img secondary
```

Syntax: copy tftp flash <ip-addr> <filename> primary | secondary

Possible values: See above.

Default value: N/A

copy tftp running-config

Downloads a copy of a running-config from a TFTP server into the running-config of a Foundry device.

NOTE: This command does the same thing as the **ncopy tftp** <ip-addr> <from-name> **running-config** command. See "ncopy tftp <ip-addr> <from-name> running-config" on page 5-38.

EXAMPLE:

```
BigIron# copy tftp running-config 192.22.33.4 newrun.cfg
```

Syntax: copy tftp running-config <ip-addr> <filename>

Possible values: See above.

Default value: N/A

copy tftp slot1 | slot2

Copies a file from a TFTP server to a PCMCIA flash card.

NOTE: This command does the same thing as the **ncopy tftp** <ip-addr> <from-name> **slot1** | **slot2** [<to-name>] command. See "ncopy tftp <ip-addr> <from-name> slot1 | slot2 [<to-name>]" on page 5-38.

EXAMPLE:

To copy a file from a TFTP server to a flash card, enter a command such as the following:

```
BigIron# copy tftp slot1 192.168.1.17 notes.txt
Downloading from tftp server ...
Tftp 254 bytes done, copy to slot1 ...
Write to slot1 cfg.cfg succeeded
```

Syntax: copy tftp slot1 | slot2 <ip-addr> <from-name> [[\<to-dir-path>\]<to-name>]

If the file name you specify is not on the TFTP server, the CLI displays messages such as those shown in the following example:

```
BigIron# copy tftp slot1 192.168.1.17 nots.txt
Downloading from tftp server ...
TFTP: received error request -- code 1 message File not found: C:/TFTP/nots.txt.
Error - can't download data from TFTP server, error code 17. Abort!
```

To simplify troubleshooting, especially when the file is present on your server but the command doesn't find it, the messages list the complete TFTP path name on your TFTP server.

Possible values: See above.

Default value: N/A

copy tftp startup-config

Downloads a copy of a configuration file from a TFTP server into the startup configuration file of the Layer 2 Switch or Layer 3 Switch. To activate this configuration file, reload (reset) the system.

NOTE: This command does the same thing as the **ncopy tftp** <ip-addr> <from-name> **startup-config** command. See "ncopy tftp <ip-addr> <from-name> startup-config" on page 5-38.

EXAMPLE:

```
BigIron# copy tftp startup-config 192.22.33.4 new.cfg
```

Syntax: copy tftp startup-config <ip-addr> <filename>

Possible values: See above.

Default value: N/A

delete

Deletes a file from a flash card. This command applies only to management modules with PCMCIA slots.

NOTE: By default, the delete option deletes all files on the flash card. Make sure you specify the files you want to delete.

NOTE: The software does not have an undelete option. Make sure you really want to delete the file.

EXAMPLE:

To delete a file on the flash card that has the management focus, enter a command such as the following:

```
BigIron# delete cfg.cfg
```

If the command is successful, the CLI displays a new command prompt.

Syntax: delete [slot1 | slot2] [<file-name>]

The command in this example deletes the specified file. To delete all files that contain a specific string of characters, enter a command such as the following:

```
BigIron# delete test*.*
```


This command deletes all files whose names start with “test”. To delete all the files on a flash card, enter a command such as the following:

```
BigIron# delete slot2
```

The command in this example deletes all files on the flash card in slot 2. In this example, slot 1 has the management focus, but the files to be deleted are on the flash card in slot 2.

Possible values: See above.

Default value: Deletes all files on the flash card!

dir

List the files on a flash card in a Management IV module’s PCMCIA slot.

NOTE: By default, the software displays the contents of the flash card in the slot that has the management focus. However, you do not need to change the focus to list the files on another flash card. You can specify the other flash card when you display the files.

EXAMPLE:

To display a directory of all the files on the flash card that has the management focus, enter the following command:

```
BigIron# dir
 Volume in slot1 has no label
 Volume Serial Number is 19ED-1725

 Directory of slot1

01/01/2000  00:00a    685935     POS.BIN
01/01/2000  00:00a   2157693     M4R.BIN
01/01/2000  00:00a       184     A22.CFG
01/01/2000  00:00a       254   R CFG.CFG
01/01/2000  00:00a       256     STR.CFG
01/01/2000  00:00a   1027230     M5.BIN
01/01/2000  00:00a       184     A8.CFG
01/01/2000  00:00a   1029838     M4S.BIN
01/01/2000  00:00a    687026     P3R.BIN
01/01/2000  00:00a   1029838     MM.BIN
              10 File(s)        6618438 bytes
                               74180608 bytes free
```

Syntax: dir [slot1 | slot2] [<file-name>]

To list only files that contain a specific pattern of characters in the name, enter a command such as the following:

```
BigIron# dir *.bin
 Volume in slot1 has no label
 Volume Serial Number is 19ED-1725

 Directory of slot1

01/01/2000  00:00a    685935     POS.BIN
01/01/2000  00:00a   2157693     M4R.BIN
01/01/2000  00:00a   1027230     M5.BIN
01/01/2000  00:00a   1029838     M4S.BIN
01/01/2000  00:00a    687026     P3R.BIN
01/01/2000  00:00a   1029838     MM.BIN
               6 File(s)        6617560 bytes
                               74180608 bytes free
```

The command in this example lists all the image files on the flash card in the slot that has the management focus. (More specifically, the command lists all the files that end with “.bin”.)

For information about the command's display, see the "Displaying a Directory of the Files on a Flash Card" section in the "Using Redundant Management Modules" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Possible values: See above.

Default value: Displays all files on the flash card that has the management focus.

disable

Disables a forwarding module to prepare it for removal from a Chassis device.

When you remove a module from a Chassis device, disable the module first before removing it from the chassis. Disabling the module before removing it prevents a brief service interruption on other forwarding modules. The brief interruption can be caused by the Chassis device reinitializing other modules in the chassis when you remove an enabled module.

NOTE: This section does not apply to the active or standby management modules. The **disable module** and **enable module** commands are not applicable to management modules.

EXAMPLE:

```
BigIron# disable module 3
```

This command disables the module in slot 3.

Syntax: disable module <slot-num>

The <slot-num> parameter specifies the slot number.

- Slots in a 4-slot chassis are numbered 1 – 4, from top to bottom.
- Slots in an 8-slot chassis are numbered 1 – 8, from left to right.
- Slots in a 15-slot chassis are numbered 1 – 15, from left to right.

NOTE: If you remove the module without first disabling it, the chassis re-initializes the other modules in the chassis, causing a brief interruption in service after which the chassis resumes normal operation.

If you decide after disabling a module that you do not want to remove the module, re-enable the module using the following command:

```
BigIron# enable module 3
```

Syntax: enable module <slot-num>

NOTE: You do not need to enable a module after inserting it in the chassis. The module is automatically enabled when you insert the module into a live chassis or when you power on the chassis.

NOTE: On all Chassis devices, if you plan to replace the removed module with a different type of module, you must configure the slot for the module. To configure a slot for a module, use the **module** command at the global CONFIG level of the CLI.

Possible values: See above

Default value: Enabled

dot1x initialize

Initializes 802.1X port security on a port.

EXAMPLE:

```
BigIron# dot1x initialize e 3/1
```

Syntax: dot1x initialize <portnum>

Possible values: Port number

Default value: N/A

dot1x re-authenticate

Manually re-authenticates 802.1X-enabled Clients connected to a specified port.

EXAMPLE:

```
BigIron# dot1x re-authenticate e 3/1
```

Syntax: dot1x re-authenticate <portnum>

Possible values: 802.1X-enabled port

Default value: N/A

enable

Re-enables a forwarding module. See “disable” on page 5-26.

NOTE: The **disable module** and **enable module** commands are not applicable to management modules.

erase flash primary

Erases the image stored in primary flash.

EXAMPLE:

```
BigIron# er f pri
```

Syntax: erase flash primary

Possible values: N/A

Default value: N/A

erase flash secondary

Erases the image stored in secondary flash.

EXAMPLE:

```
BigIron# er f sec
```

Syntax: erase flash secondary

Possible values: N/A

Default value: N/A

erase startup-config

Erases the configuration stored in the startup-config file.

EXAMPLE:

```
BigIron# er start
```

Syntax: erase startup-config

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the user EXEC level.

EXAMPLE:

To move from the privileged EXEC level back to the user EXEC level, enter the following:

```
BigIron# exit
```



```
BigIron>
```

Syntax: exit

Possible values: N/A

Default value: N/A

ext clear mac-vlan

Clears entries that have been added to the MAC table from a MAC VLAN list. See the "Configuring Virtual LANs (VLANs)" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

This command applies only to Stackable switches.

EXAMPLE:

To clear MAC VLAN entries from the MAC table, enter the following command:

```
FastIron# ext clear mac-vlan
```

Syntax: ext clear mac-vlan [mac <mac-addr> <mask>] [vlan < vlan-id>]

If you enter **ext clear mac-vlan** without any of the optional parameters, all the entries added from MAC VLAN lists are cleared.

The **mac** <mac-addr> <mask> parameter clears only the entries that match the specified address and mask.

The **vlan** <vlan-id> parameter clears only the entries that match the specified VLAN.

Possible values: see above

Default value: all MAC VLAN entries are cleared

ext refresh config-file

Updates the MAC table by loading the MAC addresses from a MAC VLAN list. This command enables you to load changes made to a MAC VLAN file without reloading the switch. This command adds the entries in the MAC VLAN list to the MAC table. Existing entries in the table are not cleared. See the "Configuring Virtual LANs (VLANs)" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide* for more information.

This command applies only to Stackable switches.

EXAMPLE:

To clear MAC VLAN entries from the MAC table, enter the following command:

```
FastIron# ext refresh 209.157.22.3 MAC-VLAN.TXT
```

Syntax: ext refresh config-file <ip-addr> <external-file-name>

The <ip-addr> parameter specifies the IP address of the TFTP server on which you placed the MAC VLAN file.

The <external-file-name> parameter specifies the name of the MAC VLAN file.

Possible values: see above

Default value: N/A

fastboot

Provides a configurable option to speed up the system startup time. By default, this option is turned off, providing a three-second pause to allow you to break into the boot prompt, if necessary. Use fastboot on to turn this option on and eliminate the three-second pause. To turn this feature off later, enter the command **fastboot off**. Fastboot changes will be saved automatically but will not become active until after a system reset.

To execute an immediate reload from the console of the boot code without a three-second delay, you can enter the **fast-reload** command.

EXAMPLE:

```
BigIron# fastboot on
```


Syntax: fastboot [on | off]

Possible values: on or off

Default value: N/A

fast-reload

Initiates an immediate fast boot. Fastboot requires a boot flash image version of 02.00.06 or later to be operational. You can use the CLI command **show flash** to check the boot image version number.

NOTE: The **fast-reload** command is a hidden command of the privileged level of the CLI.

EXAMPLE:

```
BigIron# fast-reload
```

Syntax: fast-reload

Possible values: N/A

Default value: Disabled

format

Reformats a flash card in a Management IV module's PCMCIA slot.

EXAMPLE:

To reformat a flash card, enter the following command:

```
BigIron# format slot2
Formatting Flash Card(256 clusters per dot) ..................................
..............................................................................
.....................................
Verifying Flash Card(256 clusters per dot)  ..................................
..............................................................................
.....................................

 80809984 bytes total card space.
 80809984 bytes available on card.

     2048 bytes in each allocation unit.
    39458 allocation units available on card.

Flash card format done
```

As shown in this example, the software formats the sector on the flash card, then verifies the formatting. In this example, the software did not find any bad sectors, so all the bytes on the card are available.

Syntax: format slot1 | slot2 [<label>]

The **slot1** | **slot2** parameter specifies the PCMCIA slot that contains the flash card you are formatting.

The <label> parameter specifies the label. You can specify up to 11 alphanumeric characters. You cannot use special characters or spaces.

Possible values: See above

Default value: N/A

hd

Displays the data in a file on a flash card in hexadecimal format. This command applies only to management modules with PCMCIA flash slots.

EXAMPLE:

To display the data in a file in hexadecimal format, enter a command such as the following:

```
BigIron# hd cfg.cfg
00000000: 76657220 30362e35 2e303054 35310a6d    ver 06.5.00T51 m
00000010: 6f64756c 65203120 62692d34 2d706f72    odule 1 bi-4-por
00000020: 742d6769 672d6d34 2d6d616e 6167656d    t-gig-m4-managem
00000030: 656e742d 6d6f6475 6c650a6d 6f64756c    ent-module modul
00000040: 65203220 62692d32 342d706f 72742d63    e 2 bi-24-port-c
00000050: 6f707065 722d6d6f 64756c65 0a6d6f64    opper-module mod
00000060: 756c6520 33206269 2d342d70 6f72742d    ule 3 bi-4-port-
00000070: 6769672d 6d342d6d 616e6167 656d656e    gig-m4-managemen
00000080: 742d6d6f 64756c65 0a6d6f64 756c6520    t-module module
00000090: 34206269 2d342d70 6f72742d 6769672d    4 bi-4-port-gig-
000000a0: 6d342d6d 616e6167 656d656e 742d6d6f    m4-management-mo
000000b0: 64756c65 0a210a21 0a210a21 0a6d3220    dule ! ! ! ! m2
000000c0: 61637469 76652d6d 616e6167 656d656e    active-managemen
000000d0: 7420330a 69702061 64647265 73732031    t 3 ip address 1
000000e0: 39322e31 36382e32 2e353820 3235352e    92.168.2.58 255.
000000f0: 3235352e 3235352e 300a656e             255.255.0 end
```

Syntax: hd [slot1 | slot2] <file-name>

Each row of hexadecimal output contains the following parts:

- The byte offset of the date that is displayed to the right of the offset
- A row of hexadecimal data
- The ASCII equivalent of the hexadecimal data shown in the row

Possible values: see above

Default value: N/A

kill

Terminates an active CLI session.

The **kill** command terminates the specified active CLI session and resets the CONFIG token. If the terminated session was a console, the console is sent back into User EXEC mode. If the terminated CLI session was a Telnet session, the Telnet connection is closed.

EXAMPLE:

```
BigIron# kill telnet 1
```

Syntax: kill console | telnet <session-id>

Possible values: see above

Default value: N/A

To display the active console and Telnet CLI sessions:

```
BigIron# show who
Console connections:
 established
Telnet connections:
 1 established, client ip address 209.157.22.63
 2 closed
 3 closed
 4 closed
 5 closed
```


Syntax: show who

The **show who** command lists the status of the Console connection and the session ID and status of the five possible Telnet connections. Once you know the session ID of a Telnet connection, you can terminate it with the **kill** command.

locate

Displays or changes the save location for the startup-config file.

EXAMPLE:

```
BigIron# locate startup-config
```

Syntax: locate startup-config

EXAMPLE:

By default, when you save configuration changes, the changes are saved to the startup-config file on the device's flash memory module. If you want to change the save location to a PCMCIA slot, enter a command such as the following:

```
BigIron# locate startup-config slot1 router1.cfg
BigIron# write memory
```

The first command in this example sets the device to save configuration changes to the file named "router1.cfg" in the flash card in PCMCIA slot 1. The second command saves the running-config to the router1.cfg file on the flash card in slot 1.

NOTE: In this example, after you save the configuration changes using the **write memory** command, the router1.cfg file will include the command that designates PCMCIA slot1 as the save location for configuration changes.

Syntax: locate startup-config [[slot1 | slot2] <file-name>]

You can specify a relative path name or full path name as part of the file name.

Possible values: See above

Default value: N/A

md

Another form of the **md** command. See "mkdir" on page 5-31.

mkdir

Creates a subdirectory on a PCMCIA flash card.

EXAMPLE:

```
BigIron# mkdir slot1 \TEST
```

To verify successful creation of the subdirectory, enter a command to change to the new subdirectory level:

```
BigIron# chdir \TEST
Current directory of slot1 is: \TEST
```

Syntax: md | mkdir [slot1 | slot2] <dir-name>

You can enter either **md** or **mkdir** for the command name.

The **slot1** | **slot2** parameter specifies a PCMCIA slot. If you do not specify a slot, the command applies to the slot that currently has the management focus.

The <dir-name> parameter specifies the subdirectory name. You can enter a name that contains any combination of the following characters. Do not enter a backslash " / " in front of the name.

- All upper and lowercase letters
- All digits

- Spaces
- Any of the following special characters:
  - $
  - %
  - '
  - -
  - _
  - @
  - ~
  - `
  - !
  - (
  -)
  - {
  - }
  - ^
  - #
  - &

You can use spaces in a file or subdirectory name if you enclose the name in double quotes. For example, to specify a subdirectory name that contains spaces, enter a string such as the following: “a long subdirectory name”.

A subdirectory or file name can be a maximum of 256 characters long. A complete subdirectory path name cannot contain more than 263 characters.

The name is not case sensitive. You can enter upper- or lowercase letters. The CLI displays the name using uppercase letters.

Possible values: See above

Default value: N/A

more

Displays the data in a file on a flash card in a Management IV module’s PCMCIA slot.

EXAMPLE:

To display the contents of a file, enter a command such as the following:

```
BigIron# more cfg.cfg
ver 06.5.00T51
module 1 bi-4-port-gig-m4-management-module
module 2 bi-24-port-copper-module
module 3 bi-4-port-gig-m4-management-module
module 4 bi-4-port-gig-m4-management-module
!
!
!
!
m2 active-management 3
ip address 192.168.2.58 255.255.255.0
end
```


Syntax: more [slot1 | slot2] <file-name>

This example shows the contents of a simple configuration file.

Possible values: See above.

Default value: N/A

mrinfo

Displays the PIM configuration of another PIM router.

EXAMPLE:

```
BigIron# 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]
```

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)

NOTE: This display shows the PIM interface configuration information, but does not show the link states for the interfaces.

Syntax: mrinfo <ip-addr>

Possible values: The <ip-addr> parameter specifies the IP address of the PIM router.

Default value: N/A

mtraceroute

Traces a PIM route.

EXAMPLE:

To trace a PIM route to PIM source 209.157.24.62 in group 239.255.162.1:

```
BigIron# mtraceroute 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  207.157.24.162
```

Syntax: mtraceroute source <ip-addr> group <multicast-group>

Possible values: 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.

Default value: N/A

ncopy flash primary | secondary slot1 | slot2 <to-name>

Copies a file from flash memory to a PCMCIA flash card on the management module.

NOTE: This command does the same thing as the **copy flash slot1** | **slot2** command. See "copy flash slot1 | slot2" on page 5-19.

EXAMPLE:

To copy a file from flash memory to a flash card, enter a command such as the following:

```
BigIron# ncopy flash primary slot2 BIS07000.bin
Flash Card Write (128 KBytes per dot) .......
Write to slot2 BIS07000.bin succeeded
```

The command in this example copies a software image file from the primary area in flash memory onto the flash card in slot 2.

If the copy does not succeed, the software lists messages to indicate the reason the copy did not work. For example, the following messages indicate that the copy did not work because the slot specified for the copy does not contain a flash card.

```
BigIron# ncopy flash secondary slot2 m4s.car
The system can not find the drive specified
Write to slot2 m4s.car failed
```

Syntax: ncopy flash primary | secondary slot1 | slot2 [\<to-dir-path>\]<to-name>

Possible values: See above.

Default value: N/A

ncopy flash primary | secondary tftp <ip-addr> <from-name>

Uploads a copy of the primary or secondary software image to a TFTP server.

NOTE: This command does the same thing as the **copy flash tftp** <ip-addr> <filename> **primary** | **secondary** command. See "copy flash tftp" on page 5-20.

EXAMPLE:

```
BigIron# ncopy flash secondary tftp 192.22.33.4 test.img
```

Syntax: ncopy flash primary | secondary tftp <ip-addr> <from-name>

Possible values: See above.

Default value: N/A

ncopy running slot1 | slot2 <to-name>

Copies the device's running-config to a PCMCIA flash card. The running-config contains the device's currently active configuration information. When you copy the running-config to a flash card, you are making a copy of the device's current configuration, including any configuration changes you have not saved to the startup-config file.

NOTE: This command does the same thing as the **copy running slot1** | **slot2** <to-name> command. See "copy running slot1 | slot2" on page 5-20.

EXAMPLE:

To copy the device's running configuration into a file on a flash card, enter a command such as the following:

```
BigIron# ncopy running slot1 runip.1
Write to slot1 run.sw succeeded
```

Syntax: ncopy running slot1 | slot2 [\<to-dir-path>\]<to-name>

Possible values: See above.

Default value: N/A

ncopy running-config tftp <ip-addr> <from-name>

Uploads a copy of the running configuration file from the Layer 2 Switch or Layer 3 Switch to a TFTP server.

NOTE: This command does the same thing as the **copy running-config tftp** <ip-addr> <filename> command. See "copy running-config tftp" on page 5-21.

EXAMPLE:

```
BigIron# ncopy running-config tftp 192.22.3.44 newrun.cfg
```

Syntax: ncopy running-config tftp <ip-addr> <from-name>

Possible values: See above.

Default value: N/A

ncopy slot1 | slot2 <from-name> flash primary | secondary

Copies a file from a PCMCIA flash card to the primary area in flash memory.

NOTE: This command does the same thing as the **copy slot1** | **slot2 flash** <from-name> **primary** | **secondary** command. See "copy flash slot1 | slot2" on page 5-19.

EXAMPLE:

To copy a file from a flash card to the primary area in flash memory, enter a command such as the following:

```
BigIron# ncopy slot1 B2P07000.bin flash primary
BigIron# Flash Erase ----------------------------------------
Flash Memory Write (8192 bytes per dot) ..................................
.........................................................................
......code flash copy done
```

Syntax: ncopy slot1 | slot2 [\<from-dir-path>\]<from-name> flash primary | secondary

Possible values: See above.

Default value: N/A

ncopy slot1 | slot2 <from-name> running

Loads ACLs from a running-config file into the device's active configuration.

For example, if the device's configuration includes a large set of Access Control Lists (ACLs), you can configure the ACLs offline in a text file on a PC, then save the file to the flash card. To load the ACLs, you can insert the flash card in the Foundry device, then copy the file to the device's running configuration.

NOTE: This feature allows you to preconfigure and load large sets of ACLs. If you accidentally try to load a running-config file that contains other types of configuration information using this method, the software might display error messages. This occurs when the device's parser encounters lines in the file that do not correspond to valid configuration commands.

NOTE: This command does the same thing as the **copy slot1** | **slot2 running** <from-name> command. See "copy slot1 | slot2 running" on page 5-21.

EXAMPLE:

To copy a running-config file from a flash card, enter a command such as the following:


```
BigIron# ncopy slot2 runip.2 running
```

Syntax: ncopy slot1 | slot2 [\<from-dir-path>\]<from-name> running

The command in this example changes the device's active configuration based on the information in the file.

Possible values: See above.

Default value: N/A

ncopy slot1 | slot2 <from-name> slot1 | slot2 [<to-name>]

Copies files from one PCMCIA flash card on a management module to the other card.

NOTE: This command does the same thing as the **copy** <from-card> <to-card> <from-name> [<to-name>] command. See "copy <from-card> <to-card>" on page 5-19.

EXAMPLE:

To copy a file from one flash card to the other, enter the following command:

```
BigIron# ncopy slot1 sales.cfg slot2
```

Syntax: ncopy slot1 | slot2 [\<from-dir-path>\]<from-name> slot1 | slot2 [[\<to-dir-path>\]<to-name>]

The command shown in the example above copies a file from the flash card in slot 1 to the flash card in slot 2. In this case, the software uses the same name for the original file and for the copy. Optionally, you can specify a different file name for the copy.

Possible values: See above.

Default value: N/A

ncopy slot1 | slot2 <from-name> start

Copies a startup-config file from a PCMCIA flash card to flash memory. By default, the device uses the startup-config in the primary area of flash memory to configure itself when you boot or reload the device.

NOTE: The device cannot use a startup-config file on a flash card to configure itself. You cannot boot or reload from a flash card.

NOTE: This command does the same thing as the **copy slot1** | **slot2 start** <from-name> command. See "copy slot1 | slot2 start" on page 5-22.

EXAMPLE:

To copy a startup-config file from a flash card to flash memory, enter a command such as the following:

```
BigIron# ncopy slot1 test2.cfg start
..Write startup-config done.
```

Syntax: ncopy slot1 | slot2 [\<from-dir-path>\]<from-name> start

This command copies a configuration file named test2.cfg from the flash card in slot 2 into the device's flash memory. The next time you reboot or reload the device, it uses the configuration information in test2.cfg.

Possible values: See above.

Default value: N/A

ncopy slot1 | slot2 <from-name> tftp <ip-addr> [<to-name>]

Copies a file from a PCMCIA flash card to a TFTP server.

NOTE: This command does the same thing as the **copy slot1** | **slot2 tftp** <ip-addr> <from-name> [<to-name>] command. See "copy slot1 | slot2 tftp" on page 5-22.

EXAMPLE:

To copy a file from a flash card to a TFTP server, enter a command such as the following:

```
BigIron# ncopy slot1 notes.txt tftp 192.168.1.17
Uploading 254 bytes to tftp server ...
Upload to TFTP server done.
```

Syntax: ncopy slot1 | slot2 [\<from-dir-path>\]<from-name> tftp <ip-addr> [<to-name>]

Possible values: See above.

Default value: N/A

ncopy start slot1 | slot2 <to-name>

Copies the device's startup-config file from flash memory onto a PCMCIA flash card.

NOTE: This command does the same thing as the **copy start slot1** | **slot2** <to-name> command. See "copy start slot1 | slot2" on page 5-22.

EXAMPLE:

To copy the device's startup-config file from flash memory onto a flash card, enter a command such as the following:

```
BigIron# ncopy start slot1 mfgtest.cfg
Write to slot1 cfgtest.cfg succeeded
```

Syntax: ncopy start slot1 | slot2 [\<to-dir-path>\]<to-name>

Possible values: See above.

Default value: N/A

ncopy startup-config tftp <ip-addr> <from-name>

Uploads a copy of the startup configuration file from the Layer 2 Switch or Layer 3 Switch to a TFTP server.

NOTE: This command does the same thing as the **copy startup-config tftp** <ip-addr> <filename> command. See "copy startup-config tftp" on page 5-23.

EXAMPLE:

```
BigIron# ncopy startup-config tftp 192.22.3.44 new.cfg
```

Syntax: ncopy startup-config tftp <ip-addr> <from-name>

Possible values: See above.

Default value: N/A

ncopy tftp <ip-addr> <from-name> flash primary | secondary

Downloads a copy of a Foundry Layer 2 Switch or Layer 3 Switch software image from a TFTP server into the system flash in the primary or secondary storage location.

NOTE: This command does the same thing as the **copy tftp flash** <ip-addr> <filename> **primary** | **secondary** command. See "copy tftp flash" on page 5-23.

EXAMPLE:

```
BigIron# ncopy tftp 192.22.33.4 test.img flash primary
```

To download into the secondary storage location, enter the command listed below instead:

```
BigIron# ncopy tftp 192.22.33.4 test.img flash secondary
```

Syntax: ncopy tftp <ip-addr> <from-name> flash primary | secondary

Possible values: See above.

Default value: N/A

ncopy tftp <ip-addr> <from-name> running-config

Downloads a copy of a running-config file from a TFTP server into the running-config of the Layer 2 Switch or Layer 3 Switch.

NOTE: This command does the same thing as the **copy tftp running-config** <ip-addr> <filename> command. See "copy tftp running-config" on page 5-23.

EXAMPLE:

```
BigIron# ncopy tftp 192.22.33.4 newrun.cfg running-config
```

Syntax: ncopy tftp <ip-addr> <from-name> running-config

Possible values: See above.

Default value: N/A

ncopy tftp <ip-addr> <from-name> slot1 | slot2 [<to-name>]

Copies a file from a TFTP server to a PCMCIA flash card.

NOTE: This command does the same thing as the **copy tftp slot1** | **slot2** <ip-addr> <from-name> [<to-name>] command. See "copy tftp slot1 | slot2" on page 5-23.

EXAMPLE:

To copy a file from a TFTP server to a flash card, enter a command such as the following:

```
BigIron# ncopy tftp 192.168.1.17 notes.txt slot1
Downloading from tftp server ...
Tftp 254 bytes done, copy to slot1 ...
Write to slot1 cfg.cfg succeeded
```

Syntax: ncopy tftp <ip-addr> <from-name> slot1 | slot2 [[\<to-dir-path>\]<to-name>]

If the file name you specify is not on the TFTP server, the CLI displays messages such as those shown in the following example:

```
BigIron# ncopy tftp 192.168.1.17 nots.txt slot1
Downloading from tftp server ...
TFTP: received error request -- code 1 message File not found: C:/TFTP/nots.txt.
Error - can't download data from TFTP server, error code 17. Abort!
```

To simplify troubleshooting, especially when the file is present on your server but the command doesn't find it, the messages list the complete TFTP path name on your TFTP server.

Possible values: See above.

Default value: N/A

ncopy tftp <ip-addr> <from-name> startup-config

Downloads a copy of a configuration file from a TFTP server into the startup configuration file of the Layer 2 Switch or Layer 3 Switch. To activate this configuration file, reload (reset) the system.

NOTE: This command does the same thing as the **copy tftp startup-config** <ip-addr> <filename> command. See "copy tftp startup-config" on page 5-24.

EXAMPLE:

```
BigIron# ncopy tftp 192.22.33.4 new.cfg startup-config
```

Syntax: ncopy tftp <ip-addr> <from-name> startup-config

Possible values: See above.

Default value: N/A

page-display

Enables page-by-page display of the configuration file. When you display or save the file, one "page" (window-full) of the file is displayed. The following line provides you with options to continue the display or to cancel:

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

If you disable the page-display mode, the CLI displays the entire file without interruption.

Page-display mode is enabled by default. To disable it, enter the **skip-page-display** command.

NOTE: This command is equivalent to the **enable skip-page-display** command at the global CONFIG level.

EXAMPLE:

```
BigIron# page-display
```

Syntax: page-display

Possible values: N/A

Default value: N/A

ping

Verifies connectivity to a Foundry Layer 2 Switch or Layer 3 Switch or other device. The command performs an ICMP echo test to confirm connectivity to the specified device.

NOTE: If you address the ping to the IP broadcast address, the device lists the first four responses to the ping.

EXAMPLE:

```
BigIron# ping 192.22.2.33
```

Syntax: ping <ip addr> | <hostname> [source <ip addr>] [count <num>] [timeout <msec>] [ttl <num>] [size <byte>] [quiet] [numeric] [no-fragment] [verify] [data <1-to-4 byte hex>] [brief]

The only required parameter is the IP address or host name of the device.

NOTE: If the device is a Foundry Layer 2 Switch or Layer 3 Switch, you can use the host name only if you have already enabled the Domain Name Server (DNS) resolver feature on the device from which you are sending the ping. See the "Configuring Basic Features" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

The **source** <ip addr> specifies an IP address to be used as the origin of the ping packets.

The **count** <num> parameter specifies how many ping packets the device sends. You can specify from 1 – 4294967296. The default is 1.

The **timeout** <msec> parameter specifies how many milliseconds the Foundry device waits for a reply from the pinged device. You can specify a timeout from 1 – 4294967296 milliseconds. The default is 5000 (5 seconds).

The **ttl** <num> parameter specifies the maximum number of hops. You can specify a TTL from 1 – 255. The default is 64.

The **size** <byte> parameter specifies the size of the ICMP data portion of the packet. This is the payload and does not include the header. You can specify from 0 – 4000. The default is 16.

The **no-fragment** parameter turns on the “don’t fragment” bit in the IP header of the ping packet. This option is disabled by default.

The **quiet** parameter hides informational messages such as a summary of the ping parameters sent to the device and instead only displays messages indicating the success or failure of the ping. This option is disabled by default.

The **verify** parameter verifies that the data in the echo packet (the reply packet) is the same as the data in the echo request (the ping). By default the device does not verify the data.

The **data** <1 – 4 byte hex> parameter lets you specify a specific data pattern for the payload instead of the default data pattern, “abcd”, in the packet’s data payload. The pattern repeats itself throughout the ICMP message (payload) portion of the packet.

NOTE: For numeric parameter values, the CLI does not check that the value you enter is within the allowed range. Instead, if you do exceed the range for a numeric value, the software rounds the value to the nearest valid value.

The **brief** parameter causes ping test characters to be displayed. The following ping test characters are supported:

! Indicates that a reply was received.

. Indicates that the network server timed out while waiting for a reply.

U Indicates that a destination unreachable error PDU was received.

I Indicates that the user interrupted ping.

Possible values: see above

Default value: see above

ping atmvc

Verifies configuration of an ATM PVC.

EXAMPLE:

```
BigIron# ping atmvc 3/1 0 13
ATM: OAM: sending loopback request on ATM port 3/1, vpi 0, vci 13
BigIron#
ATM: OAM :loopback response received on 3/1, 0,13
```

This example shows the output that results from a successful ping. The command sends the ping from ATM port 3/1 on VPI 0, VCI 13. The device at the other end of the link is configured for the same VPI and VCI and responds to the ping.

Syntax: ping atmvc <portnum> <vpi> <vci>

The <portnum> parameter specifies the port on which the VC is configured on the local device (the device on which you are entering the command).

The <vpi> and <vci> parameters specify the virtual path and virtual channel, which together make the VC.

Possible values: see above

Default value: N/A

pos boot tftp

Boots the POS module from a TFTP server.

EXAMPLE:

```
BigIron# pos boot tftp 209.157.22.26 B2R06000
```

Syntax: pos boot tftp <tftp-server-ip-addr> <pos-image-file-name>

Possible values: The <tftp-server-ip-addr> parameter specifies the IP address of the TFTP server.

The <pos-image-file-name> parameter lists the name of the image file you want the module to boot from the TFTP server.

Default value: N/A

pos copy slot1 | slot2 flash

Allows you to upgrade the system software in the primary flash on all POS modules by copying a software image file from a PCMCIA flash card. By default, the flash code on all the POS modules in the chassis is upgraded. If you want to upgrade only a particular module, you can specify the module's slot number.

EXAMPLE:

To copy a POS image file from a flash card onto all the POS modules in the chassis, enter a command such as the following:

```
BigIron# pos copy slot1 flash P2R07000.bin primary
```

Syntax: pos copy slot1 | slot2 flash <pos-image-file-name> boot | primary | secondary [slot]

The command in this example copies a POS image file named P2R07000.bin from the flash card in slot 1 to all the POS modules in the chassis.

The **boot** parameter specifies that you are upgrading the boot code rather than the flash code.

The **primary** and **secondary** parameters identify either the primary or secondary flash on the module. For each command, the parameter specifies the destination of the copy operation.

The **slot** parameter specifies a chassis slot. This parameter is optional. If you specify a slot number, the upgrade affects only the module in the slot you specify. If you do not specify a slot, the upgrade affects all the POS modules in the chassis.

- Slots in a four-slot chassis are numbered 1 – 4, from top to bottom.
- Slots in an eight-slot chassis are numbered 1 – 8, from left to right.
- Slots in a fifteen-slot chassis are numbered 1 – 15, from left to right.

To copy a POS image file from a flash card onto a specific POS module, enter a command such as the following:

```
BigIron# pos copy slot1 P2R07000.bin flash primary 4
```

The command in this example copies the specified image file onto the POS module in chassis slot 4 only, but does not copy the file to other POS modules in the chassis.

Possible values: See above.

Default value: If you do not specify a slot, the flash code on all the POS modules in the chassis is upgraded.

pos copy tftp flash

Allows you to upgrade the system software in the primary flash on all POS modules by downloading a software image file from a TFTP server. By default, the flash code on all the POS modules in the chassis is upgraded. If you want to upgrade only a particular module, you can specify the module's slot number.

EXAMPLE:

```
BigIron# pos copy tftp flash 109.157.22.26 P2R05300 primary
```

Syntax: pos copy tftp flash <tftp-server-ip-addr> <pos-image-file-name> boot | primary | secondary [<slot-num>]

The **boot** parameter specifies that you are upgrading the boot code rather than the flash code.

The **primary** and **secondary** parameters identify either the primary or secondary flash on the module. For each command, the parameter specifies the destination of the copy operation.

The **slot** parameter specifies a chassis slot. This parameter is optional. If you specify a slot number, the upgrade affects only the module in the slot you specify. If you do not specify a slot, the upgrade affects all the POS modules in the chassis.

- Slots in a four-slot chassis are numbered 1 – 4, from top to bottom.
- Slots in an eight-slot chassis are numbered 1 – 8, from left to right.
- Slots in a fifteen-slot chassis are numbered 1 – 15, from left to right.

Default value: If you do not specify a slot, the flash code on all the POS modules in the chassis is upgraded.

pwd

Indicates which flash card in a Management IV module's PCMCIA slot has the management focus.

The management focus determines the default flash card for a file management operation. For example, when you list a directory of the files on a flash card, the PCMCIA slot parameter is optional. If you do not specify the slot, the software displays the contents of the flash card in the slot that currently has the management focus. As another example, the command for deleting a file from a flash card does not require that you specify the PCMCIA slot. If you do not specify the slot, the command deletes the file from the flash card that has the management focus.

When you power on or reload a device, if the management module contains only one flash card, the slot that contains the flash card receives the management focus by default. If both slots contain flash cards, slot 1 receives the management focus by default.

EXAMPLE:

To display which flash card currently has the management focus, enter the following command:

```
BigIron# pwd
slot1
```

Syntax: pwd

In this example, the flash card in slot 1 has the management focus.

Possible values: N/A

Default value: N/A

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron# quit

BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

rconsole

Logs in to a Velocity Switching Processor (VSP) CPU on the Velocity Management Module.

EXAMPLE:

```
BigIron# rconsole 2 1
BigIron2/1 #
```

This command changes the management session from the module's Management Processor (MP) to VSP CPU 1 on the Velocity Management Module in slot 2. Notice that the end of the command prompt changes to indicate the slot number and VSP CPU number.

Syntax: rconsole <slotnum> <cpunum>

The <slotnum> parameter specifies the chassis slot that contains the module.

- Slots in a four-slot chassis are numbered 1 – 4, from top to bottom.

- Slots in an eight-slot chassis are numbered 1 – 8, from left to right.
- Slots in a fifteen-slot chassis are numbered 1 – 15, from left to right.

The <cpunum> parameter specifies the VSP CPU. The VSP CPUs are numbered from 1 – 3.

To log out of a VSP, enter the **rconsole-exit** command.

For more information, see the "Using the Velocity Management Module" chapter in the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Possible values: See above.

Default value: Disabled

rconsole-exit

Logs out of a Velocity Switching Processor (VSP) CPU on the Velocity Management Module.

EXAMPLE:

To log out from a management session with a VSP CPU, enter the following command at the VSP command prompt:

```
BigIron2/1 # rconsole-exit
BigIron#
```

Syntax: rconsole-exit

NOTE: You must enter the entire command name (**rconsole-exit**). The CLI will not accept abbreviated forms of the command.

Possible values: See above.

Default value: N/A

rd

Another form of the **rmdir** command. See “rmdir” on page 5-45.

reload

Initiates a system reset. All configuration changes made since the last reset or start of the Layer 2 Switch or Layer 3 Switch will be saved to the startup configuration file.

EXAMPLE:

```
BigIron# reload
```

Syntax: reload [after <dd:hh:mm>] | [at <hh:mm:ss> <mm-dd-yy>] | [cancel] [primary | secondary]

Possible values:

after <dd:hh:mm> causes the system to reload after the specified amount of time has passed.

at <hh:mm:ss> <mm-dd-yy> causes the system to reload at exactly the specified time.

cancel cancels the scheduled reload

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.

NOTE: The **reload** command must be typed in its entirety.

Default value: N/A

reload atm

Reloads the software on an individual ATM module, without also reloading the management module.

EXAMPLE:

```
BigIron# reload atm 2
```

This command reloads the ATM module in slot 2. Messages are displayed in the CLI to show the status of the reload. The management module is not also reloaded and thus continues to operate while the ATM module is being reloaded.

Syntax: reload atm <slotnum>

Possible values: See above.

Default value: N/A

reload pos

Reloads the software on an individual POS module, without also reloading the management module.

EXAMPLE:

```
BigIron# reload pos 2
POS MODULE (2) App CPU in running mode:
        CPU 1 in state of POS_STATE_RUNNING
        CPU 2 in state of POS_STATE_RUNNING
Taking down module 2 ...
Module 2 is now deleted
BigIron#Detected module 2 being inserted
Bringing up module 2 ...
POS module at slot 2 is up and running
All POS Modules Up (1)
```

This command reloads the POS module in slot 2. Messages are displayed in the CLI to show the status of the reload. The management module is not also reloaded and thus continues to operate while the POS module is being reloaded.

Syntax: reload pos <slotnum>

Possible values: See above.

Default value: N/A

rename

Renames a file on a flash card in a Management IV module's PCMCIA slot.

EXAMPLE:

To rename a file, enter a command such as the following:

```
BigIron# rename oldname newname
```

Syntax: rename [slot1 | slot2] <old-name> <new-name>

If the command is successful, the CLI displays a new command prompt.

Possible values: See above.

Default value: N/A

reset

Forces the active redundant management module in a Chassis device that contains redundant management modules to switch over to the standby module, thus making it the active redundant management module.

NOTE: This command applies only to devices containing redundant management modules.

EXAMPLE:

To switch over to the redundant management module in chassis slot 2, enter a command such as the following:

```
BigIron# reset 2
```


Syntax: reset <slot-num>

Specify the slot number containing the currently active management module. Do not specify the slot number containing the standby module to which you want to switch over.

Possible values: 'Reset' must be typed in its entirety.

Default value: N/A

rmdir

Removes a subdirectory from a PCMCIA flash card.

EXAMPLE:

```
BigIron# rmdir \TEST
```

Syntax: rd | rmdir [slot1 | slot2] <dir-name>

You can enter either **rd** or **rmdir** for the command name.

The **slot1** | **slot2** parameter specifies a PCMCIA slot.

The <dir-name> parameter specifies the subdirectory you want to delete. You can enter a path name if the subdirectory is not in the current directory.

NOTE: You can remove a subdirectory only if the subdirectory does not contain files or other subdirectories.

If you receive a message such as the following, enter the **pwd** command to verify that the management focus is at the appropriate level of the directory tree.

```
BigIron# rmdir \TEST
File not found
```

Possible values: See above.

Default value: N/A

show

Displays a variety of configuration and statistical information about the Layer 2 Switch or Layer 3 Switch. See "Show Commands" on page 43-1.

skip-page-display

Disables page-display mode. Page-display mode displays the file one page at a time and prompts you to continue or cancel the display. When page-display mode is disabled, if you display or save the configuration file, the CLI displays the entire file without interruption.

Serial console and Telnet CLI users can individually enable or disable page-display mode without affecting the page-display mode of other CLI users.

Page display mode is enabled by default.

NOTE: This command is equivalent to the no **enable skip-page-display** command at the global CONFIG level.

EXAMPLE:

```
BigIron# skip-page-display
```

Syntax: skip-page-display

Possible values: N/A

Default value: N/A

sntp sync

Synchronizes the device's time counter with your SNTP server time. This will allow a system to automatically retrieve clock references from a designated SNTP server in the network.

You define the SNTP server using the **sntp server**... command found at the global CONFIG level. You can also define how often the clock references are validated between the Foundry Layer 2 Switch or Layer 3 Switch and the SNTP server by using the **sntp poll-interval** command found at the global CONFIG level.

NOTE: Configure the **clock timezone** parameter before configuring an SNTP server.

EXAMPLE:

```
BigIron# sntp sync
```

Syntax: sntp sync

Possible values: N/A

Default value: N/A

ssh no-show-host-keys

Configures the Foundry device to hide the RSA host key pair in the running-config file.

EXAMPLE:

```
BigIron# ssh no-show-host-keys
```

Syntax: ssh no-show-host-keys

Possible values: N/A

Default value: N/A

ssh show-host-keys

Configures the Foundry device to display the RSA host key pair in the running-config file after you have hidden it with the **ssh no-show-host-keys** command,

EXAMPLE:

```
BigIron# ssh show-host-keys
```

Syntax: ssh show-host-keys

Possible values: N/A

Default value: N/A

stop-traceroute

Stops an initiated trace on a Foundry Layer 2 Switch or Layer 3 Switch.

EXAMPLE:

```
BigIron# stop-traceroute
```

Syntax: stop-trace-route

Possible values: N/A

Default value: N/A

sync-standby

Immediately synchronizes software between the active and standby management modules. When you synchronize software, the active module copies the software you specify to the standby module, replacing the software on the standby module.

EXAMPLE:

To immediately synchronize the boot code on the standby module with the boot code on the active module, enter the following command at the Privileged EXEC level of the CLI:

```
BigIron# sync-standby boot
```

Syntax: sync-standby boot

To immediately synchronize the flash code (system software) on the standby module with the boot code on the active module, enter the following command at the Privileged EXEC level of the CLI:

```
BigIron# sync-standby code
```

Syntax: sync-standby code

NOTE: The **sync-standby code** command does not synchronize the VSP CPUs (non-management CPUs) on the VM1. To synchronize the VSP CPUs, use the following command:

vm copy tftp flash <tftp-server-ip-addr> <image-file-name> **primary | secondary**

This command upgrades the VSP CPU flash code on all VSP CPUs on both VM1 modules in the chassis.

To immediately synchronize the running-config on the standby module with the running-config on the active module, enter the following command at the Privileged EXEC level of the CLI:

```
BigIron# sync-standby running-config
```

Syntax: sync-standby running-config

To immediately synchronize the startup-config file on the standby module with the startup-config file on the active module, enter the following command at the Privileged EXEC level of the CLI:

```
BigIron# sync-standby startup-config
```

Syntax: sync-standby startup-config

Possible values: See above

Default value: N/A

telnet

Allows a Telnet connection to a remote Layer 2 Switch or Layer 3 Switch using the console. Up to five read-access Telnet sessions are supported on a Foundry Layer 2 Switch or Layer 3 Switch at one time. Write access through Telnet is limited to one session and only one outgoing Telnet session is supported on a Layer 2 Switch or Layer 3 Switch at one time. To see the number of open Telnet sessions at any time, enter the command **show telnet**.

EXAMPLE:

```
BigIron# telnet 208.96.6.101
```

Syntax: telnet <ip-addr> | <hostname> [<portnum>]

Possible values: The port number can be between 1 – 65535.

Default value: If you do not specify a port number, the Telnet connection is established on port 23.

temperature shutdown

Changes the shutdown temperature of a module containing a temperature sensor. If the temperature matches or exceeds the shutdown temperature, the software sends a Syslog message to the Syslog buffer and also to the Syslog server if configured. The software also sends an SNMP trap to the SNMP trap receiver, if you have configured the device to use one.

If the temperature equals or exceeds the shutdown temperature for five consecutive polls of the temperature by the software, the software shuts down the module to prevent damage.

EXAMPLE:

To change the shutdown temperature from 55 to 57 degrees Celsius, enter the following command:

```
BigIron# temperature shutdown 57
```

Syntax: temperature shutdown <value>

The <value> can be 0 – 125.

Possible values: 0 – 125 degrees Celsius

Default value: 55

temperature warning

Changes the warning temperature of a module containing a temperature sensor. If the temperature of the module reaches the warning value, the software sends a Syslog message to the Syslog buffer and also to the Syslog server, if configured. In addition, the software sends an SNMP trap to the SNMP trap receiver, if you have configured the device to use one.

NOTE: You cannot set the warning temperature to a value higher than the shutdown temperature.

EXAMPLE:

To change the warning temperature from 45 to 47 degrees Celsius, enter the following command:

```
BigIron# temperature warning 57
```

Syntax: temperature warning <value>

The <value> can be 0 – 125.

Possible values: 0 – 125 degrees Celsius

Default value: 45

terminal monitor

Enables real-time display of Syslog messages for a Telnet or SSH session.

NOTE: You first must enable real-time display by entering the **logging console** command at the global CONFIG level of the CLI. You can enter this command from the serial console or from a Telnet or SSH session. See “logging” on page 6-82.

EXAMPLE:

To enable the real-time display for a Telnet or SSH session, enter the following command from the Privileged EXEC level of the session:

```
telnet@BigIron# 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@BigIron# terminal monitor
Syslog trace was turned OFF
```

Here is an example of how the Syslog messages are displayed:

```
telnet@BigIron# terminal monitor
Syslog trace was turned ON
SYSLOG: <9>BigIron, Power supply 2, power supply on left connector, failed

SYSLOG: <14>BigIron, Interface ethernet 1/6, state down

SYSLOG: <14>BigIron, Interface ethernet 1/2, state up
```

Possible values: N/A

Default value: Disabled

traceroute

Allows you to trace the path from the current Foundry Layer 2 Switch or Layer 3 Switch to a host address.

The CLI displays trace route information for each hop as soon as the information is received. Traceroute requests display all responses to a given TTL. In addition, if there are multiple equal-cost routes to the destination, the Foundry device displays up to three responses by default.

EXAMPLE:

```
BigIron> traceroute 192.33.4.7 minttl 5 maxttl 5 timeout 5
```

Syntax: traceroute <host-ip-addr> [maxttl <value>] [minttl <value>] [numeric] [timeout <value>] [source-ip <ip addr>]

Possible and default values:

minttl – minimum TTL (hops) value: Possible values are 1 – 255. Default value is 1 second.

maxttl – maximum TTL (hops) value: Possible values are 1 – 255. Default value is 30 seconds.

timeout – Possible values are 1 – 120. Default value is 2 seconds.

numeric – Lets you change the display to list the devices by their IP addresses instead of their names.

source-ip <ip addr> – Specifies an IP address to be used as the origin for the traceroute.

undelete

Recovers a file deleted from a PCMCIA flash card.

NOTE: When you delete a file from a flash card, the CLI leaves the file intact but removes the first letter in the file name from the file directory. However, if you save file changes or new files that use part of the space occupied by the deleted file, you cannot undelete the file. The **undelete** command lists only the files that can be undeleted.

EXAMPLE:

```
BigIron# undelete
Undelete file "?LD.CFG" ? (enter 'y' or 'n'): y
Input one character: O
File recovered successfully and named to OLD.CFG
```

The command in this example starts the undelete process for the flash card and subdirectory that currently have the management focus. For each file that can be undeleted, the CLI displays the remaining name entry in the file directory and prompts you for the first character of the file name. You can enter any valid file name character. You do not need to enter the character that was used before in the deleted file name.

Once you enter a character and the CLI undeletes the file, the CLI continues with the next file that can be undeleted. For each file, specify “y” or “n”, and specify a first character for the files that you select to undelete.

To end the undelete process, enter the CTRL + C key combination.

Syntax: undelete [slot1 | slot2] [\<to-dir-path>]

Possible values: See above

Default value: N/A

vm boot

Boots software on the Velocity Management Module. To use this command, you must set the module’s boot source to be interactive. See “vm boot” on page 6-141.

EXAMPLE:

After you set the boot source to interactive and reboot, enter a command such as the following at the Privileged EXEC level of the CLI to boot the VSPs:

```
BigIron# vm boot tftp 192.168.1.170 vsp07300.bin
```

This command copies the Velocity Switching Processor (VSP) flash code image from the specified TFTP server to a VSP address space from which the VSP can boot.

Syntax: vm boot primary | secondary | tftp <ip-addr> <image-file-name>

Possible values: See above

Default value: N/A

vm copy flash flash

Copies software from one flash area to the other on a Velocity Management Module.

EXAMPLE:

To copy the flash code from the primary flash to the secondary flash, enter a command such as the following:

```
BigIron# vm copy flash flash secondary
```

Syntax: vm copy flash flash primary | secondary

The **primary** and **secondary** parameters identify either the primary or secondary flash as the destination of the copy operation.

Possible values: See above

Default value: N/A

vm copy tftp flash

Upgrades software on a Velocity Management Module.

NOTE: The command upgrades either the Management Processor (MP) or Velocity Switching Processor (VSP) software, depending on the type of flash image you copy. To upgrade both types of software, you need to enter the command twice.

EXAMPLE:

```
BigIron# vm copy tftp flash 109.157.22.26 VSP07300.bin primary
```

This command upgrades the VSPs by copying a flash code image from a TFTP server to the primary flash for each of the VSPs on the module.

Syntax: vm copy tftp flash <tftp-server-ip-addr> <image-file-name> boot | primary | secondary

The **boot** parameter indicates that are upgrading boot code.

The **primary** and **secondary** parameters indicate that you are upgrading flash code and identify either the primary or secondary flash area as the destination of the copy operation.

Possible values: See above

Default value: N/A

whois

Performs a whois lookup on a specified domain.

EXAMPLE:

```
BigIron# whois boole.com
```

Syntax: whois <host-ip-addr> | <domain>

Possible values: <host-ip-addr> is a valid IP address; <domain> is a valid domain name.

NOTE: A DNS gateway must be defined in order to use this command.

Default value: N/A

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron# write memory
```


Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron# wr t
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 6
Global CONFIG Commands

aaa accounting

Configures RADIUS or TACACS+ accounting for recording information about user activity and system events. When you configure accounting on a Foundry device, information is sent to an accounting server when specified events occur, such as when a user logs into the device or the system is rebooted.

EXAMPLE:

To send an Accounting Start packet to a TACACS+ accounting server when an authenticated user establishes a Telnet or SSH session on the Foundry device, and an Accounting Stop packet when the user logs out:

```
BigIron(config)# aaa accounting exec default start-stop tacacs+
```

Syntax: [no] aaa accounting exec default start-stop radius | tacacs+ | none

You can configure accounting for CLI commands by specifying a privilege level whose commands require accounting. For example, to configure the Foundry device to perform RADIUS accounting for the commands available at the Super User privilege level (that is; all commands on the device), enter the following command:

```
BigIron(config)# aaa accounting commands 0 default start-stop radius
```

Syntax: [no] aaa accounting commands <privilege-level> default start-stop radius | tacacs+ | none

The <privilege-level> parameter can be one of the following:

- **0** – Records commands available at the Super User level (all commands)
- **4** – Records commands available at the Port Configuration level (port-config and read-only commands)
- **5** – Records commands available at the Read Only level (read-only commands)

You can configure accounting to record when system events occur on the Foundry device. System events include rebooting and when changes to the active configuration are made.

The following command causes an Accounting Start packet to be sent to a TACACS+ accounting server when a system event occurs, and a Accounting Stop packet to be sent when the system event is completed:

```
BigIron(config)# aaa accounting system default start-stop tacacs+
```

Syntax: [no] aaa accounting system default start-stop radius | tacacs+ | none

Possible values: see above

Default value: N/A

aaa authentication

Defines an authentication-method list for access to a Layer 2 Switch or Layer 3 Switch.

EXAMPLE:

To configure an access method list, enter a command such as the following:

```
BigIron(config)# aaa authentication web-server default local
```

This command configures the device to use the local user accounts to authenticate access to the device through the Web management interface. If the device does not have a user account that matches the user name and password entered by the user, the user is not granted access.

To configure the device to consult a RADIUS server first for Enable access, then consult the local user accounts if the RADIUS server is unavailable, enter the following command:

```
BigIron(config)# aaa authentication enable default radius local
```

Syntax: [no] aaa authentication snmp-server | web-server | enable | login | dot1x default <method1> [<method2>] [<method3>] [<method4>] [<method5>] [<method6>] [<method7>]

Syntax: aaa authentication login privilege-mode

The **snmp-server** | **web-server** | **enable** | **login** | **dot1x** parameter specifies the type of access this authentication-method list controls. You can configure one authentication-method list for each type of access.

The **aaa authentication login privilege-mode** command configures the device so that a user enters Privileged EXEC mode after a Telnet or SSH login.

NOTE: TACACS/TACACS+ is supported only for enable and login. RADIUS is supported only for enable, login, and dot1x.

The <method1> parameter specifies the primary authentication method. The remaining optional <method> parameters specify the secondary methods to try if an error occurs with the primary method. A method can be one of the values listed in the Method Value column in the following table.

Table 6.1: Authentication Method Values

Method Value	Description
tacacs or **tacacs+**	A TACACS/TACACS+ server. You can use either parameter. Each parameter supports both TACACS and TACACS+. You also must identify the server to the device using the **tacacs-server** command.
radius	A RADIUS server. You also must identify the server to the device using the **radius-server** command.
local	A local user name and password you configured on the device. Local user names and passwords are configured using the **username**… command.
line	The password you configured for Telnet access. The Telnet password is configured using the **enable telnet password**… command.
enable	The super-user "enable" password you configured on the device. The enable password is configured using the **enable super-user-password**… command.
none	No authentication is used. The device automatically permits access.

Possible values: see above

Default value: N/A

aaa authorization

Configures authorization for controlling access to management functions in the CLI. Foundry devices support RADIUS and TACACS+ authorization.

- When RADIUS authorization is enabled, the Foundry device consults the list of commands supplied by the RADIUS server during authentication to determine whether a user can execute a command he or she has entered.
- Two kinds of TACACS+ authorization are supported: Exec authorization determines a user's privilege level when they are authenticated; Command authorization consults a TACACS+ server to get authorization for commands entered by the user

EXAMPLE:

You enable command authorization by specifying a privilege level whose commands require authorization. For example, to configure the Foundry device to perform RADIUS authorization for the commands available at the Super User privilege level (that is; all commands on the device), enter the following command:

```
BigIron(config)# aaa authorization commands 0 default radius
```

Syntax: [no] aaa authorization commands <privilege-level> default tacacs+ | radius | none

The <privilege-level> parameter can be one of the following:

- **0** – Authorization is performed for commands available at the Super User level (all commands)
- **4** – Authorization is performed for commands available at the Port Configuration level (port-config and read-only commands)
- **5** – Authorization is performed for commands available at the Read Only level (read-only commands)

NOTE: TACACS+ and RADIUS command authorization is performed only for commands entered from Telnet, SSH, or console sessions. No authorization is performed for commands entered using the Web management interface, or IronView.

NOTE: Since RADIUS authorization relies on the command list supplied by the RADIUS server during authentication, you cannot perform RADIUS authorization without RADIUS authentication.

When TACACS+ exec authorization is configured, the Foundry device consults a TACACS+ server to determine the privilege level for an authenticated user. To configure TACACS+ exec authorization, on the Foundry device, enter the following command:

```
BigIron(config)# aaa authorization exec default tacacs+
```

Syntax: [no] aaa authorization exec default tacacs+ | none

Possible values: see above

Default value: N/A

access-list (standard)

Configures standard Access Control Lists (ACLs), which permit or deny packets based on source IP address (in contrast to extended ACLs, which permit or deny packets based on source and destination IP address and also based on IP protocol information). You can configure up to 99 standard ACLs. You can configure up to 1024 individual ACL entries. There is no limit to the number of ACL entries an ACL can contain except for the system-wide limitation of 1024 total ACL entries.

Starting in software release 07.6.03, you can use standard ACLs to control the following multicast features:

- Limit the number of multicast groups that are covered by a static rendezvous point (RP). See "rp-address" on page 17-4.
- Control which multicast groups for which candidate RPs sends advertisement messages to bootstrap routers. See "rp-candidate" on page 17-5.

- Identify which multicast group packets will be forwarded or blocked on an interface. See “ip multicast boundary” on page 7-31.

EXAMPLE:

To configure a standard ACL and apply it to outgoing traffic on port 1/1, enter the following commands.

```
BigIron(config)# access-list 1 deny host 209.157.22.26 log
BigIron(config)# access-list 1 deny 209.157.29.12 log
BigIron(config)# access-list 1 deny host IPHost1 log
BigIron(config)# access-list 1 permit any
BigIron(config)# int eth 1/1
BigIron(config-if-1/1)# ip access-group 1 out
BigIron(config-if-1/1)# write memory
```

The commands in this example configure an ACL to deny packets from three source IP addresses from being forwarded 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.

Syntax: [no] access-list <num> deny | permit <source-ip> | <hostname> <wildcard> [log]

Syntax: [no] access-list <num> deny | permit <source-ip> | <hostname>/<mask-bits> [log]

Syntax: [no] access-list <num> deny | permit host <source-ip> | <hostname> [log]

Syntax: [no] access-list <num> deny | permit any [log]

Syntax: [no] ip access-group <num> in | out

The <num> parameter is the access list number and can be 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 sub-net 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”.

NOTE: When you save ACL policies to the startup-config file, the software changes your <source-ip> values if appropriate to contain zeros where the packet value must match. For example, if you specify 209.157.22.26/24 or 209.157.22.26 255.255.255.0, then save the startup-config file, the values appear as 209.157.22.0/24 (if you have enabled display of sub-net lengths) or 209.157.22.0 255.255.255.0 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 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 sub-net 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.

The **in | out** parameter specifies whether the ACL applies to incoming traffic or outgoing traffic on the port to which you apply the ACL.

Possible values: see above

Default value: N/A

access-list (extended)

Configures extended ACLs, which 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)

EXAMPLE:

To configure an extended ACL that blocks all Telnet traffic received on port 1/1 from IP host 209.157.22.26, enter the following commands.

```
BigIron(config)# access-list 101 deny tcp host 209.157.22.26 any eq telnet log
BigIron(config)# access-list 101 permit ip any any
BigIron(config)# int eth 1/1
BigIron(config-if-1/1)# ip access-group 101 in
BigIron(config)# write memory
```

Syntax: [no] access-list <num> deny | permit <ip-protocol> <source-ip> | <hostname> <wildcard>
[<operator> <source-tcp/udp-port>] <destination-ip> | <hostname> [<icmp-type>] <wildcard>
[<operator> <destination-tcp/udp-port>]
[established]
[precedence <name> | <num>]
[tos <num>]
[priority 0 | 1 | 2 | 3]
[priority-force 0 | 1 | 2 | 3]
[priority-mapping <8021p-value>]
[dscp-mapping <dscp-value>]
[dscp-marking <dscp-value>]
[log]

Syntax: [no] access-list <num> deny | permit host <ip-protocol> any any [log]

Syntax: [no] ip access-group <num> in | out

The <num> parameter indicates the ACL number and can be from 100 – 199 for an extended ACL.

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. In release 07.6.01 and later, 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 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".

NOTE: When you save ACL policies to the startup-config file, the software changes your IP address values if appropriate to contain zeros where the packet value must match. For example, if you specify 209.157.22.26/24 or 209.157.22.26 255.255.255.0, then save the startup-config file, the values appear as 209.157.22.0/24 (if you have enabled display of sub-net lengths) or 209.157.22.0 255.255.255.0 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 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 sub-net 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> parameter specifies the ICMP protocol type.

NOTE: This parameter applies only if you specified **icmp** as the <ip-protocol> value. The <icmp-type> parameter is supported in software release 07.2.06 and later.

NOTE: If you do not specify a message type, the ACL applies to all types of ICMP messages. The <num> parameter can be a value from 0 – 255.

This parameter can have one of the following values:

- **echo**
- **echo-reply**
- **information-request**
- **mask-reply**
- **mask-request**
- **parameter-problem**
- **redirect**
- **source-quench**
- **time-exceeded**
- **timestamp-reply**
- **timestamp-request**
- **unreachable**
- <num>

The <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**.
- **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.
- **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.

The <tcp/udp-port> parameter specifies the TCP or UDP port number or well-known name. In release 07.6.01 and later, 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** | **out** parameter specifies whether the ACL applies to incoming traffic or outgoing traffic on the port to which you apply the ACL.

The **precedence** <name> | <num> parameter of the **ip access-list** command specifies the IP precedence. The **precedence** option for 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**.
- **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 **priority** option enables you to assign traffic that matches the ACL to a specific hardware forwarding queue (qosp0, qosp1, qosp2, or qosp3). The **0 | 1 | 2 | 3** parameter specifies the QoS queue:

- 0 – qosp0
- 1 – qosp1
- 2 – qosp2
- 3 – qosp3

NOTE: This **priority** option provides the same function as the Layer 4 IP access policies supported on BigIron Chassis devices. If you configure both a Layer 4 IP access policy and an extended ACL to set the hardware forwarding priority for the same traffic, the device uses the ACL instead of the IP access policy.

The **priority-force** parameter allows you assign packets of outgoing traffic that match the ACL to a specific hardware forwarding queue, even though the incoming packet may be assigned to another queue. Specify one of the following QoS queue:

- 0 – qosp0
- 1 – qosp1
- 2 – qosp2
- 3 – qosp3

The **priority-mapping** <8021p-value> parameter matches on the packet's 802.1p priority.

NOTE: This option does not change the packet's priority through the device or mark the packet. This option is not supported on 10 Gigabit Ethernet modules.

The **dscp-mapping** <dscp-value> parameter matches on the packet's DSCP value.

NOTE: This option does not change the packet's priority through the device or mark the packet.

The **dscp-marking** <dscp-value> parameter maps a DSCP value to an internal forwarding priority. The DSCP value can be from 0 – 63.

The **log** parameter enables SNMP traps and Syslog messages for packets denied by the ACL.

Possible values: see above

Default value: N/A

access-list rate-limit

Configures a rate-limiting ACL.

NOTE: After you configure the rate limiting policy, you need to apply the policy to an interface for the policy to take effect. See “rate-limit input | output” on page 7-70.

EXAMPLE:

The following command configures a rate limit ACL to characterize the traffic. In this case, the rate policy is for a specific host, so the rate limit ACL specifies a host MAC address.

```
BigIron(config)# access-list rate-limit 100 aaaa.bbbb.cccc
```

Syntax: [no] access-list rate-limit <num> <mac-addr> | <precedence> | mask <precedence-mask>

The <num> parameter specifies the ACL number.

The <mac-addr> | <precedence> | mask <precedence-mask> parameter specifies a MAC address, an IP precedence, or a mask value representing a set of IP precedence values.

To specify a MAC address, enter the address in the following format: xxxx.xxxx.xxxx.

To specify an IP precedence, specify one of the following:

- **0** – The ACL matches packets that have the routine precedence.
- **1** – The ACL matches packets that have the priority precedence.
- **2** – The ACL matches packets that have the immediate precedence.
- **3** – The ACL matches packets that have the flash precedence.
- **4** – The ACL matches packets that have the flash override precedence.
- **5** – The ACL matches packets that have the critical precedence.
- **6** – The ACL matches packets that have the internetwork control precedence.
- **7** – The ACL matches packets that have the network control precedence.

To specify a mask value for a set of IP precedence values, enter **mask** followed by a two-digit hexadecimal number for the precedence values.

The precedence values are in an 8-bit field in the IP packet header. To calculate the hexadecimal number for a combination of precedence values, write down the values for the entire field to create the binary number for the mask value, then convert the number to hexadecimal. For example, to specify a mask for precedences 2, 4, and 5, write down the following values for the precedence field:

Bit position	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**
Precedence	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
Bit pattern	0	0	1	1	0	1	0	0

Then, reading the digits from right to left, convert the number to hexadecimal. In this case, 00110100 binary becomes 0x34. Enter the mask as **mask 34**.

For simplicity, you can convert the digits in groups of four bits each.

For example, you can convert bits 1 – 4 (binary 0100) to get hexadecimal “4” for the right digit. Then convert bits 5 – 8 (binary 0011) to get hexadecimal “3” for the left digit. The result is “34”.

Alternatively, you can enter the entire eight-bit binary number in a calculator, then convert the number to hexadecimal. For example, you can enter the binary number “00110100” and convert it to hexadecimal to get “34”. (Without the leading zeros, enter “110100”.)

NOTE: The bits appear in this order in the IP precedence field and the software reads them from right to left. The least significant digit is the rightmost digit (bit position 1) and the most significant digit is the leftmost digit (bit position 8).

Possible values: See above

Default value: N/A

access-list remark

Adds optional comment text to describe entries in an ACL. The comment text appears in the output of **show** commands that display ACL information.

EXAMPLE:

The following commands add comments to entries in ACL 100:

```
BigIron(config)# access-list 100 remark The following line permits TCP packets
BigIron(config)# access-list 100 permit tcp 192.168.4.40/24 2.2.2.2/24

BigIron(config)# access-list 100 remark The following permits UDP packets
BigIron(config)# access-list 100 permit udp 192.168.2.52/24 2.2.2.2/24

BigIron(config)# access-list 100 deny ip any any
```

The following commands add comments to entries in ACL TCP/UDP:

```
BigIron(config)# access-list TCP/UDP remark The following line permits TCP packets
BigIron(config)# access-list TCP/UDP permit tcp 192.168.4.40/24 2.2.2.2/24
BigIron(config)# access-list TCP/UDP remark The following permits UDP packets
BigIron(config)# access-list TCP/UDP permit udp 192.168.2.52/24 2.2.2.2/24
BigIron(config)# access-list TCP/UDP deny ip any any
```

Syntax: [no] access-list <acl-num> | <acl-name> remark <comment-text>

Possible values:

Enter the ACL's number or ACL's 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.

Default value: N/A

acl-denied-icmp-msg

Enables a Layer 3 Switch to send an ICMP unreachable message to a device when an ACL denies a packet from the device.

By default, a Foundry device does not send a message to another device when an ACL on the Foundry device denies a packet from the other device.

NOTE: This command applies only to Layer 3 Switches.

NOTE: This command does not take effect in the following cases:

—Hardware-based ACLs are enabled.

—The **hw-drop-acl-denied-packet** command is in effect.

In either case, all packets denied by the ACL are dropped by hardware without sending an ICMP message.

EXAMPLE:

```
BigIron(config)# acl-denied-icmp-msg
```

Syntax: [no] acl-denied-icmp-msg

The command applies globally to all ACLs configured on the device.

Possible values: N/A

Default value: Disabled

aggregated-vlan

Enables a larger Ethernet frame size for VLAN aggregation. This feature changes the maximum Ethernet size to 1530 bytes.

NOTE: Use this command when you are configuring Super Aggregated VLANs. See the "Configuring VLANs" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

EXAMPLE:

```
BigIron(config)# aggregated-vlan
```

Syntax: aggregated-vlan

Possible values: N/A

Default value: Disabled

all-client

Restricts management access to the Foundry device to the host whose IP address you specify. No other device except the one with the specified IP address can access the Foundry device through Telnet (CLI), the Web (Web management interface), or SNMP (IronView).

If you want to restrict access for some of the management platforms but not all of them, use one or two of the following commands:

- **snmp-client** – restricts IronView access and all other SNMP access. See "snmp-client" on page 6-116.
- **telnet-client** – restricts Telnet access. See "telnet-client" on page 6-135.
- **web-client** – restricts web access. See "web-client" on page 6-143.

EXAMPLE:

To restrict all management access to the Foundry device to the host with IP address 209.157.22.26, enter the following command:

```
BigIron(config)# all-client 209.157.22.26
```

Syntax: [no] all-client <ip-addr>

Possible values: a valid IP address. You can enter one IP address with the command. You can use the command up to ten times for up to ten IP addresses.

Default value: N/A

appletalk arp-age

Defines how long an AppleTalk ARP entry will remain active before being aged out.

EXAMPLE:

```
BigIron(config)# appletalk arp-age 115
```

Syntax: appletalk arp-age <1 – 240>

Possible values: 1 – 240 minutes

Default value: 10 minutes

appletalk arp retransmit-count

Allows you to modify the maximum number of times that a packet will be sent out for ARP cache informational updates. The packet will be sent out to the maximum amount defined, until the information is received.

If no response is received before the count number expires, no additional packets will be sent.

EXAMPLE:

To modify the number of times packet requests will be sent out for ARP updates from the default value of 2 to 8, enter the following:

```
BigIron(config)# appletalk arp retransmit-count 8
```

Syntax: appletalk arp retransmit-count <value>

Possible values: 1 – 10

Default value: 2

appletalk arp retransmit-interval

Allows you to modify the interval between the transmission of ARP packets.

EXAMPLE:

To modify the retransmission interval from the default value of 1 to 15 seconds, enter the following:

```
BigIron(config)# appletalk arp retransmit-interval 15
```

Syntax: appletalk arp retransmit-interval <value>

Possible values: 1 – 120 seconds

Default value: 1

appletalk glean-packets

When the **glean-packets** parameter is enabled on an AppleTalk router, it will try to learn the MAC address from the packet instead of sending out an AARP request.

EXAMPLE:

To enable glean packets on an AppleTalk router, enter the following:

```
BigIron(config)# appletalk glean-packets
```

Syntax: appletalk glean-packets

Possible values: enabled or disabled

Default value: disabled

appletalk qos socket

You can use the Quality of Service (QoS) socket parameter to assign a higher priority to specific AppleTalk sockets. Enter a value from 0 – 7 (Chassis devices, the FastIron 4802, or the TurboIron/8) or either normal or high (Stackable Layer 3 Switches).

For information about Foundry QoS, see the "IronClad Quality of Service" chapter in the *Foundry Enterprise Configuration and Management Guide*.

EXAMPLE:

To assign socket 123 to the premium queue, enter the following command:

```
BigIron(config)# appletalk qos socket 123 priority 7
```

Here is the syntax for Chassis devices, the FastIron 4802, and the TurboIron/8 (Layer 3 Switches).

Syntax: [no] appletalk qos socket <num> priority <num>

Here is the syntax for Stackable Layer 3 Switches.

Syntax: [no] appletalk qos socket <num> high | normal

The first <num> parameter specifies the socket number.

The second <num> parameter (Chassis devices) can be from 0 – 7 and specifies the IEEE 802.1 equivalent to one of the four QoS queues.

The **high** | **normal** parameter (Stackable devices) indicates the priority level.

Possible values: See above.

Default value: By default, all AppleTalk sockets are in the best effort queue (Chassis devices, the FastIron 4802, and the TurboIron/8) or normal queue (Stackable Layer 3 Switches).

appletalk rtmp-update-interval

Allows you to modify how often RTMP updates are sent out on AppleTalk interfaces.

EXAMPLE:

To change the value to 50 seconds from a default value of 10 seconds, enter the following:

```
BigIron(config)# appletalk rtmp-update-interval 50
```

Syntax: appletalk rtmp-update-interval <seconds>

Possible values: 1 – 3600 seconds

Default value: 10 seconds

appletalk zip-query-interval

Allows you to modify how often ZIP query messages are retransmitted.

EXAMPLE:

To change the ZIP query interval to 30 seconds from a default value of 10 seconds, enter the following:

```
BigIron(config)# appletalk zip-query-interval 30
```

Syntax: appletalk zip-query-interval <seconds>

Possible values: 1 – 1000 seconds

Default value: 10 seconds

arp

Enters a static IP ARP entry for static routes on a Foundry Layer 3 Switch. This command is not available on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron(config)# arp 1 192.53.4.2 1245.7654.2348 e 4/11
```

Syntax: [no] arp <num> <ip-addr> <mac-addr> ethernet <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 **ethernet** <portnum> command specifies the port number attached to the device that has the MAC address of the entry.

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

Possible values: The maximum number of ARP entries you can add depends on the device. To display the maximum number you can configure on your device, enter the **show default values** command and look at the row of information for the **ip-arp** parameter. See “show default” on page 43-21.

Default value: N/A

atm boot interactive

Changes the boot source for an ATM module.

EXAMPLE:

To change the boot source from the ATM module’s primary flash to its secondary flash, enter the following commands:

```
BigIron(config)# atm boot secondary
BigIron(config)# write memory
```

Syntax: atm boot interactive | primary | secondary

The **primary** and **secondary** parameters identify either the primary or secondary flash on the ATM module.

The **interactive** parameter enables you to enter a separate command after the module comes up to boot the module from a TFTP server. If you use this method, you also need to use the **atm boot tftp...** command to boot the module after the module comes up. See “atm boot tftp” on page 5-2.

Possible values: See above

Default value: primary

auto-acl-rebind

Enables automatic unbinding and rebinding of ACLs. Use this command if you going to copy a configuration file containing ACLs into the device’s running-config.

EXAMPLE:

```
BigIron(config)# auto-acl-rebind
BigIron(config)# end
BigIron# copy tftp running newacls.cfg
```

Possible values: Enabled or disabled

Default value: Disabled

banner exec

Configures the Foundry device to display a message when a user enters the Privileged EXEC CLI level.

EXAMPLE:

```
BigIron(config)# banner exec $ (Press Return)
Enter TEXT message, End with the character '$'.
You are entering Privileged EXEC level
Don’t foul anything up! $
```

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

A delimiting character is established on the first line of the **banner exec** 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 exec** command.

Possible values: N/A

Default value: N/A

banner incoming

Configures 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.

EXAMPLE:

```
BigIron(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>

A delimiting character is established on the first line of the **banner incoming** 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 incoming** command.

Possible values: N/A

Default value: N/A

banner motd

Configures the Foundry device to display a message on a user's terminal when he or she establishes a Telnet CLI session.

EXAMPLE:

To display the message "Welcome to BigIron 15000!" when a Telnet CLI session is established:

```
BigIron(config)# banner motd $ (Press Return)
Enter TEXT message, End with the character '$'.
Welcome to BigIron 15000! $
```

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

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.

When you access the Web management interface, the banner is displayed on the login panel.

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

Possible values: N/A

Default value: N/A

boot system bootp

Configures the device to use BootP as the primary boot source.

NOTE: If you enter another **boot system** command at the global CONFIG level after entering this command, the software adds the new boot source as the primary source and changes the previously entered source to be the secondary source.

EXAMPLE:

```
BigIron(config)# boot system bootp
```


Syntax: boot system bootp

Possible values: N/A

Default value: primary flash

boot system flash primary

Configures the device to use the primary flash location as the primary boot source. This is the default primary boot source.

NOTE: If you enter another **boot system** command at the global CONFIG level after entering this command, the software adds the new boot source as the primary source and changes the previously entered source to be the secondary source.

EXAMPLE:

```
BigIron(config)# boot system flash primary
```

Syntax: boot system flash primary

Possible values: N/A

Default value: primary flash

boot system flash secondary

Configures the device to use the secondary flash location as the primary boot source.

NOTE: If you enter another **boot system** command at the global CONFIG level after entering this command, the software adds the new boot source as the primary source and changes the previously entered source to be the secondary source.

EXAMPLE:

```
BigIron(config)# boot system flash secondary
```

Syntax: boot system flash secondary

Possible values: N/A

Default value: primary flash

boot system slot1 | slot2

Configures the device to use a PCMCIA flash card as the primary boot source.

EXAMPLE:

```
BigIron(config)# boot system slot2 B2R07100.bin
```

The command in this example sets PCMCIA slot 1 as the primary boot source for the device. When you reload the software or power cycle the device, the device looks for the flash image file you specify on the flash card in the slot you specify.

Syntax: boot system slot1 | slot2 <file-name>

NOTE: The command syntax is the same for immediately reloading and for changing the primary boot source, except the <file-name> must be the full path name. You cannot specify a relative path name. If the first character in the path name is not a backslash (\ ), the CLI treats the name you specify as relative to the root directory.

If you enter multiple **boot system** commands at the global CONFIG level, the software places them in the running-config in the order you enter them, and saves them to the startup-config file in the same order when you save the configuration. When you reload or power cycle the device, the device tries the boot sources in the order they appear in the startup-config file and running-config.

Possible values: See above

Default value: N/A

boot system tftp

Configures the device to use a TFTP server as the primary boot source.

NOTE: If you enter another **boot system** command at the global CONFIG level after entering this command, the software adds the new boot source as the primary source and changes the previously entered source to be the secondary source.

EXAMPLE:

```
BigIron(config)# boot sys tftp 192.22.33.44 current.img
```

NOTE: Before entering the TFTP boot command, you must first assign an IP address, IP mask and default gateway (if applicable) at the boot prompt as shown.

EXAMPLE:

```
boot> ip address 192.22.33.44 255.255.255.0
boot> ip default-gateway 192.22.33.1
```

You now can proceed with the **boot system tftp**… command.

Syntax: boot system tftp <ip-addr> <filename>

Possible values: N/A

Default value: primary flash

bootp-relay-max-hops

Defines the maximum number of hops that a BootP request will be allowed to traverse before being dropped.

EXAMPLE:

```
BigIron(config)# bootp-relay-max-hops 5
```

Syntax: bootp-relay-max-hops <value>

Possible values: 1 – 15

Default value: 4

broadcast filter

Configures a Layer 2 broadcast packet filter. You can filter on all broadcast traffic or on IP UDP broadcast traffic.

EXAMPLE:

To configure a Layer 2 broadcast filter to filter all types of broadcasts, then apply the filter to ports 1/1, 1/2, and 1/3, enter the following commands:

```
BigIron(config)# broadcast filter 1 any
BigIron(config-bcast-filter-id-1)# exclude-ports ethernet 1/1 to 1/3
BigIron(config-bcast-filter-id-1)# write memory
```

EXAMPLE:

To configure two filters, one to filter IP UDP traffic on ports 1/1 – 1/4, and the other to filter all broadcast traffic on port 4/6, enter the following commands:

```
BigIron(config)# broadcast filter 1 ip udp
BigIron(config-bcast-filter-id-1)# exclude-ports ethernet 1/1 to 1/4
BigIron(config-bcast-filter-id-1)# exit
BigIron(config)# broadcast filter 2 any
BigIron(config-bcast-filter-id-2)# exclude-ports ethernet 4/6
```



```
BigIron(config-bcast-filter-id-2)# write memory
```

EXAMPLE:

To configure an IP UDP broadcast filter and apply that applies only to port-based VLAN 10, then apply the filter to two ports within the VLAN, enter the following commands:

```
BigIron(config)# broadcast filter 4 ip udp vlan 10
BigIron(config-bcast-filter-id-4)# exclude-ports eth 1/1 eth 1/3
BigIron(config-bcast-filter-id-4)# write memory
```

Syntax: [no] broadcast filter <filter-id> any | ip udp [vlan <vlan-id>]

The <filter-id> specifies the filter number and can a number from 1 – 8. The software applies the filters in ascending numerical order. As soon as a match is found, the software takes the action specified by the filter (block the broadcast) does not compare the packet against additional broadcast filters.

You can specify **any** or **ip udp** as the type of broadcast traffic to filter. The **any** parameter prevents all broadcast traffic from being sent on the specified ports. The **ip udp** parameter prevents all IP UDP broadcasts from being sent on the specified ports but allows other types of broadcast traffic.

If you specify a port-based VLAN ID, the filter applies only to the broadcast domain of the specified VLAN, not to all broadcast domains (VLANs) on the device.

As soon as you press Enter after entering the command, the CLI changes to the configuration level for the filter you are configuring. You specify the ports to which the filter applies at the filter's configuration level.

Syntax: [no] exclude-ports ethernet <portnum> to <portnum>

Or

Syntax: [no] exclude-ports ethernet <portnum> ethernet <portnum>

These commands specify the ports to which the filter applies.

NOTE: This is the same command syntax as that used for configuring port-based VLANs. Use the first command for adding a range of ports. Use the second command for adding separate ports (not in a range). You also can combine the syntax. For example, you can enter **exclude-ports ethernet 1/4 ethernet 2/6 to 2/9**.

Possible values: see above

Default value: N/A

broadcast limit

Specifies the maximum number of broadcast packets the device can forward each second. By default the device sends broadcasts and all other traffic at wire speed and is limited only by the capacities of the hardware. However, if other devices in the network cannot handle unlimited broadcast traffic, this command allows you to relieve those devices by throttling the broadcasts at the Foundry device.

NOTE: The broadcast limit does not affect multicast or unicast traffic. However, you can use the **multicast limit** and **unknown-unicast limit** commands to control these types of traffic. See “multicast limit” on page 6-90 and “unknown-unicast limit” on page 6-138.

EXAMPLE:

```
BigIron(config)# broadcast limit 30000
```

Syntax: broadcast limit <num>

Possible values: 0 – 4294967295; if you specify 0, limiting is disabled.

Default value: N/A

cam-partition

Specifies CAM partition percentages on the device. See the *Foundry Diagnostic Guide* for information on how to configure this feature.

EXAMPLE:

```
BigIron(config)# cam-partition l2 0 l3 100 l4 0
Slot 1 (DMA 0) CAM Partition:
    IronCore Module, Total Size 1Mbits
    L2 232.530029Mbits 88789.002929%, L3 0.75Mbits 75%, L4 232.655029Mbits 88801
.502929%
    L3 = 12288 (level2 = 2048, level3 = 2048), Pool0 = 2048, Pool1 = 2048, Pool2
 = 544488408, Pool3 = 0
Slot 1 (DMA 2) CAM Partition:
    IronCore Module, Total Size 1Mbits
    L2 232.530029Mbits 88789.002929%, L3 0.75Mbits 75%, L4 232.655029Mbits 88801
.502929%
    L3 = 12288 (level2 = 2048, level3 = 2048), Pool0 = 2048, Pool1 = 2048, Pool2
 = 544488408, Pool3 = 0
Cold start required.  Please write memory and then reload or power cycle.
```

Syntax: cam-partition l2 <percent> l3 <percent> l4 <percent> (router image)

Syntax: cam-partition l2 <percent> l4 <percent> (switch image)

NOTE: You must reload the software after enabling this protocol to place the change into effect.

Possible values: 1 – 100

Default value: See the *Foundry Diagnostic Guide* for the default CAM partitions.

cdp run

Enables a Foundry device to intercept and display Cisco Discovery Protocol (CDP) packets.

NOTE: When you enable interception of CDP packets, the Foundry device drops the packets. As a result, Cisco devices will no longer receive the packets.

EXAMPLE:

```
BigIron(config)# cdp run
```

Syntax: [no] cdp run

The feature is disabled by default.

Possible values: N/A

Default value: Disabled

chassis name

Assigns an administrative ID to the device.

NOTE: This command does not change the CLI prompt. To change the CLI prompt, use the **hostname** command. See “hostname” on page 6-32.

EXAMPLE:

```
BigIron(config)# chassis name routernyc
```

Syntax: chassis name <text>

Possible values: Up to 32 alphanumeric characters

Default value: Null string

chassis poll-time

Changes the number of seconds between polls of the power supply, fan, and temperature status.

Use the **show chassis** command to display the hardware status.

EXAMPLE:

To change the hardware poll time from 60 seconds (the default) to 30 seconds:

```
BigIron(config)# chassis poll-time 30
```

Syntax: chassis poll-time <num>

Possible values: 0 – 65535

Default value: 60

chassis trap-log

Disables or re-enables status polling for individual power supplies and fans. When you disable status polling, a fault in the power supply does not generate a trap in the system log.

EXAMPLE:

To disable polling of power supply 2, enter the following command:

```
BigIron(config)# no chassis trap-log ps2
```

Syntax: [no] chassis trap-log ps1 | ps2 | ps3 | ps4 | fan1 | fan2 | fan3 | fan4

Possible values: see above

To disable a fan failure trap or power supply trap, use one of the following values:

- ps1
- ps2
- ps3
- ps4
- fan1
- fan2
- fan3
- fan4

Default value: all traps enabled

clock summer-time

Causes daylight savings time to be automatically activated and deactivated for the relevant time zones.

EXAMPLE:

```
BigIron# clock summer-time
```

Syntax: clock summer-time

Possible values: N/A

Default value: N/A

clock timezone

Allows you to define the time zone of the clock. This parameter is used in conjunction with the **clock set** command or for timestamps obtained from an SNTP server. The **clock set**... command is configured at the privileged EXEC level of the CLI.

NOTE: Use this **clock** command before all others to ensure accuracy of the clock settings.

NOTE: For those time zones that recognize daylight savings time, the **clock summer-time** command will also need to be defined.

NOTE: Clock settings are not saved over power cycles; however, you can configure the system to reference an SNTP server at power up. This server will then automatically download the correct time reference for the network. The local device will then adjust the time according to its time zone setting. For more details on setting up an SNTP reference clock, refer to the **sntp** command at the privileged EXEC level and the **sntp poll-interval** and **sntp server** commands at the global CONFIG level.

EXAMPLE:

```
BigIron# clock timezone us eastern
```

Syntax: clock timezone gmt | us <timezone>

Possible values: The following time zones can be entered for US or GMT:

- US time zones: alaska, aleution, arizona, central, east-indiana, eastern, hawaii, michigan, mountain, pacific, samoa
- GMT time zones: gmt+12, gmt+11, gmt+10...fmt+01, gmt+00, gmt-01...gmt-10, gmt-11, gmt-12

Default value: pacific

confirm-port-up

Reduces the number of up-status confirmations the software requires before bringing a port up for use. This command is useful for network interface cards (NICs) that are designed to come up very quickly in certain applications and are sensitive to the slight delay caused by the Foundry ports as they wait for the multiple status indications before coming up. You can configure a Foundry device to reduce the number of status indications the software requires before bringing up a 10/100Base-Tx port.

NOTE: Do not use this command unless advised to do so by Foundry technical support.

By default, Foundry devices wait for multiple indications that a port is good before bringing the port up. Specific types of networking devices are sensitive to the very slight delay caused by the multiple status indications. In this case, you can use one of the following methods to reduce the number of status indications the software requires before bringing up a 10/100Base-Tx port.

For Chassis devices, you can set this parameter on individual ports. For Stackable devices, you can set the parameter globally for all 10/100 ports.

EXAMPLE:

By default, Chassis devices and the TurboIron/8 bring a 10/100 Base-Tx port up after receiving three consecutive up-status indications for the port. You can reduce this number to just one indication. To reduce the up-status indications required to bring up 10/100 ports 1/1 – 1/8 to just one, enter the following commands:

```
BigIron(config)# int ethernet 1/1 to 1/8

BigIron(config-mif-1/1-1/8)# confirm-port-up 1

BigIron(config-mif-1/1-1/8)# write memory
```

Syntax: [no] confirm-port-up <num>

The <num> parameter specifies the number of indications required by the software and can be from 1 – 10. The default for Chassis devices and the TurboIron/8 is 3.

EXAMPLE:

By default,Stackable devices bring a 10/100 Base-Tx port up after receiving ten consecutive up-status indications for the port. You can reduce this number to as few as one indication.

To reduce the up-status indications required to bring up 10/100 ports on a Stackable device, enter the following commands:

```
NetIron(config) confirm-port-up 1

NetIron(config) write memory
```

Syntax: [no] confirm-port-up <num>

The <num> parameter specifies the number of indications required by the software and can be from 1 – 10. The default for Stackable devices is 10.

Possible values: 1 – 10

Default value: 10 for Stackable devices; 3 for Chassis devices

console

Times out idle serial management sessions.

By default, a Foundry device does not time out serial CLI sessions. A serial session remains open indefinitely until you close it. You can configure the device to time out serial CLI sessions if they remain idle for a specified number of minutes. You can configure an idle timeout value from 0 – 240 minutes. The default is 0.

NOTE: If a session times out, the device does not close the connection. Instead, the CLI changes to the User EXEC mode (for example: BigIron>).

EXAMPLE:

To configure the idle timeout for serial CLI sessions, enter a command such as the following:

```
BigIron(config)# console timeout 20
```

This command configures the idle timeout value to 20 minutes.

Syntax: [no] console timeout <num>

The <num> parameter specifies the number of minutes the serial CLI session can remain idle before it times out. You can specify from 0 – 240 minutes. The default is 0 (sessions never time out).

Possible values: 0 – 240 minutes

Default value: 0 (sessions never time out)

crypto key

Configures a host RSA public and private key pair for SSH. The host RSA key pair is stored in the Foundry device's system-config file. Only the public key is readable. The host RSA key pair is used to negotiate a session key and encryption method with the SSH clients trying to connect to it.

EXAMPLE 1:

To generate a public and private host RSA key pair for the Foundry device:

```
BigIron(config)# crypto key generate rsa
BigIron(config)# wri mem
```

A host RSA key pair is stored in the system-config file, and SSH is enabled on the device.

EXAMPLE 2:

To delete the host RSA key pair from the system-config file:

```
BigIron(config)# crypto key zeroize rsa
BigIron(config)# wri mem
```


The host RSA key pair is deleted from the system-config file, and SSH is disabled on the device.

Syntax: crypto key generate | zeroize rsa

Possible values: N/A

Default value: N/A

crypto random-number-seed

Creates a new seed for generating a random number that is used for generating the dynamically created server RSA key pair for SSH.

EXAMPLE:

```
BigIron(config)# crypto random-number-seed generate
```

Syntax: crypto random-number-seed generate

Possible values: N/A

Default value: N/A

default-mtu

Globally increases the MTU size.

NOTE: This command applies only to JetCore Chassis devices only.

EXAMPLE:

To globally enable jumbo support on all ports, enter commands such as the following:

```
BigIron(config)# default-mtu 14336
BigIron(config)# write memory
BigIron(config)# end
BigIron# reload
```

Syntax: [no] default-mtu <num>

The <num> parameter specifies the maximum number of bytes an Ethernet frame can have in order to be forwarded on a port. You can specify from 64 – 14436. The default is 1518.

To enable jumbo support on an individual port, see “mtu” on page 7-59.

NOTE: You must save the configuration change and then reload the software to place the jumbo support into effect.

Possible values: 64 – 14436 bytes

Default value: 1518

default-vlan-id

When you enable port-based VLAN operation, all ports are assigned to VLAN 1 by default. As you create additional VLANs and assign ports to them, the ports are removed from the default VLAN. All ports that you do not assign to other VLANs remain members of default VLAN 1. This behavior ensures that all ports are always members of at least one VLAN.

You can change the VLAN ID for the default VLAN by entering the following command at the global CONFIG level of the CLI:

```
BigIron(config)# default-vlan-id 4095
```

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 – 4095.

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.

dhcp-gateway-list

This parameter must be defined when the feature, DHCP Assist, is enabled on a Foundry Layer 2 Switch. A gateway address must be defined for each sub-net that will be requesting addresses from a DHCP server. This allows the stamping process to occur. Each gateway address defined on the 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 switch inserts the addresses into the discovery packet in a round robin fashion.

Up to 32 gateway lists can be defined for each switch.

NOTE: For more details on this command and the DHCP Assist feature, see the "Configuring IP" chapter of the *Foundry Enterprise Configuration and Management Guide*.

EXAMPLE:

To define the sub-net address 192.95.5.1 as a gateway address and assign it to interface 4/11, enter the following:

```
FastIronII(config)# dhcp-gateway-list 1 192.95.5.1
FastIronII(config)# int e 4/11
FastIronII(config-if-4/11)# dhcp-gateway-list 1
```

Syntax: dhcp-gateway-list <num> <ip-addr>

Possible values: N/A

Default value: N/A

dot1x-enable

Enables 802.1X port security on a Foundry device and launches you into the dot1x configuration level.

EXAMPLE:

```
BigIron(config)# dot1x-enable
BigIron(config-dot1x)#
```

Syntax: [no] dot1x-enable

Possible values: N/A

Default value: disabled

enable

Three levels of passwords can be assigned to provide a range of access point for various users within the network.

The three levels are:

- Super user: This user has unlimited access to all levels of the CLI. This level is generally reserved for system administration. The super user is also the only user that can assign a password access level to another user.
- Configure Port: This user has the ability to configure interface parameters only. The user can also use the **show** commands.
- Read only: A user with this password level is able to use only the **show** commands. No configuration is allowed with this access type.

EXAMPLE:

```
BigIron(config)# enable super-user-password Larry
BigIron(config)# enable read-only-password Moe
```



```
BigIron(config)# enable port-config-password Curly
```

Syntax: enable super-user-password | read-only-password | port-config-password <text>

Possible values: Up to 32 alphanumeric characters can be assigned in the <text> field. The password cannot begin with a number.

Default value: No system default

enable aaa console

Configures the device to perform command authorization and command accounting for commands entered at the console.

EXAMPLE:

```
BigIron(config)# enable aaa console
```

Syntax: enable aaa console

NOTE: If you have previously configured the device to perform command authorization using a RADIUS server, entering the **enable aaa console** command may prevent the execution of any subsequent commands entered on the console.

This happens because RADIUS command authorization requires a list of allowable commands from the RADIUS server. This list is obtained during RADIUS authentication. For console sessions, RADIUS authentication is performed only if you have configured Enable authentication and specified RADIUS as the authentication method (for example, with the **aaa authentication enable default radius** command). If RADIUS authentication is never performed, the list of allowable commands is never obtained from the RADIUS server. Consequently, there would be no allowable commands on the console.

Possible values: N/A

Default value: N/A

enable password-display

Enables clear-text display of passwords and authentication strings in the output of some show commands:

- Enables display of SNMP community strings in the output of the **show snmp server** command
- Enables display of MD5 authentication strings for BGP4 neighbors and peer groups in the output of the **show ip bgp neighbors** command
- Enables display of passwords and MD5 authentication strings for OSPF virtual links in the output of the **show ip ospf virtual-links** command

---

NOTE: This command does not override encryption of passwords and authentication strings in the running-config and startup-config file.

EXAMPLE:

```
BigIron(config)# enable password-display
```

Syntax: enable password-display

Possible values: N/A

Default value: Disabled

enable password-min-length

Configures the device to require that Line (Telnet), Enable, and Local passwords be at least a specified length.

EXAMPLE:

```
BigIron(config)# enable password-min-length 8
```

Syntax: enable password-min-length <number-of-characters>

Possible values: 1 – 48

Default value: By default, the Foundry device imposes no minimum length on the Line, Enable, or Local passwords.

enable skip-page-display

Removes the stop page display characteristic for the **write terminal** command. For example, by default, when you enter the command **write terminal**, the full configuration file displayed will generally involve more than a single page display. You are prompted to press the Return key to view the next page of information. When this command is enabled, this page-by-page prompting will be removed and the entire display will roll on the screen until the end is reached.

To re-enable the stop page display characteristic, enter the **no enable skip-page-display** command.

EXAMPLE:

To remove the page-by-page display of configuration information, enter the following:

```
BigIron(config)# enable skip-page-display
```

Syntax: enable skip-page-display

Possible values: N/A

Default value: Disabled

enable snmp config-radius

Enables users of IronView or other SNMP management applications to configure RADIUS authentication parameters on the Foundry device.

EXAMPLE:

To enable IronView users to configure RADIUS authentication parameters on the Foundry device, enter the following:

```
BigIron(config)# enable snmp config-radius
```

Syntax: enable snmp config-radius

Possible values: N/A

Default value: Disabled

enable snmp config-tacacs

Enables users of IronView or other SNMP management applications to configure TACACS/TACACS+ authentication parameters on the Foundry device.

EXAMPLE:

To enable IronView users to configure TACACS/TACACS+ authentication parameters on the Foundry device, enter the following:

```
BigIron(config)# enable snmp config-tacacs
```

Syntax: enable snmp config-tacacs

Possible values: N/A

Default value: Disabled

enable telnet authentication

Allows you to use local access control, a RADIUS server, or a TACACS/TACACS+ server to authenticate telnet access to the device.

EXAMPLE:

```
BigIron(config)# enable telnet authentication
```

Syntax: [no] enable telnet authentication

Possible values: N/A

Default value: Disabled

enable telnet password

Allows you to assign a password for Telnet session access. To close a Telnet session, enter **logout**.

EXAMPLE:

```
BigIron(config)# enable telnet password secretsalso
```

Syntax: enable telnet password <text>

Possible values: Up to 32 alphanumeric characters can be assigned as the password.

Default value: No system default.

enable-acl-counter

Enables ACL accounting. Once accounting is enabled, you can disable it with the **no enable-acl-counter** command. Disabling and then re-enabling ACL accounting resets the counters to zero.

EXAMPLE:

```
BigIron(config)# enable-acl-counter
```

Syntax: [no] enable-acl-counter

Possible values: N/A

Default value: By default, ACL accounting is disabled.

end

Moves activity to the privileged EXEC level from any level of the CLI, with the exception of the user level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
BigIron(config)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the privileged level.

EXAMPLE:

To move from the global level, back to the privileged level, enter the following:

```
BigIron(config)# exit
BigIron#
```

Syntax: exit

Possible values: N/A

Default value: N/A

ext get config-file

Identifies the location of a MAC VLAN file. Use this command when configuring a Stackable Layer 2 Switch to load an external MAC VLAN list. See the "Configuring Virtual LANs (VLANs)" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

This command applies only to the FastIron Backbone Layer 2 Switch.

EXAMPLE:

To move from the global level, back to the privileged level, enter the following:

```
FastIron(config)# ext get config-file 209.157.22.3 MAC-VLAN.TXT
```

Syntax: ext get config-file <ip-addr> <external-file-name>

The <ip-addr> specifies the IP address of the TFTP server.

The <external-file-name> parameter specifies the name of the text file containing the **ext mac-vlan** commands.

Possible values: see above

Default value: N/A

fast port-span

Configures the Fast Port Span feature, which allows faster STP convergence on ports that are attached to end stations.

EXAMPLE:

To enable Fast Port Span:

```
BigIron(config)# fast port-span
```

EXAMPLE:

To exclude a port from Fast Port Span, while leaving Fast Port Span enabled globally:

```
BigIron(config)# fast port-span exclude ethernet 1/1
```

Syntax: [no] fast port-span [exclude ethernet <portnum> [ethernet <portnum>… | to <portnum>]

Possible values: Valid port numbers

Default value: Enabled

fast uplink-span

Configures the Fast Uplink Span feature, which reduces the convergence time for uplink ports to another device to just four seconds (two seconds for listening and two seconds for learning).

EXAMPLE:

To configure a group of ports for Fast Uplink Span, enter the following commands:

```
BigIron(config)# fast uplink-span ethernet 4/1 to 4/4
```

Syntax: [no] fast uplink-span [ethernet <portnum> [ethernet <portnum>… | to <portnum>]

Possible values: Ports that have redundant uplinks on a wiring closet switch.

Default value: Disabled

fdp holdtime

Changes the FDP hold time. By default, a Foundry device that receives an FDP update holds the information until one of the following events occurs:

- The device receives a new update.
- 180 seconds have passed since receipt of the last update. This is the hold time.

Once either of these events occurs, the device discards the update.

EXAMPLE:

```
BigIron(config)# fdp holdtime 360
```

Syntax: [no] fdp holdtime <secs>

The <secs> parameter specifies the number of seconds a Foundry device that receives an FDP update can hold the update before discarding it.

Possible values: 10 – 255 seconds

Default value: 180 seconds

fdp run

Enables a Foundry device to send Foundry Discovery Protocol (FDP) packets. The Foundry Discovery Protocol (FDP) enables Foundry devices to advertise themselves to other Foundry devices on the network.

EXAMPLE:

```
BigIron(config)# fdp run
```

Syntax: [no] fdp run

Possible values: N/A

Default value: Disabled

fdp timer

Changes the FDP update timer.

```
BigIron(config)# fdp timer 120
```

Syntax: [no] fdp timer <secs>

The <secs> parameter specifies the number of seconds between updates.

Possible values: 5 – 900 seconds

Default value: 60 seconds

flash <num>

Changes the block size for TFTP file transfers.

When you use TFTP to copy a file to or from a 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

EXAMPLE:

To change the block size for TFTP file transfers, enter a command such as the following at the global CONFIG level of the CLI:

```
BigIron(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.

Possible values: See above

Default value: 8192

flow-control

Allows you to turn flow control (802.3x) for full-duplex ports on or off (no). By default, flow control is on. To turn the feature off, enter the command **no flow-control**.

EXAMPLE:

```
BigIron(config)# no flow-control
```

To turn the feature back on later, enter the following command:

```
BigIron(config)# flow-control
```

Syntax: [no] flow-control

Possible values: N/A

Default value: on

gig-default

Changes the default negotiation mode for Gigabit ports on Chassis devices. You can configure the default Gigabit negotiation mode to be one of 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 for Chassis devices (including the TurboIron/8).
- 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.

See the "Configuring Basic Features" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide* for more information.

NOTE: This command does not apply to Stackable devices. To change the negotiation mode for a Stackable Gigabit Ethernet port, use the [**no**] **auto-gig** command at the Interface level. See “auto-gig” on page 7-10.

EXAMPLE:

To change the mode globally to negotiation-off, enter the following command:

```
BigIron(config)# gig-default neg-off
```

To override the global default on an individual Gigabit port, see “gig-default” on page 7-16.

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

Possible values: see above

Default value: neg-full-auto

global-protocol-vlan

The software places this command into the configuration the first time you configure a protocol VLAN. When you save the configuration to the startup-config file, the software places the command in the file.

NOTE: The protocol VLAN flag is not directly configurable. This command is used only by the software.

global-stp

Allows STP configuration using IronView or IronView Network Manager.

The software automatically places this command in the configuration when you select the option from within IronView or IronView Network Manager to allow STP configuration using the application. If you deselect the option to configure STP, the software removes the command from the configuration.

NOTE: This command applies only to configuring STP using IronView or IronView Network Manager. The command does not affect the ability to configure STP using the CLI or the Web management interface.

EXAMPLE:

Allow IronView or IronView Network Manager to manage this command.

Syntax: [no] global-stp

Possible values: see above

Default value: Disabled

gvrp-base-vlan-id

Changes the GVRP base VLAN ID.

By default, GVRP uses VLAN 4093 as a base VLAN for the protocol. All ports that are enabled for GVRP become tagged members of this VLAN. If you need to use VLAN ID 4093 for a statically configured VLAN, you can change the GVRP base VLAN ID.

NOTE: If you want to change the GVRP base VLAN ID, you must do so before enabling GVRP.

EXAMPLE:

```
BigIron(config)# gvrp-base-vlan-id 1001
```

This command changes the GVRP VLAN ID from 4093 to 1001.

Syntax: [no] gvrp-base-vlan-id <vlan-id>

The <vlan-id> parameter specifies the new VLAN ID. You can specify a VLAN ID from 2 – 4092 or 4095.

Possible values: 2 – 4092 or 4095

Default value: 4093

gvrp-enable

Enables GVRP and changes the CLI to the GVRP configuration level.

EXAMPLE:

```
BigIron(config)# gvrp-enable
BigIron(config-gvrp)#
```

Syntax: [no] gvrp-enable

For information about the commands at the GVRP configuration level, see “GVRP Commands” on page 38-1.

Possible values: N/A

Default value: Disabled

gvrp-max-leaveall-timer

Increases the maximum value you can specify for the GVRP Leaveall timer.

By default, the highest value you can specify for the Leaveall timer is 300000 ms. You can increase the maximum configurable value of the Leaveall timer to 1000000 ms.

NOTE: You must enter this command before enabling GVRP. Once GVRP is enabled, you cannot change the maximum Leaveall timer value.

NOTE: This command does not change the default value of the Leaveall timer itself. The command only changes the maximum value to which you can set the Leaveall timer.

EXAMPLE:

```
BigIron(config)# gvrp-max-leaveall-timer 1000000
```

Syntax: [no] gvrp-max-leaveall-timer <ms>

The <ms> parameter specifies the maximum number of ms to which you can set the Leaveall timer. You can specify from 300000 – 1000000 (one million) ms. The value must be a multiple of 100 ms. The default is 300000 ms.

Possible values: 300000 – 1000000 (one million) ms

Default value: 300000 ms

hostname

Changes the hostname field to more easily identify Foundry devices within the network.

EXAMPLE:

To change the hostname to “Router1” from the default, “BigIron”, enter the following:

```
BigIron(config)# hostname Router1
Router1(config)#
```

Syntax: hostname <text>

Possible values: Up to 32 alphanumeric characters can be assigned to hostname text string.

Default value: The product name

hw-drop-acl-denied-packet

Enables hardware filtering of packets denied by ACLs.

By default, packets denied by ACLs are filtered by the CPU. You can enable the device to create Content Addressable Memory (CAM) entries for packets denied by ACLs. This causes the filtering to occur in hardware instead of in the CPU.

When you enable hardware filtering of denied packets, the first time the device filters a packet denied by an ACL, the device sends the packet to the CPU for processing. The CPU also creates a CAM entry for the denied packet. Subsequent packets with the same address information are filtered using the CAM entry. The CAM entry ages out after two minutes if not used.

EXAMPLE:

```
BigIron(config)# hw-drop-acl-denied-packet
```

Syntax: [no] hw-drop-acl-denied-packet

Possible values: N/A

Default value: Disabled

hw-module

Specifies a hardware module to which to change the CAM partition. See the *Foundry Diagnostic Guide* for more information on CAM partitioning.

EXAMPLE:

```
BigIron(config)# hw-module 3
BigIron(config-module-3/8)#
```


Syntax: hw-module <module>

Possible values: Module number

Default value: N/A

interface

Accesses the interface CONFIG level of the CLI. You can define a physical interface, loopback interface, virtual interface (ve), Asynchronous Transfer Mode (ATM) interface, or Packet Over SONET (POS) interface at the Interface level.

- For information about ATM interfaces, see the "Using Asynchronous Transfer Mode Modules" chapter in the *Foundry Switch and Router Installation and Basic Configuration Guide*.
- For information about POS interfaces, see the "Using Packet Over SONET Modules" chapter in the *Foundry Switch and Router Installation and Basic Configuration Guide*.

By default, you can add up to 24 IP addresses to a physical, virtual, or loopback interface.

You can configure up to 64 virtual interfaces on a Stackable Layer 3 Switch and up to 255 virtual interfaces on a Chassis Layer 3 Switch.

You can configure up to four loopback interfaces on a Stackable Layer 3 Switch and up to eight loopback interfaces on a Chassis Layer 3 Switch.

NOTE: You also can increase the total number of IP sub-net interfaces that you can configure on a Layer 3 Switch. See “system-max” on page 6-131.

EXAMPLE:

To change the configuration for port 1 on a Stackable device, enter the following:

```
FastIron(config)# inter e 1
FastIron(config-if-1)#
```

NOTE: To change the port for a Chassis device, you also need to enter the slot number of the module on which the port resides.

Syntax: [no] interface atm <slot>/<port>.<subif> [multipoint | point-to-point]

Syntax: interface ethernet <portnum> [to <portnum>]

Syntax: interface loopback <num>

Syntax: interface pos <slot>/<port>

Syntax: interface ve <num>

EXAMPLE:

To add a virtual interface to a Layer 3 Switch, enter the following. Use commands at the Virtual Interface level (vif) to configure the interface.

```
BigIron(config)# inter ve 1
BigIron(config-vif-1)#
```

Syntax: interface ve <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.

Possible values: See above

Default value: N/A

EXAMPLE:

To add a loopback interface to a Layer 3 Switch, enter the following:

```
BigIron(config)# int loopback 1
BigIron(config-lbif-1)# ip address 10.0.0.1/24
```

Syntax: interface loopback <num>

Possible values: 1 – 4 (Stackable devices); 1 – 15 (Chassis devices)

Default value: N/A

NOTE: For information about the commands you can enter at the interface configuration level, see “Interface Commands” on page 7-1.

EXAMPLE:

To configure a point-to-point ATM interface, enter commands such as the following:

```
BigIron(config)# interface atm 3/1.1
BigIron(config-subif-3/1.1)# atm pvc 1 200 cbr 10000
```

To configure a point-to-multipoint ATM interface, enter commands such as the following:

```
BigIron(config)# interface atm 4/1 multipoint
BigIron(config-subif-4/1.1)# atm pvc 1 1 cbr 10000 ip inarp
BigIron(config-subif-4/1.1)# atm pvc 1 2 cbr 20000 ip inarp
BigIron(config-subif-4/1.1)# atm pvc 1 3 cbr 25000 ip inarp
BigIron(config-subif-4/1.1)# ip address 10.10.10.4
```

The first command configures a point-to-multipoint ATM interface. The remaining commands configure PVCs on the interface. In this example, all three PVCs use Inverse ARP to resolve the remote IP addresses.

Syntax: [no] interface atm <slot>/<port>.<subif> [multipoint | point-to-point]

For information about the commands at the subinterface level, see “atm pvc” on page 7-6.

Possible values: See above.

Default value: None configured

interface group-ve

Begins configuration of a virtual interface group. A virtual interface group allows you to configure virtual interface attributes one time, then apply the attributes to multiple virtual interfaces.

NOTE: This feature applies only to VLAN groups. See the "Configuring Virtual LANs (VLANs)" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

EXAMPLE:

To configure a virtual interface group, enter commands such as the following:

```
BigIron(config)# vlan-group 1
BigIron(config-vlan-group-1)# group-router-interface
BigIron(config-vlan-group-1)# exit
BigIron(config)# interface group-ve 1
BigIron(config-vif-group-1)# ip address 10.10.10.1/24
```

These commands enable VLAN group 1 to have a group virtual interface, then configure virtual interface group 1. The software always associates a virtual 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 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 interface group. Enter this command at the configuration level for the VLAN group. This command configures the VLAN group to use the virtual 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 interface group. The VLAN group must already be configured and enabled to use a virtual interface group. The software automatically associates the virtual interface group with the VLAN group that has the same ID. You can associate a virtual 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.

Possible values: See above

Default value: N/A

interface link-hold-down

Delays initialization of the device's ports following a software reload.

By default, the software brings up the ports on a Foundry device as soon as the software has fully finished booting. Some devices attached to the Foundry device might require more time to properly initialize and establish a link with the Foundry device.

NOTE: This feature does not apply to Layer 2 Switches.

In this case, you can configure the software to delay bringing up the device's ports for an additional number of milliseconds, up to 100 (one second).

NOTE: The actual amount of time it takes to bring a port up is slightly longer than the hold-down time. After fully booting the software, the device initializes the ports, which takes an additional few seconds.

EXAMPLE:

To delay port initialization on a Foundry device, enter a command such as the following at the global CONFIG level for the port:

```
BigIron(config)# interface link-hold-down 50
```

This command delays initialization of the device's ports for 50 milliseconds (one half second) following completion of a software reload.

Syntax: [no] interface link-hold-down <msecs>

The <msecs> parameter specifies the number of milliseconds to wait before initializing the ports. You can specify from 0 – 100. The default is 0.

Possible values: See above

Default value: Disabled

ip access-list

Configures a named IP ACL.

You can use this command to configure a standard or extended IP ACL.

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.

EXAMPLE:

To configure a named standard ACL entry, enter commands such as the following.

```
BigIron(config)# ip access-list standard Net1
BigIron(config-std-nac1)# deny host 209.157.22.26 log
BigIron(config-std-nac1)# deny 209.157.29.12 log
BigIron(config-std-nac1)# deny host IPHost1 log
BigIron(config-std-nac1)# permit any
BigIron(config-std-nac1)# exit
BigIron(config)# int eth 1/1
BigIron(config-if-1/1)# ip access-group Net1 out
```

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 1/1. 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 the “Configuring Standard ACLs“ section of the “Using Access Control Lists (ACLs)” chapter in the *Foundry Enterprise Configuration and Management Guide*.

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 are configuring a named ACL.

Syntax: ip access-list extended | standard <string> | <num>

The **extended** | **standard** parameter indicates the ACL type.

The <string> parameter is the ACL 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 <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 or 100 – 199 for extended 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 options at the ACL configuration level and the syntax for the **ip access-group** command are the same for numbered and named ACLs. See “access-list (standard)” on page 6-3.

EXAMPLE:

To configure a named extended ACL entry, enter commands such as the following.

```
BigIron(config)# ip access-list extended “block Telnet”
BigIron(config-ext-nac1)# deny tcp host 209.157.22.26 any eq telnet log
BigIron(config-ext-nac1)# permit ip any any
BigIron(config-ext-nac1)# exit
BigIron(config)# int eth 1/1
BigIron(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. See "access-list (extended)" on page 6-5.

Possible values: see above

Default value: N/A

ip access-list disable-log-to-cpu

Globally disables ACL logging.

This command is useful for JetCore devices. Hardware-based ACLs do not support the **log** option. Even when hardware-based ACLs are enabled, if an ACL entry has the **log** option, traffic that matches that ACL is sent to the CPU for processing.

If your configuration already contains ACLs that you want to use with hardware-based ACLs, but some of the ACLs contain the **log** option, you can globally disable ACL logging without the need to remove the **log** option from each ACL entry. When you globally disable ACL logging, the ACL entries remain unchanged but the **log** option is ignored and the ACL can use the hardware-based ACL mode.

EXAMPLE:

```
BigIron(config)# ip access-list disable-log-to-cpu
```

Syntax: [no] ip access-list disable-log-to-cpu

To re-enable ACL logging, enter the following command:

```
BigIron(config)# no ip access-list disable-log-to-cpu
```

Possible values: N/A

Default value: ACL logging is enabled

ip access-list frag-rate-on-interface

Sets the fragment threshold for JetCore rule-based ACLs, for individual interfaces. If an individual interface receives more than the specified maximum number of fragments, the device takes the exceed action you specify.

The device can send to the CPU only the number of fragments you specify per second, regardless of which interfaces the fragments come in on. If the threshold is exceeded, the device takes the exceed action you specify.

By default, when you enable CPU filtering of packet fragments, all fragments are sent to the CPU.

NOTE: This command applies only to JetCore devices.

EXAMPLE:

```
BigIron(config)# ip access-list frag-rate-on-interface 5000 exceed-action forward
reset-interval 5
```

This command sets the fragment threshold at 5,000 for individual interfaces. If any interface on the device receives more than 5,000 fragments in a one-second interval, the device takes the specified action. In this case, the action is to forward the fragments in hardware without filtering them. The device continues forwarding fragments in hardware for five minutes before beginning to send fragments to the CPU again.

Syntax: [no] ip access-list frag-rate-on-interface <num> exceed-action drop | forward reset-interval <mins>

The <num> parameter specifies the maximum number of fragments the device or an individual interface can receive and send to the CPU in a one-second interval.

The <num> parameter specifies the maximum number of fragments per second. You can specify from 300 – 8000.

The **drop | forward** parameter specifies the action to take if the threshold (<num> parameter) is exceeded:

- **drop** – fragments are dropped without filtering by the ACLs
- **forward** – fragments are forwarded in hardware without filtering by the ACLs

The <mins> parameter specifies the number of minutes the device will enforce the drop or forward action after a threshold has been exceeded. You can specify from 1 – 30 minutes.

Possible values: see above

Default value: see above

ip access-list frag-rate-on-system

Sets the fragment threshold for JetCore rule-based ACLs, for the entire device. The device can send to the CPU only the number of fragments you specify per second, regardless of which interfaces the fragments come in on. If the threshold is exceeded, the device takes the exceed action you specify.

By default, when you enable CPU filtering of packet fragments, all fragments are sent to the CPU.

NOTE: This command applies only to JetCore devices.

EXAMPLE:

```
BigIron(config)# ip access-list frag-rate-on-system 15000 exceed-action drop reset-
interval 10
```

This command sets the fragment threshold at 15,000 per second, for the entire device. If the device receives more than 15,000 packet fragments in a one-second interval, the device takes the specified action. The action specified with this command is to drop the excess fragments and continue dropping fragments for a holddown time of ten minutes. After the ten minutes have passed, the device starts sending fragments to the CPU again for processing.

Syntax: [no] ip access-list frag-rate-on-system <num> exceed-action drop | forward reset-interval <mins>

The <num> parameter specifies the maximum number of fragments the device or an individual interface can receive and send to the CPU in a one-second interval.

The <num> parameter specifies the maximum number of fragments per second. You can specify from 600 – 12800.

The **drop | forward** parameter specifies the action to take if the threshold (<num> parameter) is exceeded:

- **drop** – fragments are dropped without filtering by the ACLs
- **forward** – fragments are forwarded in hardware without filtering by the ACLs

The <mins> parameter specifies the number of minutes the device will enforce the drop or forward action after a threshold has been exceeded. You can specify from 1 – 30 minutes.

Possible values: see above

Default value: see above

ip access-list logging-age

Configures the Layer 4 session log timer, which is used for keeping track of packets explicitly denied by an ACL. In releases prior to 07.6.03, the timer interval was set to 5 minutes and was not configurable.

The timer keeps track of all packets explicitly denied by the ACL entries. When the timer expires, the software generates a single Syslog entry for each ACL entry that has denied a packet. The message indicates the number of packets denied by the ACL entry from the time that the timer was started. If no ACL entries explicitly permit or deny packets during an entire timer interval, the timer stops. The timer restarts when an ACL entry explicitly permits or denies a packet.

To store information about denied packets during the timer interval, the device makes entries in its Layer 4 session table. If a large number of packets are denied by the ACL during the timer interval, it can consume a large portion of the device's Layer 4 resources. To prevent this from happening, starting in release 07.6.03, you can configure the timer interval to be a shorter length of time.

EXAMPLE:

For example, to set the timer interval to 2 minutes, enter the following command:


```
BigIron(config)# ip access-list logging-age 2
```

Syntax: ip access-list logging-age <minutes>

Possible values: You can set the timer to between 1 and 10 minutes.

Default: The default is 5 minutes.

ip access-policy

Configures permit and deny policies and Layer 4 QoS policies on Layer 2 Switches and Layer 3 Switches. See the "Policies and Filters" appendix of the *Foundry Enterprise Configuration and Management Guide* for more information.

NOTE: Access policies on Layer 3 Switches can permit or deny packets (filter) or allocate packets to specific QoS levels. Access policies on Layer 2 Switches can only allocate traffic to specific QoS levels.

NOTE: After you configure an IP access policy, you need to apply it to specific ports using the **ip access-policy-group** command at the Interface level of the CLI. See “ip access-policy-group” on page 7-21.

Permit and Deny Policies

IP access policies are rules that determine whether the device forwards or drops IP packets. You create an IP access policy by defining an IP filter, then applying it to an interface. The filter consists of source and destination IP information and the action to take when a packet matches the values in the filter. You can configure an IP filter to permit (forward) or deny (drop) the packet.

NOTE: You can configure permit and deny IP access policies only on Layer 3 Switches, not on Layer 2 Switches. On Layer 2 Switches, the **ip access-policy** command configures Layer 4 QoS.

You can apply an IP filter to inbound or outbound packets. When you apply the filter to an interface, you specify whether the filter applies to inbound packets or outbound packets. Thus, you can use the same filter on multiple interfaces and specify the filter direction independently on each interface.

EXAMPLE:

To configure an IP access policy to explicitly permit HTTP traffic (TCP port 80) from IP address 10.0.0.1 on port 1/2, enter the following commands:

```
BigIron(config)# ip access-policy 2 permit 10.0.0.1 255.0.0.0 tcp eq 80
BigIron(config)# int e 1/2
BigIron(config-if-1/2)# ip access-policy-group in 2
```

Here is the syntax for Chassis devices.

Syntax: ip access-policy <num> deny | permit <ip-addr> <mask> | any <ip-addr> <ip-mask> | any icmp | igmp | igrp | ospf | tcp | udp | <num> [<operator> [<tcp/udp-port-num>]] [log]

Syntax: ip access-policy-group in | out <policy-list>

Here is the syntax for Stackable devices.

Syntax: ip access-policy <num> deny | permit <ip-addr> <mask> | any <ip-addr> <ip-mask> | any tcp | udp [<operator> [<tcp/udp-port-num>]] [log]

Syntax: ip access-policy-group in | out <policy-list>

The <num> parameter is the policy number.

The **deny** | **permit** parameter specifies the action the router takes if a packet matches the policy.

- If you specify **deny**, the router drops the packet.
- If you specify **permit**, the router forwards the packet.

The <ip-addr> <ip-mask> | **any** <ip-addr> <ip-mask> | **any** parameters specify the source and destination IP addresses. If you specify a particular IP address, you also need to specify the mask for that address. If you specify **any** to apply the policy to all source or destination addresses, you do not need to specify **any** again for the mask. Make sure you specify a separate address and mask or **any** for the source and destination address.

The **icmp** | **igmp** | **igrp** | **ospf** | **tcp** | **udp** | <num> parameter specifies the Layer 4 port to which you are applying the policy. If you specify **tcp** or **udp**, you also can use the optional <operator> and <tcp/udp-port-num> parameters to fine-tune the policy to apply to specific TCP or UDP ports.

The <operator> parameter applies only if you use the tcp or udp parameter above. Use the <operator> parameter to specify the comparison condition for the specific TCP or UDP ports. For example, if you are configuring QoS 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**.
- **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.
- **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.

The **log** parameter applies only to deny policies. This parameter generates a Syslog entry for packets that are denied by the policy. See Example 4 in “show logging” on page 43-93.

Layer 4 Policies

EXAMPLE:

To assign a priority of 4 to all HTTP traffic on port 3/12 on a BigIron Layer 3 Switch, enter the following:

```
BigIron(config)# ip access-policy 1 priority 4 any any tcp eq http

BigIron(config)# int e 3/12

BigIron(config-if-3/12)# ip access-policy-group out 1
```

Here is the syntax for Chassis devices, the FastIron 4802, and the TurboIron/8.

Syntax: ip access-policy <num> priority <0-7> <ip-addr> <ip-mask> | any <ip-addr> <ip-mask> | any tcp | udp [<operator> [<tcp/udp-port-num>]]

Syntax: ip access-policy-group in | out <policy-list>

Here is the syntax for Stackable devices.

Syntax: ip access-policy <num> high | normal <ip-addr> <ip-mask> | any <ip-addr> <ip-mask> | any tcp | udp [<operator> [<tcp/udp-port-num>]]

Syntax: ip access-policy-group in | out <policy-list>

The <num> parameter is the policy number.

The **priority** <0-7> and **high** | **normal** parameters specify the QoS priority level. The defaults are 0 (normal priority) and **normal**. The highest priority on a Chassis device, the FastIron 4802, or the TurboIron/8 is 7.

The <ip-addr> <ip-mask> | **any** <ip-addr> <ip-mask> | **any** parameters specify the source and destination IP addresses. If you specify a particular IP address, you also need to specify the mask for that address. If you specify any to apply the policy to all source or destination addresses, you do not need to specify any again for the mask. Make sure you specify a separate address and mask or any for the source and destination address.

The **icmp** | **igmp** | **igrp** | **ospf** | **tcp** | **udp** | <num> parameter specifies the Layer 4 port to which you are applying the policy. If you specify tcp or udp, you also can use the optional <operator> and <tcp/udp-port-num> parameters to fine-tune the policy to apply to specific TCP or UDP ports.

The <operator> parameter applies only if you use the tcp or udp parameter above. Use the <operator> parameter to specify the comparison condition for the specific TCP or UDP ports. For example, if you are configuring QoS 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**.
- **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.
- **established** – This operator applies only to TCP packets. If you use this operator, the QoS 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.

ip address

Assigns an IP address and network mask to a Layer 2 Switch to support Telnet and SNMP management.

EXAMPLE:

```
FastIron(config)# ip address 192.22.3.44 255.255.255.0
```

```
FastIron(config)# ip address 192.22.3.44/24
```

Syntax: ip address <ip-addr> <ip-mask>

or

Syntax: ip address <ip-addr>/<mask-bits>

Possible values: N/A

Default value: N/A

ip arp-age

Defines how long an ARP entry will be resident in the ARP cache before it is aged out.

EXAMPLE:

```
BigIron(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.

Possible values: 0 – 240 minutes

Default value: 10 minutes

ip as-path

Configures an AS-path ACL. You can use AS-path ACLs to permit or deny routes based on their AS path information.

EXAMPLE:

To configure an AS-path list that uses ACL 1, enter a command such as the following:

```
BigIron(config)# ip as-path access-list 1 permit 100
BigIron(config)# router bgp
BigIron(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 "match" on page 19-1.

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 the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

The **neighbor** command uses the **filter-list** parameter to apply the AS-path ACL to the neighbor. See "neighbor" on page 13-12.

Possible values: see above

Default value: N/A

ip broadcast-zero

Enables or disables 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, but does not treat packets with all zeros in the host portion as IP sub-net broadcasts.

NOTE: When you enable the Layer 3 Switch for zero-based sub-net broadcasts, it 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.

EXAMPLE:

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.

```
BigIron(config)# ip broadcast-zero
```

Syntax: [no] ip broadcast-zero

Possible values: enabled or disabled

Default value: disabled

ip community-list

Configures a community ACL. You can use community ACLs to permit or deny routes based on their communities.

EXAMPLE:

To configure community ACL 1, enter a command such as the following:

```
BigIron(config)# ip community-list 1 permit 123:2
```

This command configures a community ACL that permits routes that contain community 123:2.

NOTE: See “match” on page 19-1 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.

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 is a regular expression. For syntax information for the regular expressions, see the “Using Regular Expressions” section of the “Configuring BGP4” chapter in the *Foundry Enterprise Configuration and Management Guide*. You can specify a regular expression only in an extended community ACL.

Possible values: see above

Default value: N/A

ip default-gateway

Assigns an IP address and mask to a Layer 2 Switch to support Telnet and SNMP management.

This command is not available on Foundry Layer 3 Switches.

EXAMPLE:

```
FastIron(config)# ip default-gateway 192.22.33.100
```


Syntax: ip default-gateway <ip-addr>

Possible values: N/A

Default value: N/A

ip default-network

Configures a default network route, use one of the following methods. You can configure up to four default network routes.

EXAMPLE:

To configure a default network route, enter commands such as the following:

```
BigIron(config)# ip default-network 209.157.22.0
BigIron(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:

```
BigIron(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
       209.157.20.0      255.255.255.0     0.0.0.0           lb1    1      D
       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.

Possible values: valid IP network address

Default value: N/A

ip directed-broadcast

Enables or disables forwarding of directed IP broadcasts on a Layer 3 Switch.

EXAMPLE:

```
BigIron(config)# ip directed-broadcast
```

Syntax: [no] ip directed-broadcast

Possible values: N/A

Default value: disabled

ip dns domain-name

Defines a domain name for a range of addresses on the Foundry Layer 2 Switch or Layer 3 Switch. This eliminates the need to type in the domain name. It will automatically be appended to the hostname.

EXAMPLE:

```
BigIron(config)# ip dns domain-name newyork.com
```

Syntax: ip dns domain-name

Possible values: N/A

Default value: N/A

ip dns server-address

Up to four DNS servers can be defined for each DNS entry. The first entry serves as the primary default address (207.95.6.199). If a query to the primary address fails to be resolved after three attempts, the next gateway address will be queried for three times as well. This process will continue for each defined gateway address until a query is resolved. The order in which the default gateway addresses are polled is tied to the order in which they are entered when initially defined as shown in the example.

EXAMPLE:

```
BigIron(config)# ip dns server-address 207.95.6.199 205.96.7.1 5 208.95.7.25
201.98.7.15
```

Syntax: ip dns server-address <ip-addr> [<ip-addr>] [<ip-addr>] [<ip-addr>]

Possible values: Up to four IP addresses

Default value: N/A

ip dont-use-acl

Disables all packet-forwarding IP ACLs (those associated with specific ports) and also prevents you from associating an IP ACL with a port. However, the command does not remove existing IP ACLs from the startup-config file. In addition, the command does not affect IP ACLs used for controlling management access to the device.

NOTE: A Layer 3 Switch cannot actively use both IP access policies and IP ACLs for filtering IP traffic. When you boot a Layer 3 Switch with software release 06.5.00 or higher, the software checks the device's startup-config file for **ip access-policy-group** commands, which associate IP access policies with ports. If the software finds an **ip access-policy-group** command in the file, the software disables all packet-forwarding IP ACLs (those associated with specific ports) and also prevents you from applying an IP ACL to a port.

The next time you save the startup-config file, the software adds the **ip dont-use-acl** command near the top of the file, underneath the ver (software version) statement.

EXAMPLE:

Disabling ACL Mode

If the ACL mode is enabled, a message is displayed when you try to apply an IP access policy to a port, as shown in the following CLI example:

```
BigIron(config-if-e1000-1/1)# ip access-policy-group 1 in
Must disable ACL mode first by using ip dont-use-acl command, write memory and
reload
```

To disable the ACL mode, enter the following commands:

```
BigIron(config-if-e1000-1/1)# exit
BigIron(config)# ip dont-use-acl
BigIron(config)# write memory
BigIron(config)# end
BigIron# reload
```

EXAMPLE:

Enabling ACL Mode

If you try to apply an IP ACL to a port when the ACL mode is disabled (when the **ip dont-use-acl** command is in effect), a message is displayed, as shown in the following CLI example:

```
BigIron(config-if-e1000-1/1)# ip access-group 1 out
Must enable ACL mode first by using no ip dont-use-acl command and removing all ip
access-policy-group commands from interfaces, write memory and reload
```

As the message states, if you want to use IP ACLs, you must first enable the ACL mode. To do so, use either of the following methods.

To enable the ACL mode, enter the following commands:

```
BigIron(config-if-e1000-1/1)# exit
BigIron(config)# no ip dont-use-acl
BigIron(config)# write memory
BigIron(config)# end
BigIron# reload
```

The write memory command removes the **ip dont-use-acl** command from the startup-config file. The reload command reloads the software. When the software finishes loading, you can apply IP ACLs to ports.

The commands that configure the IP access policies and apply them to ports remain in the startup-config file in case you want to use them again, but they are disabled. If you later decide you want to use the IP access policies again instead of IP ACLs, you must disable the IP ACL mode again. See Example 1 above.

Syntax: [no] ip dont-use-acl

Possible values: N/A

Default value: see above

ip dr-aggregate

Optimizes the CAM for devices that have few explicit routes (about 30 or fewer) and use the default route for most of the traffic.

Without CAM default route aggregation, the device programs a CAM entry for each destination that uses an explicit route in the route table and also programs a separate CAM entry for each destination that uses the default route. For example, suppose the IP route table contains two explicit routes, 20.0.0.x and 30.0.0.x and uses the default route for all other destinations. When the device needs to forward traffic to 20.0.0.x, the device uses the existing CAM entry for the destination. If this is the first time the device is forwarding traffic to the destination and the CAM entry therefore hasn't been programmed yet, the device programs the entry for 20.0.0.x.

The same process occurs for traffic destined to a network that doesn't have an explicit route in the IP route table. When the device needs to forward traffic to a destination that requires the default route, the device creates a CAM entry for the destination network. For example, if the device needs to forward traffic to 40.40.40.x and 40.41.41.x, the device creates two CAM entries, one for 40.40.40.x and another for 40.41.41.x.

When the device needs to forward traffic on the default route, the device attempts to build an aggregate route that does not conflict with an explicit route in the IP route table. (A conflict occurs if an explicit host route in the table overlaps with the aggregate.)

For example, with CAM default route aggregation enabled, the device creates a single CAM entry, 40.0.0.0/8, for 40.40.40.x and 40.41.41.x. In fact, traffic for any network that overlaps with 40.0.0.0/8 uses the same CAM entry.

The device begins with a /8 aggregate.

- If there are no conflicts with explicit routes, the device programs the /8 aggregate into the CAM.
- If there is a conflict, the device tries a /12 aggregate, and so on in increments of 4 (/16, /20, /24, and so on) until a non-conflicting entry can be programmed into the CAM.

NOTE: CAM default route aggregation requires a default route in the IP route table.

Compare with “ip net-aggregate” on page 6-63.

EXAMPLE:

```
BigIron(config)# ip dr-aggregate
```

Syntax: [no] ip dr-aggregate

Possible values: N/A

Default value: Disabled

ip flow-aggregation cache

Configures an aggregate cache for NetFlow.

EXAMPLE:

```
BigIron(config)# ip flow-aggregation cache as
BigIron(config-flow-cache_as)#
```

This command begins configuration for the AS aggregate cache. Notice that the CLI prompt changes to the configuration level for the aggregate cache. At the aggregate cache's configuration level, you can change cache parameters. Here is an example:

```
BigIron(config)# ip flow-aggregation cache as
BigIron(config-flow-cache_as)# cache entries 2046
BigIron(config-flow-cache_as)# cache timeout inactive 200
BigIron(config-flow-cache_as)# cache timeout active 45
BigIron(config-flow-cache_as)# export destination 10.42.42.1 9992
BigIron(config-flow-cache_as)# enabled
```

The **cache** commands change cache parameters. The **enable** command enables the cache. A cache does not go into effect until you enable it.

Syntax: [no] ip flow-aggregation cache as | destination-prefix | prefix | protocol-port | source-prefix

The **as | destination-prefix | prefix | protocol-port | source-prefix** parameter specifies the cache type.

- **as** – Configures an AS cache. Flows are aggregated based on AS number.
- **destination-prefix** – Configures a destination prefix cache. Flows are aggregated based on destination network prefix.
- **prefix** – Configures a prefix cache. Flows are aggregated based on both source and destination network prefixes.
- **protocol-port** – Configures a protocol port cache. Flows are aggregated based on source and destination IP protocol port.
- **source-prefix** – Configures a source prefix cache. Flows are aggregated based on source network prefix.

The following command specifies the collector. You can specify one collector for an aggregate cache.

Syntax: [no] export destination <ip-addr> <udp-portnum>

The following command specifies the maximum number of flows the cache can contain.

Syntax: [no] cache entries <num>

The <num> parameter specifies the maximum number of flows and can be from 1024 – 524288. The default is 4096.

The following commands specify the active and inactive timeouts.

Syntax: [no] cache timeout inactive <secs>

Syntax: [no] cache timeout active <mins>

The **inactive** <secs> parameter specifies the maximum number of seconds an inactive flow can remain in the cache.

The **active** <mins> parameter specifies the maximum number of minutes an active flow can remain in the cache.

The following command enables the cache.

Syntax: [no] enabled

NOTE: The **enabled** command is required to enable the cache. The **cache** commands are optional.

Possible values: See above

Default value: None configured

ip flow-cache timeout

Changes an age timer for NetFlow.

NetFlow uses the following age timers to age flows out of the cache for export.

- Inactive – The inactive timer ages out a flow after it has been unused for the specified number of seconds.
- Active – The active timer ages out a flow that is in use if the flow has remained in use continuously for the specified number of minutes.

NOTE: In addition to using these timers, NetFlow also ages out normally terminated TCP flows, and ages out flows when the cache becomes full.

NOTE: The main flow cache and the aggregate caches (if you configure them) use separate timeouts. Changing the main flow cache's timeouts does not affect the timeouts for the aggregate caches.

EXAMPLE:

```
BigIron(config)# ip flow-cache timeout active 45
```

This command changes the maximum age for active flows to 45 minutes.

Syntax: [no] ip flow-cache timeout active <mins> | inactive <secs>

The **active** <mins> parameter specifies the maximum number of minutes an active flow can remain in the cache. You can specify from 1 – 60 minutes. The default is 30 minutes.

The **inactive** <secs> parameter specifies the maximum number of seconds an inactive flow can remain in the cache. You can specify from 10 – 600 seconds. The default is 60 seconds.

Possible values: See above

Default value: See above

ip flow-export destination

Specifies a data export collector for NetFlow.

EXAMPLE:

```
BigIron(config)# ip flow-export destination 10.10.10.1 8080 1
```

Syntax: [no] ip flow-export destination <ip-addr> <udp-portnum> [<collector-id>]

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

The <udp-portnum> specifies the UDP port on the collector that listens for the exported flow packets.

The <collector-id> is a number from 1 – 10. This number applies only to the Foundry device and is not related to configuration information on the collector itself.

- If you are specifying more than one collector, make sure you also specify the collector ID and use a different ID for each collector.
- If you are specifying only one collector, you do not need to specify the ID. In this case, the software automatically assigns ID 1 to the collector.

NOTE: If you do not specify the collector ID, the software always uses ID 1. If you already have added a collector whose ID is 1, and you add another collector with ID 1, the software replaces the older collector with the new collector.

Possible values: See above

Default value: See above

ip flow-export enable

Enables NetFlow.

EXAMPLE:

```
BigIron(config)# ip flow-export enable
```

Syntax: [no] ip flow-export enable

NOTE: This command enables the feature globally. However, to begin flow collection and export, you must enable collection for individual interfaces. NetFlow collects and exports flows only for the interfaces on which you enable the feature.

NOTE: If you plan to use aggregate caches instead, you do not need to globally enable NetFlow or specify collector information. Instead, you perform this configuration as part of the aggregate cache configuration.

Possible values: N/A

Default value: Disabled

ip flow-export origin-as

Enables collection and export of the AS information for NetFlow.

EXAMPLE:

```
BigIron(config)# ip flow-export peer-as
```

Syntax: [no] ip flow-export origin-as | peer-as

The **origin-as | peer-as** parameter specifies the type of AS information you want to enable. You can enable one or the other but not both. By default, neither type of AS information is enabled.

Possible values: N/A

Default value: No AS information is collected

ip flow-export protocol-disable

Reduces flow exports to NetFlow collectors.

By default, NetFlow exports flows for all IP protocols (TCP, UDP, IGRP, OSPF, and so on).

EXAMPLE:

```
BigIron(config)# ip flow-export protocol-disable udp
```

Syntax: [no] ip flow-export protocol-disable tcp | udp | other

The **tcp | udp | other** parameter specifies the protocol for which you are disabling export. You can specify one of the following:

- **tcp** – TCP
- **udp** – udp
- **other** – All IP protocols ***except*** TCP and UDP

Possible values: See above

Default value: All protocols are exported

ip flow-export source

Specifies a source interface for NetFlow.

By default, the Foundry device uses the port that is connected to a collector as the source interface for flows exported to that collector. You can specify an Ethernet port or POS port, a loopback interface, or the null interface as the source for NetFlow export packets.

- Ethernet or POS port – NetFlow sends the export packets out the specified interface.
- Loopback interface – NetFlow sends the export packets from the specified loopback address, using a physical port connected to the collector to transmit the packet.
- Null interface – NetFlow continues to collect flows but does not export them to the collector. Use this type of interface when you want to administratively stop flow export without stopping flow collection and without removing configuration information.

EXAMPLE:

```
BigIron(config)# ip flow-export source ethernet 1/1
```

This command configures port 1/1 to be the source interface for NetFlow packets. Since the command does not specify the collector ID, NetFlow exports the flows to collector 1.

To specify the collector ID, enter a command such as the following:

```
BigIron(config)# ip flow-export source ethernet 1/1 2
```

This command uses port 1/1 as the source for flows exported to collector 2.

Syntax: [no] ip flow-export source ethernet | pos | loopback <portnum> [<collector-id>]

Syntax: [no] ip flow-export source null [<collector-id>]

The **ethernet | pos | loopback** <portnum> specifies a physical port or loopback interface.

The **null** parameter discards the export packets instead of sending them to a collector. However, NetFlow continues to collect flows.

The <collector-id> specifies the collector. If you do not specify the collector ID, the device assumes you mean collector 1.

Possible values: See above

Default value: The interface connected to the collector.

ip flow-export version

Changes the format version for data export packets sent by NetFlow.

By default, NetFlow uses format version 5 for exporting flows from the main cache to the external collector.

```
BigIron(config)# ip flow-export version 1
```

Syntax: [no] ip flow-export version 1 | 5

NOTE: The format for the main cache is not related to the format for aggregate caches. The software automatically uses format 8 for export packets from the aggregate caches.

Possible values: See above

Default value: 5

ip forward-protocol

This command is used in conjunction with the UDP helper feature to define the type of application traffic (port number socket) that is being forwarded to the server.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron(config)# ip forward-protocol udp snmp-trap
```

Syntax: ip forward-protocol udp <udp-port-name> | <udp-port-num>

Possible values:

number	echo	snmp-trap
bootpc	mobile-ip	tacacs
bootps	netbios-dgm	talk
discard	netbios-ns	
dnsix	ntp	
tftp	snmp	

In addition, you can specify any UDP application by using the application's UDP port number.

Default value: By default, when an IP helper address is configured on an interface, UDP broadcast forwarding is enabled for the following UDP packet types: bootps, domain, tftp, time, netbios-dgm, netbios-ns and tacacs.

ip high-perf

Disables the unicast high-performance mode.

NOTE: To place a change to the high-performance mode into effect, you must reload the software after saving the change to the startup-config file.

EXAMPLE:

To disable the high-performance mode, enter the following commands:

```
BigIron(config)# no ip high-perf
BigIron(config)# write memory
BigIron(config)# end
BigIron# reload
```

To enable the high-performance mode, enter the following commands:

```
BigIron(config)# ip high-perf
BigIron(config)# write memory
BigIron(config)# end
BigIron# reload
```

Syntax: [no] ip high-perf

Possible values: N/A

Default value: Enabled

NOTE: The feature is disabled by default in software releases earlier than 07.5.01.

ip icmp

Causes the Foundry device to drop ICMP packets when excessive numbers are encountered, as is the case when the device is the victim of a Smurf attack. This command allows you to set threshold values for ICMP packets targeted at the router and drop them when the thresholds are exceeded.

EXAMPLE:

In the following example, if the number of ICMP packets received per second exceeds 5,000, the excess packets are dropped. If the number of ICMP packets received per second exceeds 10,000, the device drops all ICMP packets for the next 300 seconds (five minutes).

```
BigIron(config)# ip icmp burst-normal 5000 burst-max 10000 lockup 300
```


You can set threshold values for ICMP packets received on an interface and drop them when the thresholds are exceeded. For example:

```
BigIron(config)# int e 3/11
BigIron(config-if-e100-3/11)# ip icmp burst-normal 5000 burst-max 10000 lockup 300
```

Syntax: ip icmp burst-normal <value> burst-max <value> lockup <seconds>

The burst-normal value can be from 1 – 100000.

The burst-max value can be from 1 – 100000.

The lockup value can be from 1 – 10000.

The number of incoming ICMP packets per second are measured and compared to the threshold values as follows:

- If the number of ICMP packets exceeds the burst-normal value, the excess ICMP packets are dropped.
- If the number of ICMP packets exceeds the burst-max value, all ICMP packets are 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.

Possible values: The burst-normal and burst-max values can be between 1 – 100000 packets. The burst-normal value must be smaller than the burst-max value. The lockup value can be between 1 – 10000 seconds.

Default value: N/A

ip icmp echo broadcast-request

Disables ICMP echo (ping) replies. By default, Foundry devices are enabled to respond to broadcast ICMP echo packets, which are ping requests. You can disable response to ping requests on a global basis.

EXAMPLE:

To disable response to broadcast ICMP echo packets (ping requests), enter the following command:

```
BigIron(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:

```
BigIron(config)# ip icmp echo broadcast-request
```

Possible values: enabled or disabled

Default value: enabled

ip icmp redirects

Disables ICMP redirect messages on a global basis.

NOTE: The device forwards misdirected traffic to the appropriate router, even if you disable the redirect messages.

EXAMPLE:

To disable ICMP redirect messages globally, enter the following command at the global CONFIG level of the CLI:

```
BigIron(config)# no ip icmp redirects
```

Syntax: [no] ip icmp redirects

Possible values: N/A

Default value: Redirect messages are enabled

ip icmp unreachable

Disables 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.
- Network – The Foundry device cannot reach the network specified in the destination IP address of the packet.
- 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.

EXAMPLE:

To disable all ICMP Unreachable messages, enter the following command:

```
BigIron(config)# no ip icmp unreachable
```

Syntax: [no] ip icmp unreachable [network | 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 **network** parameter disables ICMP Network Unreachable messages.

The **host** parameter disables ICMP Host Unreachable messages.

The **protocol** parameter disables ICMP Protocol Unreachable messages.

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 **port** parameter disables ICMP Port Unreachable messages.

The **source-route-fail** parameter disables ICMP Unreachable (caused by Source-Route-Failure) messages.

To disable ICMP Host Unreachable messages and ICMP Network Unreachable messages but leave the other types of ICMP Unreachable messages enabled, enter the following commands instead of the command shown above:

```
BigIron(config)# no ip icmp unreachable host
BigIron(config)# no ip icmp unreachable network
```

If you have disabled all ICMP Unreachable message types but you want to re-enable certain types, you can do so entering commands such as the following:

```
BigIron(config)# ip icmp unreachable host
BigIron(config)# ip icmp unreachable network
```


The commands shown above re-enable ICMP Unreachable Host messages and ICMP Network Unreachable messages.

Possible values: see above

Default value: all types of ICMP Destination Unreachable messages are enabled

ip igmp group-membership-time

Defines how long a group will remain on an interface in the absence of a group report, if DVMRP is enabled on the router.

NOTE: You must enter the **ip multicast-routing** command before entering this command. Otherwise, the command does not take effect and the software uses the default value.

EXAMPLE:

```
BigIron(config)# ip igmp group-membership-time 240
```

Syntax: ip igmp group-membership-time <value>

Possible values: 1 – 7200 seconds

Default value: 140 seconds

ip igmp max-response-time

Defines 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.

NOTE: You must enter the **ip multicast-routing** command before entering this command. Otherwise, the command does not take effect and the software uses the default value.

EXAMPLE:

```
BigIron(config)# ip igmp max-response-time 8
```

Syntax: ip igmp max-response-time <value>

Possible values: 1 – 10 seconds

Default value: 5 seconds

ip igmp query-interval

Defines how often the router will query an interface for group membership.

NOTE: You must enter the **ip multicast-routing** command before entering this command. Otherwise, the command does not take effect and the software uses the default value.

EXAMPLE:

```
BigIron(config)# ip igmp query 120
```

Syntax: ip igmp query-interval <value>

Possible values: 1 – 3600 seconds

Default value: 60 seconds

ip irdp

Enables a router to advertise its network IP addresses to the network. The router will also answer queries. IRDP stands for ICMP Router Discovery Protocol (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.

When IRDP is enabled, the Layer 3 Switch periodically sends Router Advertisement messages out all its IP interfaces. 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 Router Solicitation messages to discover their default gateway. When IRDP is enabled on the Foundry Layer 3 Switch, it 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.

When IRDP is enabled, the Layer 3 Switch sends the Router Advertisement messages every 450 – 600 seconds. The interval the device selects is random for each message and is not affected by traffic loads or other network factors. The interval is not configurable.

EXAMPLE:

```
BigIron(config)# ip irdp
```

Syntax: [no] ip irdp

Possible values: n/a

Default value: disabled

ip load-sharing

Allows traffic being sent from one router to another to be sent across multiple paths of equal cost for faster transmission when using OSPF or BGP4 routing. OSPF or BGP4 routing must be enabled on the router for this command to operate. IP load sharing is enabled by default.

See the "Configuring IP" chapter of the *Foundry Enterprise Configuration and Management Guide* for more information about this feature.

EXAMPLE:

```
BigIron(config)# ip load-sharing 6
```

Syntax: ip load-sharing [<num>]

The <num> parameter specifies the number of equal paths across which the Layer 3 Switch will load share traffic to a given destination. You can specify from 2 – 8. The destinations among which the device load shares can be network addresses or individual host addresses, depending on the load sharing method that is enabled. See “ip load-sharing by-host”.

Possible values: 2 – 8

Default value: 4

ip load-sharing by-host

Disables network-based load sharing (load sharing using destination address aggregation) and configures the Layer 3 Switch to instead perform load sharing based on individual host destination addresses.

See the "Configuring IP" chapter of the *Foundry Enterprise Configuration and Management Guide* for more information about this feature.

EXAMPLE:

To enable host-based IP load sharing, enter the following command:

```
BigIron(config)# ip load-sharing by-host
```

This command enables host-based IP load sharing on the device. The command also disables network-based IP load-sharing (the default) at the same time.

Syntax: [no] ip load-sharing by-host

NOTE: The VM1 uses hash-based load balancing regardless of the type of IP load sharing enabled (**by-host** or **by-network**). On the VM1, a hash value is calculated based on the source and destination IP addresses. Each of the paths to a given destination is associated with one of the possible hash values, and the traffic flow is assigned to a path based on its calculated hash value.

Possible values: see above

Default value: disabled

ip load-sharing route-by-host

Overrides network-based IP load sharing for a specific destination network. Use this feature when you want to use network-based load sharing by default but also want to use host-based load sharing for specific destinations (hosts or sub-nets).

NOTE: This feature applies only to Chassis Layer 3 Switches. Stackable Layer 3 Switches perform host-based load sharing for all destinations and cannot be configured for network-based load sharing. Use this feature only when network-based load sharing is enabled.

When you configure host-based load sharing for a specific destination network, the Layer 3 Switch distributes traffic to hosts on the network evenly across the available paths. For other networks, the Layer 3 Switch uses a single path for all traffic to hosts on a given network.

NOTE: The host-based load sharing for the destination takes effect only if the IP route table contain an entry that exactly matches the destination network you specify. For example, if you configure host-based load sharing for destination network 207.95.7.0/24, the IP route table must contain a route entry for that network. In fact, for load sharing to occur, the IP route table needs to contain multiple equal-cost paths to the network.

EXAMPLE:

To enable host-based load sharing for a specific destination network, enter a command such as the following at the global CONFIG level of the CLI:

```
BigIron(config)# ip load-sharing route-by-host 207.95.7.0/24
```

This command configures the Layer 3 Switch to use host-based load sharing for traffic to destinations on the 207.95.7.0/24 network. The Layer 3 Switch uses network-based load sharing for traffic to other destination networks.

Syntax: [no] ip load-sharing route-by-host <ip-addr> <ip-mask>

or

Syntax: [no] ip load-sharing route-by-host <ip-addr>/<mask-bits>

You can disable host-based load sharing for specific destination networks or for all networks. When you disable host-based load sharing for a destination network (or for all destination networks), the software removes the host-based forwarding cache entries for the destination network(s) and uses network-based forwarding entries instead.

NOTE: This method applies only to networks for which you have explicitly enabled host-based load sharing. If you have enabled host-based load sharing globally but want to change to network-based load sharing, enter the **no ip load-sharing by-host** command at the global CONFIG level of the CLI.

To disable host-based load sharing for all the destination networks for which you have explicitly enabled the host-based load sharing, enter the following command at the global CONFIG level of the CLI:

```
BigIron(config)# no ip load-sharing route-by-host
```

To disable host-based load sharing for a specific destination network, enter a command such as the following:

```
BigIron(config)# no ip load-sharing route-by-host 207.95.7.0/24
```

This command removes the host-based load sharing for the 208.95.7.0/24 network, but leaves the other host-based load sharing configurations intact.

Possible values: a network address

Default value: disabled

ip mroute

Configures a static multicast route. If you 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.

NOTE: Static multicast routes are not supported for DVMRP.

EXAMPLE:

```
BigIron(config)# ip mroute 1 207.95.10.0/24 interface ethernet 1/2 distance 1
```

Syntax: mroute <routenum> <ip-addr> interface ethernet <portnum> | ve <num> [distance <num>]

Or

Syntax: mroute <routenum> <ip-addr> rpf_address <rpf-num>

Possible values: The <ip-addr> parameter 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** <portnum> parameter to specify a physical port or the ve <num> parameter to specify a virtual interface.

NOTE: The **ethernet** <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.

Default value: N/A

ip multicast

By default, Foundry Layer 2 Switches forward all IP multicast traffic out all ports except the port on which the traffic was received. To reduce multicast traffic through the Layer 2 Switch, you can enable IP Multicast Traffic Reduction. This feature configures the Layer 2 Switch to forward multicast traffic only on the ports attached to multicast group members. The Layer 2 Switch determines the ports that are attached to multicast group members based on entries in the IGMP table. Each entry in the table consists of an IP multicast group address and the Foundry Layer 2 Switch ports from which the Layer 2 Switch has received Group Membership reports for that group.

NOTE: You must reload the software after making this configuration change and saving it to the startup-config file.

After you enable IP Multicast Traffic Reduction, when the Layer 2 Switch receives traffic for an IP multicast group, the Layer 2 Switch looks in the IGMP table for an entry for that group. If the Layer 2 Switch finds an entry, the Layer 2 Switch forwards the group traffic out the ports listed in the group entry. If the table does not contain an entry for the group, the Layer 2 Switch broadcasts the traffic.

When you enable IP Multicast Traffic Reduction on the Layer 2 Switch, IGMP also is enabled. The Layer 2 Switch uses IGMP to maintain a table of the Group Membership reports received by the Layer 2 Switch. You can use active or passive IGMP mode. The default mode is passive.

- Active – When active IGMP mode is enabled, a Foundry Layer 2 Switch 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 Layer 2 Switch is in a stand-alone switched network with no external IP multicast router attachments. In this case, enable the active IGMP mode on only one of the Layer 2 Switches and leave the other Layer 2 Switches configured for passive IGMP mode.

- Passive – When passive IGMP mode is enabled, the Layer 2 Switch 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.

EXAMPLE:

```
FastIron(config)# ip multicast passive
FastIron(config)# write memory
FastIron(config)# end
FastIron# reload
```

Syntax: ip multicast [active | passive]

Possible values: Active or passive

Default value: Disabled

ip multicast age-interval

Changes the IGMP age interval on Layer 2 Switches configured for IP Multicast Traffic Reduction.

When the Layer 2 Switch receives a Group Membership report, it 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 Layer 2 Switch receiving another Group Membership report.

NOTE: You must reload the software after making this configuration change and saving it to the startup-config file.

EXAMPLE:

To modify the age interval, enter a command such as the following:

```
FastIron(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.

Possible values: 10 – 1220 seconds

Default value: 140 seconds

ip multicast filter

Enables IP multicast filtering on Layer 2 Switches that are enabled for IP Multicast Traffic Reduction.

By default, Foundry devices forward multicast traffic for all valid multicast groups. You can configure a Foundry Layer 2 Switch to filter out all multicast traffic for groups other than the ones for which the Layer 2 Switch has received Group Membership reports.

Thus configured, the Layer 2 Switch forwards all multicast groups once the Layer 2 Switch is started, until it receives a Group Membership report. Once the Layer 2 Switch receives a Group Membership report, it drops all multicast packets for groups other than the ones for which the Layer 2 Switch has received the Group Membership report. Once the Layer 2 Switch receives a Group Membership report for a given group, it forwards traffic for that group instead of dropping the traffic.

NOTE: You must reload the software after making this configuration change and saving it to the startup-config file.

EXAMPLE:

To enable IP multicast filtering, enter the following command:

```
FastIron(config)# ip multicast filter
```

Syntax: [no] ip multicast filter

Possible values: N/A

Default value: Disabled

ip multicast-perf

Enables the device to forward all the fragments of fragmented IP multicast packet through hardware. By default, a Foundry Layer 3 Switch forwards the first fragment of a fragmented IP multicast packet through hardware, but forwards the remaining fragments through the software.

NOTE: This feature applies only to Layer 3 Switches, not to Layer 2 Switches.

EXAMPLE:

```
BigIron(config)# ip multicast-perf
BigIron(config)# write memory
BigIron(config)# end
BigIron# reload
```

Syntax: [no] ip multicast-perf

NOTE: You must save the configuration and reload the software to place the change into effect.

Possible values: N/A

Default value: Disabled

ip multicast query-interval

Changes the IGMP query interval on Layer 2 Switches configured for IP Multicast Traffic Reduction.

The query interval specifies how often a Foundry Layer 2 Switch 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.

NOTE: You must reload the software after making this configuration change and saving it to the startup-config file.

EXAMPLE:

To modify the query interval, enter a command such as the following:

```
FastIron(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.

Possible values: 10 – 600 seconds

Default value: 60 seconds

ip multicast-routing

Allows you to change the following global IP Multicast parameters:

- IGMP query interval

- IGMP group membership time
- IGMP maximum response time

NOTE: You must enter the **ip multicast-routing** command before changing these parameters. Otherwise, the changes do not take effect and the software uses the default values.

EXAMPLE:

```
BigIron(config)# ip multicast-routing
```

Syntax: [no] ip multicast-routing

Possible values: N/A

Default value: Disabled

ip nat inside destination list

Configures a source IP address list for dynamic inside destination NAT. You also need to configure an IP ACL and an address pool. See “ip nat pool” on page 6-62.

EXAMPLE:

To configure dynamic inside-destination NAT, enter commands such as the following at the global CONFIG level of the CLI:

```
BigIron(config)# access-list 1 permit 209.157.1.2/24
BigIron(config)# ip nat pool InAdds 10.10.10.0 10.10.10.254 prefix-length 24
BigIron(config)# ip nat inside destination list 1 pool InAdds
```

These commands configure a standard ACL for the public network 10.10.10.x/24, then enable inside-destination NAT for the network. Make sure you specify **permit** in the ACL, rather than **deny**. If you specify **deny**, the Foundry device will not provide NAT for the addresses.

Syntax: [no] ip nat inside destination list <acl-name-or-num> pool <pool-name>

The **inside destination** parameter specifies that the translation applies to public addresses sending traffic to private addresses.

The **list** <acl-id> parameter specifies an IP ACL (standard or extended). You can specify a numbered or named ACL.

NOTE: Named ACLs are not supported with NAT. You must use a numbered ACL.

The **pool** <pool-name> parameter specifies the pool. You must create the pool before you can use it with this command.

Possible values: See above

Default value: Not configured

ip nat inside destination static

Configures static inside destination NAT for an IP address.

EXAMPLE:

To configure static inside-destination NAT for an IP address, enter a command such as the following:

```
BigIron(config)# ip nat inside destination static 209.157.1.69 10.10.10.69
```

The command in this example statically maps the Internet address 209.157.1.69 to the private address 10.10.10.69.

To include TCP or UDP application port numbers in the translation, enter a command such as the following:

```
BigIron(config)# ip nat inside destination static tcp 209.157.1.69 80 10.10.10.69
8080
```


This command provides the same IP address translation as the previous command example. However, this command also translates TCP port 80 to TCP port 8080. The translation applies to the destination port, for inbound traffic.

Syntax: [no] ip nat inside destination static <private-ip> <global-ip>

Syntax: [no] ip nat inside destination static tcp | udp <private-ip> <private-tcp/udp> <global-ip> <global-tcp/udp>

The **inside destination** parameter specifies that the mapping applies to the Internet address sending traffic to the private network.

The <private-ip> parameter specifies the private IP address.

The <global-ip> parameter specifies the Internet address.

NOTE: Neither of the IP address parameters needs a network mask.

The **tcp | udp** parameter indicates that you are creating a static mapping for a specify application (TCP or UDP port).

The <global-tcp/udp> parameter specifies the application port on the public host.

The <private-tcp/udp> parameter specifies the application port on the private host.

Possible values: See above

Default value: Not configured

ip nat inside source list

Configures a source IP address list for dynamic inside source NAT. You also need to configure an IP ACL and an address pool. See “ip nat pool” on page 6-62.

EXAMPLE:

```
BigIron(config)# access-list 1 permit 10.10.10.0/24
BigIron(config)# ip nat pool OutAdds 209.157.1.2 209.157.2.254 prefix-length 24
BigIron(config)# ip nat inside source list 1 pool OutAdds
```

These commands configure a standard ACL for the private sub-net 10.10.10.x/24, then enable inside NAT for the sub-net. Make sure you specify permit in the ACL, rather than deny. If you specify deny, the Foundry device will not provide NAT for the addresses.

Syntax: [no] ip nat inside source list <acl-name-or-num> pool <pool-name> [overload]

This command associates a private address range with a pool of Internet addresses and optionally enables the Port Address Translation feature.

The **inside source** parameter specifies that the translation applies to private addresses sending traffic to global addresses (Internet addresses).

The **list** <acl-id> parameter specifies a standard or extended ACL. You can specify a numbered or named ACL.

NOTE: Named ACLs are not supported with NAT. You must use a numbered ACL.

The **pool** <pool-name> parameter specifies the pool. You must create the pool before you can use it with this command.

The overload parameter enables the Port Address Translation feature. Use this parameter if the IP address pool does not contain enough addresses to ensure NAT for each private address. The Port Address Translation feature conserves Internet addresses by mapping the same Internet address to more than one private address and using a TCP or UDP port number to distinguish among the private hosts. The device supports up to 50 global IP addresses with this feature enabled.

Possible values: See above

Default value: Not configured

ip nat inside source static

Configures static inside source NAT for an IP address.

EXAMPLE:

```
BigIron(config)# ip nat inside source static 10.10.10.69 209.157.1.69
```

The commands in this example statically map the private address 10.10.10.69 to the Internet address 209.157.1.69.

Syntax: [no] ip nat inside source static <private-ip> <global-ip>

This command associates a specific private address with a specific Internet address. Use this command when you want to ensure that the specified addresses are always mapped together.

The **inside source** parameter specifies that the mapping applies to the private address sending traffic to the Internet.

The <private-ip> parameter specifies the private IP address.

The <global-ip> parameter specifies the Internet address. The device supports up to 256 global IP addresses.

Neither of the IP address parameters needs a network mask.

Possible values: See above

Default value: Not configured

ip nat pool

Configures a pool for use in a source IP address list for dynamic NAT.

EXAMPLE:

```
BigIron(config)# access-list 1 permit 10.10.10.0/24
BigIron(config)# ip nat pool OutAdds 209.157.1.2 209.157.2.254 prefix-length 24
BigIron(config)# ip nat inside source list 1 pool OutAdds
```

These commands configure a standard ACL for the private sub-net 10.10.10.x/24, then enable inside NAT for the sub-net. Make sure you specify permit in the ACL, rather than deny. If you specify deny, the Foundry device will not provide NAT for the addresses.

Syntax: [no] ip nat pool <pool-name> <start-ip> <end-ip> netmask <ip-mask> | prefix-length <length> [type match-host | rotary]

This command configures the address pool.

The <pool-name> parameter specifies the pool name. The name can be up to 255 characters long and can contain special characters and internal blanks. If you use internal blanks, you must use quotation marks around the entire name.

The <start-ip> parameter specifies the IP address at the beginning of the pool range. Specify the lowest-numbered IP address in the range.

The <end-ip> parameter specifies the IP address at the end of the pool range. Specify the highest-numbered IP address in the range.

NOTE: The address range cannot contain any gaps. Make sure you own all the IP addresses in the range. If the range contains gaps, you must create separate pools containing only the addresses you own.

The **netmask** <ip-mask> | **prefix-length** <length> parameter specifies a classical sub-net mask (example: **netmask** 255.255.255.0) or the length of a Classless Interdomain Routing prefix (example: **prefix-length** 24).

NOTE: The maximum number of global IP addresses you can configure depends on how much memory the Layer 3 Switch has and whether you enable the Port Address Translation feature. Regardless of the amount of memory, you cannot configure more than 256 global IP addresses.

The **type match-host | rotary** parameter specifies the method the software uses to assign the host portion of the translated address.

- **match-host** – The software uses the same host address as the untranslated address. For example, if the untranslated address is 192.2.4.69 and the host portion of the address is 69, the translated address also uses the host address 69. This method results in the translated addresses always having the same host addresses as their untranslated counterparts.
- **rotary** – The software assigns a host address from 1 – 254, beginning with 1 for the first translated address. This is the default.

Possible values: N/A

Default value: Disabled

ip nat translation

Changes the age timeout for NAT translations.

EXAMPLE:

To change the age timeout for all entries that do not use Port Address Translation to 1800 seconds (one half hour), enter a command such as the following at the global CONFIG level of the CLI:

```
BigIron(config)# ip nat timeout 1800
```

Syntax: [no] ip nat translation timeout | udp-timeout | tcp-timeout | finrst-timeout | dns-timeout <secs>

Use one of the following parameters to specify the dynamic entry type:

- **timeout** – All entries that do not use Port Address Translation. The default is 120 seconds.
- **udp-timeout** – Dynamic entries that use Port Address Translation based on UDP port numbers. The default is 120 seconds.
- **tcp-timeout** – Dynamic entries that use Port Address Translation based on TCP port numbers. The default is 120 seconds.
- **finrst-timeout** – TCP FIN (finish) and RST (reset) packets, which normally terminate TCP connections. The default is 120 seconds.
- **dns-timeout** – Connections to a Domain Name Server (DNS). The default is 120 seconds.

The <secs> parameter specifies the number of seconds. For each entry type, you can enter a value from 1 – 3600.

Possible values: See above

Default value: See above

ip net-aggregate

Optimizes the CAM for devices that have very large IP route tables (100,000 or more), where most of those routes use the same next hops as the default route.

When you enable CAM network aggregation, the feature divides the IP address space into 4096 aggregates. Each aggregate has a 12-bit prefix (/12).

With CAM network aggregation enabled, the device forward IP traffic as follows:

The device checks the CAM for an entry with the traffic's destination.

- If the CAM contains an entry, the device uses the entry.
- If the CAM does not contain an entry, the device checks to see whether all explicit routes in the IP route table that are in the same /12 aggregate as the needed route (all routes that overlap with the /12 aggregate), have the same set of next hops as the default route.
    - If all explicit routes in the IP route table that are within the same /12 aggregate use the same next hops as the default route, the device programs a single CAM entry that aggregates the route information for all routes within the aggregate. The device uses this single CAM entry to forward traffic to any destination

within the aggregate.

- If one or more explicit routes within the same /12 aggregate uses a next hop that is not also used by the default route, the device does not program an aggregate entry into the CAM but instead programs a separate route entry for the individual destination network.

After programming a CAM entry for the traffic's destination, the device uses the entry to forward further traffic to the same destination. If the device was able to program an aggregate entry, the device uses the entry for traffic to any destination within the aggregate.

NOTE: CAM network aggregation requires a default route in the IP route table.

Compare with "ip dr-aggregate" on page 6-46.

EXAMPLE:

```
BigIron(config)# ip net-aggregate
```

Syntax: [no] ip net-aggregate [<secs>]

The <secs> parameter specifies the update interval and can be from 1 – 60 seconds. The default is 1 second. Specifying a longer interval can help conserve CPU resources.

Possible values: See above

Default value: Disabled

ip pimsm-snooping

Enables PIM SM snooping, which provides multicast traffic reduction on Layer 2 Switches.

NOTE: This command applies only to Layer 2 Switches.

PIM SM traffic snooping eliminates superfluous PIM SM traffic by configuring the Layer 2 Switch 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 Layer 2 Switch.

NOTE: This feature applies only to PIM SM version 2 (PIM V2).

EXAMPLE:

To enable PIM SM traffic snooping, enter the following commands at the global CONFIG level of the CLI:

```
BigIron-SW(config)# ip multicast
BigIron-SW(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 Layer 2 Switch 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. Therefore, you must use the passive mode.

Syntax: [no] ip pimsm-snooping

This command enables PIM SM traffic snooping.

To disable the feature, enter the following command:

```
BigIron-SW(config)# no ip pimsm-snooping
```


If you also want to disable IP multicast traffic reduction, enter the following command:

```
BigIron-SW(config)# no ip multicast
```

Possible values: N/A

Default value: Disabled

ip policy prefer-direct-route

Configures a Policy-Based Routing (PBR) policies to always use the most direct route available.

NOTE: This command applies only to JetCore devices running software release 07.6.01 or later.

EXAMPLE:

```
BigIron(config)# ip policy prefer-direct-route
```

Syntax: [no] ip policy prefer-direct-route

Possible values: N/A

Default value: Disabled

ip policy route-map

Enables Policy-Based Routing (PBR) on the Layer 3 Switch.

EXAMPLE:

To enable PBR globally, enter a command such as the following:

```
BigIron(config)# ip policy route-map test-route
```

This command applies a route map named “test-route” to all interfaces on the device for PBR.

Syntax: [no] ip policy route-map <map-name>

Possible values: the name of a configured route map

Default value: N/A

ip prefix-list

Configures an IP prefix list. You can configure a range of IP prefixes for routes you want to send to or receive from individual neighbors.

EXAMPLE:

To configure an IP prefix list and apply it to a neighbor, enter commands such as the following:

```
BigIron(config)# ip prefix-list Routesfor20 permit 20.20.0.0/24
BigIron(config)# router bgp
BigIron(config-bgp-router)# neighbor 10.10.10.1 prefix-list Routesfrom20 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 "neighbor" on page 13-12.

Possible values: see above

Default value: N/A

ip proxy-arp

Allows a router to act as a proxy for devices on its interfaces when responding to ARP requests.

EXAMPLE:

```
BigIron(config)# ip proxy
```

Syntax: [no] ip proxy-arp

Possible values: On or off

Default value: Off

ip radius source-interface

Configures the device to use the lowest-numbered IP address configured on an interface as the source for all RADIUS packets from the device. The software uses the lowest-numbered IP address configured on the interface as the source IP address for the packets.

EXAMPLE:

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:

```
BigIron(config)# int ve 1
BigIron(config-vif-1)# ip address 10.0.0.3/24
BigIron(config-vif-1)# exit
BigIron(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 atm <portnum>.<subif> | ethernet <portnum> | loopback <num> | ve <num>

Possible values: see above

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

ip rarp

Enables Reverse Addressing Resolution Protocol (RARP) and allows the router to assign IP addresses for hosts based on their MAC addresses. A router will check the RARP table for an IP match to a MAC address sent from a host. If the table contains an entry for the MAC address, the router will answer back with the IP address.

EXAMPLE:

```
BigIron(config)# ip rarp
```

Syntax: ip rarp

Possible values: N/A

Default value: N/A

ip rebind-acl

Reapplies ACLs to their interfaces.

NOTE: This command applies on JetCore devices running software release 07.6.01 or later.

For flow-based and hardware-based ACLs, if you make an ACL configuration change, you must reapply the ACLs to their interfaces to place the change into effect. An ACL configuration change includes any of the following:

- Adding, changing, or removing an ACL or an entry in an ACL
- Changing a PBR policy
- Changing the port membership of a VLAN that has an ACL on its virtual routing interface
- Enabling or disabling the TCP strict mode or UDP strict mode (flow-based ACLs only)
- Changing JetCore ToS-based QoS mappings (since JetCore QoS uses the Layer 4 CAM)

EXAMPLE:

```
BigIron(config)# ip rebind-acl all
```

This command reapplies all ACLs to their interfaces.

To reapply a specific ACL, enter a command such as the following:

```
BigIron(config)# ip rebind-acl 101
```

This command reapplies ACL 101 only.

Syntax: [no] ip rebind-acl <num> | <name> | all

Possible values: See above.

Default value: N/A

ip route

Allows you to configure static IP routes on a Layer 3 Switch.

EXAMPLE:

```
BigIron(config)# ip route 192.128.2.0 255.255.255.0 209.157.22.1
```

Syntax: ip route <dest-ip-addr> <dest-mask>
<next-hop-ip-addr> |
atm <slotnum>/<portnum>.<subif> | ethernet <portnum> | pos <portnum> | ve <num>
[<metric>] [distance <num>] [lsp <name> | static-lsp <name>]

or

Syntax: ip route <dest-ip-addr>/<mask-bits>
<next-hop-ip-addr> |
atm <slotnum>/<portnum>.<subif> | ethernet <portnum> | pos <portnum> | ve <num>
[<metric>] [distance <num>] [lsp <name> | static-lsp <name>]

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. You can enter multiple static routes for the same destination for load balancing or redundancy. See the "Defining Static

IP Routes" section in the "Configuring IP" chapter in the *Foundry Enterprise Configuration and Management Guide*.

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 or POS port, the <portnum> is the port's number (including the slot number, if you are configuring a BigIron or FastIron II). 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 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.

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

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: You can also assign the default router as the destination by entering 0.0.0.0 0.0.0.0.

The **lsp** and **static-lsp** parameters specify the name of an MPLS LSP; packets matching the specified prefix are assigned to the LSP.

Default value: metric 1, distance 1

NOTE: The Layer 3 Switch will replace the static route if the router receives a route with a lower administrative distance. See the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide* for a list of the default administrative distances for all types of routes.

The syntax above is for all types of static routes except "null" routes. To configure a null static route, use the following syntax.

Syntax: ip route <ip-addr> <ip-mask> null0 [<metric>] [distance <num>]

or

Syntax: ip route <ip-addr>/<mask-bits> null0 [<metric>] [distance <num>]

The **null0** parameter indicates that this is a null route. You must specify this parameter to make this a null route. For more information, see the "Configuring IP" chapter of the *Foundry Enterprise Configuration and Management Guide*.

ip router-id

Assigns a router ID to a Foundry Layer 3 Switch. OSPF and BGP4 use router IDs to identify routers. A Layer 3 Switch can have one router ID, which is used by both OSPF and BGP4 if both are enabled.

Router IDs are in IP address format (for example, 1.1.1.1). The default router ID is the IP address configured on the lowest numbered loopback interface configured on the Layer 3 Switch. If the device does not have any loopback interfaces, the default router ID is the lowest numbered IP interface configured on the device. This ensures that the router ID on each router is unique even if you use the default value.

EXAMPLE:

```
BigIron(config)# ip router-id 1.1.1.1
```

Syntax: ip router-id <ip-addr>

Possible values: N/A

Default value: the numerically lowest IP address configured on the Layer 3 Switch

ip show-portname

Displays the name of the interface instead of its number. By default an interface's slot number (if applicable) and port number are displayed when you display Syslog messages. In software release 07.6.02 and later, if you want to display the name of the interface instead of its number, enter a command such as the following:

EXAMPLE:

```
BigIron(config)# ip show-portname
```

This command is applied globally to all interfaces on Layer 2 Switches and Layer 3 Switches.

Syntax: [no] Ip show-portname

When you display the messages in the Syslog, you see the interface name under the Dynamic Log Buffer section. The actual interface number is appended to the interface name. For example, if the interface name is "lab" and its port number is "2", you see "lab2". For an example screen display, see the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Possible values: N/A

Default value: N/A

ip show-subnet-length

Changes display of network mask information from class-based notation (xxx.xxx.xxx.xxx) to Classless Interdomain Routing (CIDR) notation. By default, Foundry devices display network mask information in class-based notation.

EXAMPLE:

```
BigIron(config)# ip show-subnet-length
```

Syntax: [no] ip show-subnet-length

Possible values: N/A

Default value: Disabled

ip source-route

Disables or re-enables forwarding of IP source-routed packets.

EXAMPLE:

To disable forwarding of IP source-routed packets, enter the following command:

```
BigIron(config)# no ip source-route
```

Syntax: [no] ip source-route

To re-enable forwarding of source-routed packets, enter the following command:

```
BigIron(config)# ip source-route
```

Possible values: N/A

Default value: Disabled

ip ssh authentication-retries

Sets the number of SSH authentication retries.

EXAMPLE:

The following command changes the number of authentication retries to 5:

```
BigIron(config)# ip ssh authentication-retries 5
```

Syntax: ip ssh authentication-retries <number>

Possible values: 1 – 5

Default value: 3

ip ssh idle-time

Sets the amount of time an SSH session can be inactive before the Foundry device closes it.

EXAMPLE:

```
BigIron(config)# ip ssh idle-time 30
```

Syntax: ip ssh idle-time <minutes>

Possible values: 0 – 240 minutes

Default value: 0 minutes

ip ssh key-size

Sets the SSH key size.

EXAMPLE:

The following command changes the server RSA key size to 896 bits:

```
BigIron(config)# ip ssh key-size 896
```

Syntax: ip ssh key-size <number>

NOTE: The size of the host RSA key that resides in the system-config file is always 1024 bits and cannot be changed.

Possible values: 512 – 896 bits

Default value: 768 bits

ip ssh password-authentication

Disables SSH password authentication.

After the SSH server on the Foundry device negotiates a session key and encryption method with the connecting client, user authentication takes place. Of the methods of user authentication available in SSH, Foundry's implementation of SSH supports password authentication only.

With password authentication, users are prompted for a password when they attempt to log into the device (unless empty password logins are not allowed; see "ip ssh permit-empty-passwd"). If there is no user account that matches the user name and password supplied by the user, the user is not granted access.

You can deactivate password authentication for SSH. However, since password authentication is the only user authentication method supported for SSH, this means that no user authentication is performed at all. Deactivating password authentication essentially disables the SSH server entirely.

EXAMPLE:

To deactivate password authentication:

```
BigIron(config)# ip ssh password-authentication no
```

Syntax: ip ssh password-authentication no | yes

Possible values: N/A

Default value: Enabled

ip ssh permit-empty-passwd

Enables empty password SSH logins. By default, empty password logins are not allowed. This means that users with an SSH client are always prompted for a password when they log into the device. To gain access to the device, each user must have a user name and password. Without a user name and password, a user is not granted access. See the *Foundry Security Guide* for information on setting up user names and passwords on Foundry devices.

If you enable empty password logins, users are not prompted for a password when they log in. Any user with an SSH client can log in without being prompted for a password.

EXAMPLE:

To enable empty password logins:

```
BigIron(config)# ip ssh permit-empty-passwd yes
```

Syntax: ip ssh permit-empty-passwd no | yes

Possible values: N/A

Default value: Disabled

ip ssh port

Changes the TCP port used for SSH. By default, SSH traffic occurs on TCP port 22. You can change this port number.

EXAMPLE:

The following command changes the SSH port number to 2200:

```
BigIron(config)# ip ssh port 2200
```

Note that if you change the default SSH port number, you must configure SSH clients to connect to the new port. Also, you should be careful not to assign SSH to a port that is used by another service. If you change the SSH port number, Foundry recommends that you change it to a port number greater than 1024.

Syntax: ip ssh port <number>

Possible values: a valid TCP port number

Default value: 22

ip ssh pub-key-file

Causes a public key file to be loaded onto the Foundry device.

EXAMPLE:

To cause a public key file called pkeys.txt to be loaded from the Management IV module's PCMCIA flash card each time the Foundry device is booted, enter the following command:

```
BigIron(config)# ip ssh pub-key-file slot1 pkeys.txt
```

Syntax: [no] ip ssh pub-key-file slot1 | slot2 <filename>

To cause a public key file called pkeys.txt to be loaded from a TFTP server each time the Foundry device is booted, enter a command such as the following:

```
BigIron(config)# ip ssh pub-key-file tftp 192.168.1.234 pkeys.txt
```

Syntax: [no] ip ssh pub-key-file tftp <tftp-server-ip-addr> <filename>

To reload the public keys from the file on the TFTP server or PCMCIA flash card, enter the following command:

```
BigIron(config)# ip ssh pub-key-file reload
```

Syntax: [no] ip ssh pub-key-file reload

To make the public keys in the active configuration part of the startup-config file, enter the following commands:

```
BigIron(config)# ip ssh pub-key-file flash-memory
BigIron(config)# write memory
```


Syntax: [no] ip ssh pub-key-file flash-memory

Possible values: N/A

Default value: N/A

ip ssh rsa-authentication

Disables or re-enables RSA challenge-response authentication.

EXAMPLE:

To disable RSA challenge-response authentication:

```
BigIron(config)# ip ssh rsa-authentication no
```

Syntax: [no] ip ssh rsa-authentication yes | no

Possible values: yes or no

Default value: RSA challenge-response authentication is enabled by default.

ip ssh scp

Disables or re-enables Secure Copy (SCP).

EXAMPLE:

To disable SCP:

```
BigIron(config)# ip ssh scp disable
```

Syntax: [no] ip ssh scp disable | enable

Possible values: disable or enable

Default value: SCP is enabled by default.

NOTE: If you disable SSH, SCP is also disabled.

ip ssh source-interface

Configures the device to use the lowest-numbered IP address configured on an interface as the source for all SSH packets from the device. The software uses the lowest-numbered IP address configured on the interface as the source IP address for the packets.

EXAMPLE:

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

```
BigIron(config)# int ve 1
BigIron(config-vif-1)# ip address 10.0.0.3/24
BigIron(config-vif-1)# exit
BigIron(config)# ip ssh 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 SSH packets from the Layer 3 Switch.

Syntax: ip ssh source-interface atm <portnum>.<subif> | ethernet <portnum> | loopback <num> | ve <num>

Possible values: see above

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

ip ssh timeout

Changes the SSH timeout value. When the SSH server attempts to negotiate a session key and encryption method with a connecting client, it waits a maximum of 120 seconds for a response from the client. If there is no response from the client after 120 seconds, the SSH server disconnects.

EXAMPLE:

```
BigIron(config)# ip ssh timeout 60
```

Syntax: ip ssh timeout <seconds>

Possible values: 1 – 120 second

Default value: 120 seconds

ip strict-acl-tcp

Enables the strict ACL TCP mode.

By default, when you use ACLs to filter TCP traffic, the Foundry device does not compare all TCP packets against the ACLs. Instead, the device compares TCP control packets against the ACLs, but not data packets. Control packets include packet types such as SYN (Synchronization) packets, FIN (Finish) packets, and RST (Reset) packets.

In normal TCP operation, TCP data packets are present only if a TCP control session for the packets also is established. For example, data packets for a session never occur if the TCP SYN for that session is dropped. Therefore, by filtering the control packets, the Foundry device also implicitly filters the data packets associated with the control packets. This mode of filtering optimizes forwarding performance for TCP traffic by forwarding data packets without examining them. Since the data packets are present in normal TCP traffic only if a corresponding TCP control session is established, comparing the packets for the control session to the ACLs is sufficient for filtering the entire session including the data.

However, it is possible to generate TCP data packets without corresponding control packets, in test or research situations for example. In this case, the default ACL mode does not filter the data packets, since there is no corresponding control session to filter. To filter this type of TCP traffic, use the strict ACL TCP mode. This mode compares all TCP packets to the configured ACLs, regardless of whether the packets are control packets or data packets. If the ACLs permit the packet, the device creates a session entry for forwarding other TCP packets with the same Layer 3 and Layer 4 addresses.

NOTE: Regardless of whether the strict mode is enabled or disabled, the device always compares TCP control packets against the configured ACLs before creating a session entry for forwarding the traffic.

NOTE: If the device's configuration currently has ACLs associated with interfaces, remove the ACLs from the interfaces before changing the ACL mode.

EXAMPLE:

To enable the strict ACL TCP mode, enter the following command at the global CONFIG level of the CLI:

```
BigIron(config)# ip strict-acl-tcp
```

Syntax: [no] ip strict-acl-tcp

This command configures the device to compare all TCP packets against the configured ACLs before forwarding them.

To disable the strict ACL mode and return to the default ACL behavior, enter the following command:

```
BigIron(config)# no ip strict-acl-tcp
```

Possible values: N/A

Default value: Disabled

ip strict-acl-udp

Configures the device to send all UDP packets to the CPU for ACL processing.

By default, when you use ACLs to filter UDP traffic, the Foundry device does not compare all UDP packets against the ACLs. Instead, the device compares the source and destination information against entries in the session table. The session table contains forwarding entries based on Layer 3 and Layer 4 information.

- If the session table contains a matching entry, the device forwards the packet, assuming that the first packet the device received that contains the same address information was permitted by the ACLs.
- If the session table does not contain a matching entry, the device sends the packet to the CPU, where the software compares the packet against the ACLs. If the ACLs permit the packet (explicitly by a permit ACL entry or implicitly by the absence of a deny ACL entry), the CPU creates a session table entry for the packet's forwarding information and forwards the packet.

For tighter control, the software provides the strict ACL UDP mode. When you enable strict UDP processing, the device sends every UDP packet to the CPU and compares the packet against the configured ACLs.

NOTE: If the device's configuration currently has ACLs associated with interfaces, remove the ACLs from the interfaces before changing the ACL mode.

EXAMPLE:

To enable the strict ACL UDP mode, enter the following command at the global CONFIG level of the CLI:

```
BigIron(config)# ip strict-acl-udp
```

Syntax: [no] ip strict-acl-udp

This command configures the device to compare all UDP packets against the configured ACLs before forwarding them.

To disable the strict ACL mode and return to the default ACL behavior, enter the following command:

```
BigIron(config)# no ip strict-acl-udp
```

Possible values: N/A

Default value: Disabled

ip tacacs source-interface

Configures the device to use the first IP address configured on an interface as the source for all TACACS/TACACS+ packets from the device. The software uses the lowest-numbered IP address configured on the interface as the source IP address for the packets.

EXAMPLE:

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 a the following:

```
BigIron(config)# int ve 1
BigIron(config-vif-1)# ip address 10.0.0.3/24
BigIron(config-vif-1)# exit
BigIron(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 atm <portnum>.<subif> | ethernet <portnum> | loopback <num> | ve <num>

Possible values: see above

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

ip tcp burst-normal

Causes the Foundry device to drop TCP SYN packets when excessive numbers are encountered, as is the case when the device is the victim of a TCP SYN attack. This command allows you to set threshold values for TCP SYN packets targeted at the router and drop them when the thresholds are exceeded.

EXAMPLE:

In the following example, if the number of TCP SYN packets received per second exceeds 10, the excess packets are dropped. If the number of TCP SYN packets received per second exceeds 100, the device drops all TCP SYN packets for the next 300 seconds (five minutes).

```
BigIron(config)# ip tcp burst-normal 10 burst-max 100 lockup 300
```

You can set threshold values for TCP SYN packets received on an interface and drop them when the thresholds are exceeded. For example:

```
BigIron(config)# int e 3/11
BigIron(config-if-e100-3/11)# ip tcp burst-normal 10 burst-max 100 lockup 300
```

Syntax: ip tcp burst-normal <value> burst-max <value> lockup <seconds>

The burst-normal value can be from 1 – 100000.

The burst-max value can be from 1 – 100000.

The lockup value can be from 1 – 10000.

The number of incoming TCP SYN packets per second are measured and compared to the threshold values as follows:

- If the number of TCP SYN packets exceeds the burst-normal value, the excess TCP SYN packets are dropped.
- If the number of TCP SYN packets exceeds the burst-max value, all TCP SYN packets are 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.

Possible values: The burst-normal and burst-max values can be between 1 – 100000 packets. The burst-normal value must be smaller than the burst-max value. The lockup value can be between 1 – 10000 seconds.

Default value: N/A

ip tcp keepalive

Changes TCP keepalive parameters.

The Foundry device sends keepalive messages to another device if there is no activity on a given TCP session with the device for a certain period. For example, if a Telnet client or BGP4 neighbor's TCP session with the Foundry device is inactive for two minutes (the default keepalive timeout), the Foundry device sends a TCP keepalive message to the device. If the device responds to the keepalive message, the Foundry device continues using the session. If the device does not respond, the Foundry device assumes that the other end is dead and terminates the session.

By default, the Foundry device sends a keepalive message when a session has been inactive for 120 seconds. The Foundry device sends up to three keepalive messages, at 60-second intervals. If the other device does not respond within 60 seconds after the last message is sent, the Foundry device terminates the session.

EXAMPLE:

```
BigIron(config)# ip tcp keepalive 300 60 5
```

Syntax: [no] ip tcp keepalive <timeout> <interval> <nummsgs>

The <timeout> parameter specifies the number of seconds the Foundry device allows a TCP session to be inactive before sending a keepalive packet. You can specify any number of seconds for the timeout. If you specify 0, TCP keepalive messages are disabled, in which case the Foundry device assumes that all open TCP sessions are active. The default timeout is 120 seconds.

The <interval> parameter specifies the number of seconds between TCP keepalive messages. You can specify any number of seconds. The default is 60 seconds.

The <nummsgs> parameter specifies the number of TCP keepalive messages the Foundry device will send. The Foundry device sends one message at each keepalive interval. If the other device does not respond within the keepalive interval after the last message is sent, the Foundry device terminates the session. You can specify any number of messages. The default is 3 messages.

Possible values: See above

Default value: timeout is 120 seconds; interval is 60 seconds; number of messages is 3

ip telnet source-interface

Configures the device to use the lowest-numbered IP address configured on an interface as the source for all Telnet packets from the device. The software uses the lowest-numbered IP address configured on the interface as the source IP address for the packets.

NOTE: When you specify a single Telnet source, you can use only that source address to establish Telnet management sessions with the Foundry device.

EXAMPLE:

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

```
BigIron(config)# int loopback 2
BigIron(config-lbif-2)# ip address 10.0.0.2/24
BigIron(config-lbif-2)# exit
BigIron(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 atm <portnum>.<subif> | ethernet <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 Layer 3 Switch.

```
BigIron(config)# interface ethernet 1/4
BigIron(config-if-1/4)# ip address 209.157.22.110/24
BigIron(config-if-1/4)# exit
BigIron(config)# ip telnet source-interface ethernet 1/4
```

Possible values: see above

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

ip tftp source-interface

Configures the device to use the lowest-numbered IP address configured on an interface 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.

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:

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

BigIron(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: ip tftp source-interface atm <portnum>.<subif> | ethernet <portnum> | loopback <num> | ve <num>

Possible values: see above

Default value: 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.

ip ttl

Sets the maximum time that a packet will live on the network.

This command applies both to Layer 3 Switches and to Layer 2 Switches (including the ServerIron).

EXAMPLE:

```
BigIron(config)# ip ttl 25
BigIron(config)# exit
BigIron# write memory
```

Syntax: ip ttl <hops>

Possible values: 1 – 255 hops

Default value: 64 hops

ipx forward-filter

Defines forward filters for IPX routes.

IPX must be enabled on the Foundry Layer 2 Switch (excluding FastIron Workgroup) or Layer 3 Switch and a network number and frame type defined for each IPX interface, for this command to be operational.

EXAMPLE:

```
BigIron(config)# ipx forward-filter 2 permit 1110005 451 11101050 120 any
```

Syntax: ipx forward-filter <index> permit | deny <source-network-number> | any <source-node-number> | any <destination-network-number> | any <destination-node-number> | any <destination-socket-number> | any

Possible values: up to 32 forward filters

Default value: N/A

ipx gns-round-robin

Configures the Layer 3 Switch to use round-robin to rotate among servers of a given service type when responding to GNS requests, instead of the default behavior of responding with the most recently learned server supporting the requested service.

EXAMPLE:

To enable the Layer 3 Switch to use round-robin to select servers for replies to GNS requests:

```
BigIron(config)# ipx gns-round-robin
```

Syntax: [no] ipx gns-round-robin

Possible values: N/A

Default value: N/A

ipx max-rip-entries

Defines the maximum number of IPX/RIP routes that will be stored and forwarded. IPX must be enabled on the router for this command to be operational.

NOTE: This command is only available on routers installed with release 3.0 software. In release 4.0, this parameter is set using the **system-max** command at the global CONFIG level.

EXAMPLE:

```
BigIron(config)# ipx max-rip-entries 3500
```

Syntax: ipx max-rip-entries <value>

Possible values: 64 – 8192

Default value: 2048

ipx max-sap-entries

Defines the number of IPX/SAP server entries that will be stored and forwarded. IPX must be enabled on the router for this command to be operational.

NOTE: This command is only seen on routers installed with release 3.0 software. In release 4.0, this parameter is set using the **system-max** command at the global CONFIG level.

EXAMPLE:

```
BigIron(config)# ipx max-sap-entries 4000
```

Syntax: ipx max-sap-entries <value>

Possible values: 128 – 8192

Default value: 4096

ipx netbios-allow

Enables NetBIOS broadcasts (type 20) to be routed over IPX. IPX must be enabled on the router and a network number and frame type defined for each IPX interface.

EXAMPLE:

```
BigIron(config)# ipx netbios-allow
```

Syntax: ipx netbios-allow

Possible values: N/A

Default value: disabled

ipx rip-filter

Defines IPX/RIP filters for the router. IPX must be enabled on the router for this command to be operational.

EXAMPLE:

```
BigIron(config)# ipx rip-filter 2 permit 11005000 fffff00
```

-OR-

```
BigIron(config)# ipx rip-filter 2 permit any any
```

Syntax: ipx rip-filter <index> permit | deny <network-number> | any <network-mask> | any

Possible values: up to 32 RIP filters can be defined for a router

Default value: N/A

ipx rip-filter-group

Allows a group of filters to be applied globally to all IPX interfaces at the Global Level, or individually to an IPX interface at the Interface Level. The filter can be applied to either incoming or outgoing traffic.

EXAMPLE:

To apply previously defined filters 1, 2, 3, and 10 to all incoming IPX RIP routes across all interfaces, enter the following command:

```
BigIron(config)# ipx rip-filter-group in 1 2 3 10
```

To apply filters on an individual interface (e.g. interface 4/11) basis versus globally, enter the following:

```
BigIron(config)# int e 4/11
```

```
BigIron(config-if-4/11)# ipx rip-filter-group in 1 2 3 10
```

Syntax: ipx rip-filter-group in | out <index>

Possible values: in | out, filter I ds

Default value: disabled

ipx sap-access-list

Configures access lists for filtering Service Advertisement Protocol (SAP) replies sent on a Layer 3 Switch's IPX interfaces. You configure IPX SAP access lists on a global basis, then apply them to the IPX inbound or outbound filter group on specific interfaces. You can configure up to 32 access lists. The same access list can be applied to multiple interfaces.

EXAMPLE:

```
BigIron(config-ipx-router)# ipx sap-access-list 10 deny efef.1234.1234.1234
```

Syntax: [no] ipx sap-access-list <num> deny | permit <network>[.<node>] [<network-mask>.<node-mask>] [<service-type>[<server-name>]]

Possible values: The <num> parameter specifies the access list number and can be from 1 – 32.

The **deny** | **permit** parameter specifies whether the Layer 3 Switch allows the SAP update or denies it.

The <network>[.<node>] parameter specifies the IPX network. Optionally, you also can specify a specific node (host) on the network. The <network> parameter can be an eight-digit hexadecimal number from 1 – FFFFFFFE. To specify all networks ("any"), enter **–1** as the network number. If the network number has leading zeros, you do not need to specify them. For example, you can specify network 0000abab as "**abab**".

The node is a 48-bit value represented by three four-digit numbers joined by periods; for example, 1234.1234.1234.

The [<network-mask>.<node-mask>] parameter lets you specify a comparison mask for the network and node. The mask consists of zeros (0) and ones (f). Ones indicate significant bits. For example, to configure a mask that matches on network abcdefxx, where xx can be any value and the node address can be any value, specify the following mask: ffffff00.0000.0000.0000

The **in** | **out** parameter of the **ipx sap-filter-group** command specifies whether the ACLs apply to incoming traffic or outgoing traffic.

Default value: N/A

ipx sap-filter

Defines IPX/SAP filters for all IPX interfaces on the router. The IPX network number and frame type must be defined for the interfaces for this command to be operational.

EXAMPLE:

```
BigIron(config)# ipx sap-filter 5 permit any server1
```

-OR-

```
BigIron(config)# ipx sap-filter 5 permit 0004 any
```

Syntax: ipx sap-filter <index> permit | deny <server-type> | any <server-name> | any

Possible values: Filter IDs

Default value: Disabled

ipx sap-filter-group

Allows a group of defined IPX/SAP filters to be applied either globally (at the Global Level) or individually (at the Interface Level) to IPX interfaces on the router.

The filter can be applied to either incoming or outgoing traffic.

EXAMPLE:

To apply previously defined filters 2, 3, and 10 to all incoming IPX SAP server traffic across all interfaces, enter the following command:

```
BigIron(config)# ipx sap-filter-group in 2 3 5
```

To apply filters on an individual interface basis instead of a global basis (for example, apply a filter to interface 4/11), enter the following:


```
BigIron(config)# int 4/11
BigIron(config-if)# ipx sap-filter-group in 2 3 5
```

Syntax: ipx sap-filter-group in | out <index>

Possible values: in or out, defined filter indexes

Default value: N/A

jumbo

NOTE: This command applies to software releases prior to 07.6.02. In software releases 07.6.02 and later, on IronCore devices only, this command has been changed to **jumbo1920**. See page **6-81** for a description of the **jumbo1920** command.

Globally changes the MTU for packets between ATM or POS and Ethernet interfaces to 1920 bytes.

NOTE: This command applies to IronCore and JetCore devices, and applies only to traffic between ATM or POS and Ethernet ports on the device.

The global MTU size determines the fragmentation boundary for packets received from an ATM or POS interface that need to be forwarded to an Ethernet interface. For example, if the global MTU is 1500 and the ATM MTU is 9180, the device fragments a 9180-byte packet received on the ATM interface into seven fragments and forwards the fragments to a destination Ethernet port. If the global MTU is 1920, the device fragments a 9180-byte packet into only five fragments. If ATM receives a packet of up to 1920 bytes, it will not fragment the packet. If ATM receives a packet of more than 1920 bytes, then it fragments the packet into chunks of 1872 bytes until the last fragment is sent.

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.

NOTE: This command does not affect the global MTU for traffic between Ethernet ports. The Ethernet MTU is still listed as 1492 or 1500 bytes.

NOTE: Regardless of the setting of the global MTU, the MTU for ATM PVCs is configurable up to 9180 bytes and is 4470 bytes by default.

EXAMPLE:

To change the global MTU to 1920, enter the following commands:

```
BigIron(config)# jumbo
BigIron(config)# write memory
BigIron(config)# end
BigIron# reload
```

Syntax: [no] jumbo

Possible values: N/A

Default value: Disabled

jumbo1920

NOTE: This command applies to IronCore devices running software release 07.6.02 or later. Prior to 07.6.02, this command was called **jumbo**. See page **6-80** for a description of the **jumbo** command.

Configures some Ethernet interfaces on an IronCore device to have an MTU of 1518 bytes and others to have an MTU of 1920 bytes. Software releases prior to 07.6.02 supported an MTU of 1920 bytes, but it had to be applied globally to all interfaces on the device.

The fragmentation boundary for packets received from an ATM interface that need to be forwarded to an Ethernet interface is based on the MTU of the outbound Ethernet interface. For example, if the MTU on an Ethernet interface is 1518, and the ATM MTU is 9180, the device fragments the IP payload of a 9180-byte packet received on the ATM interface into seven fragments and forwards the fragments to the destination Ethernet port. If the MTU on another Ethernet interface is 1920, the device fragments the IP payload of a 9180-byte packet into only five fragments. If ATM receives a packet of up to 1902 bytes, it will not fragment the packet. If ATM receives a packet of more than 1902 bytes, then the device fragments the packet into chunks of 1872 bytes until the last fragment is sent.

For VE traffic, the fragmentation boundary is based on the MTU of the physical outbound interface. No fragmentation is performed on Layer 2 frames. Layer 2 frames that exceed the MTU of the outbound interface are dropped.

EXAMPLE:

To change the default MTU for Ethernet interfaces to 1920 bytes, enter the following command:

```
BigIron(config)# jumbo1920
BigIron(config)# write memory
BigIron(config)# end
BigIron# reload
```

Syntax: [no] jumbo1920

Possible values: N/A

Default value: Disabled

link-keepalive ethernet <portnum>

Enables Uni-Directional Link Detection (UDLD) on a trunk group.

EXAMPLE:

To enable UDLD on a port, enter a command such as the following at the global CONFIG level of the CLI:

```
BigIron(config)# link-keepalive ethernet 1/1
```

To enable the feature on a trunk group, enter commands such as the following:

```
BigIron(config)# link-keepalive ethernet 1/1 ethernet 1/2
BigIron(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 <portnum> [ethernet <portnum>]

Possible values: See above

Default value: Disabled

link-keepalive interval

Changes the Uni-Directional Link Detection (UDLD) interval.

EXAMPLE:

```
BigIron(config)# link-keepalive interval 3
```

Syntax: [no] link-keepalive interval <num>

The <num> parameter specifies how often the ports send a UDLD packet.

Possible values: 1 – 60, in 100 ms increments

Default value: 5 (500 ms)

link-keepalive retries

Changes the maximum number of keepalive attempts the Uni-Directional Link Detection (UDLD) feature makes before taking down a port.

EXAMPLE:

```
BigIron(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.

Possible values: 3 – 10

Default value: 5

lock-address ethernet

Allows you to limit the number of devices that have access to a specific port. Access violations are reported by SNMP traps.

EXAMPLE:

```
BigIron(config)# lock-address eth 4/11 addr 15
```

Syntax: lock-address ethernet <portnum> [addr-count <number>]

Possible values: Address count: 1 – 2,048

Default value: Address count: 8

logging

You can save SNMP traps locally to an event log on the Layer 2 Switch or Layer 3 Switch by turning this feature on. You also can configure the device to use up to six third-party Syslog servers and modify the message level and facility using this command. In addition, you can change the number of log messages the local Syslog buffer will retain.

EXAMPLE:

To disable logging of SNMP traps to a locally saved event log, enter the following command:

```
BigIron(config)# no logging on
```

To re-enable logging, enter the following command:

```
BigIron(config)# logging on
```

Syntax: [no] logging on [<udp-port>]

Possible values: See above

Default value: on (enabled); UDP port 514

EXAMPLE:

To specify two third-party Syslog servers to receive Syslog messages in addition to the device's local Syslog buffer, enter commands such as the following. You can specify up to six servers.

```
BigIron(config)# logging 10.0.0.99
BigIron(config)# logging 209.157.23.69
```

Syntax: logging <ip-addr> | <server-name>

NOTE: If you specify more than one Syslog server, the Foundry device uses the same facility and message level for messages to all the servers.

Possible values: N/A

Default value: N/A

EXAMPLE:

To change the logging facility from the default facility user to local7, enter the following command:

```
BigIron(config)# logging local7
```

Syntax: logging facility <facility-name>

Possible values:

- **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

Default value: user

EXAMPLE:

To disable logging of debugging and informational messages, enter the following commands:

```
BigIron(config)# no logging buffered debugging
BigIron(config)# no logging buffered informational
```

Syntax: [no] logging buffered <level>

Possible values: The <level> can be **emergencies**, **alerts**, **critical**, **errors**, **warnings**, **notifications**, **informational**, or **debugging**.

Default value: All message levels are enabled by default. You can disable message levels individually.

EXAMPLE:

To change the local buffer capacity from the default 50 to 100, enter the following command:

```
BigIron(config)# logging buffered 100
```

Syntax: logging buffered <num-entries>

Possible values: For releases 07.6.03 and later, <num entries> can be from 1 - 1000 on on all devices. For releases prior to 07.6.03, <num-entries> can be from 1 – 1000 on Layer 3 Switches and from 1 – 100 on other devices. The change takes effect immediately and does not require you to reload the software.

Default value: default local buffer capacity on all devices is 50 entries.

EXAMPLE:

By default, a message is logged whenever a user logs into or out of the CLI's User EXEC or Privileged EXEC mode. If you want to disable logging of users' CLI access, enter the following command:

```
BigIron(config)# no logging enable user-login
```

Syntax: [no] logging enable user-login

Possible values: N/A

Default value: User logins are logged by default.

EXAMPLE:

To enable real-time display of Syslog messages in the CLI, enter the following command at the global CONFIG level of the CLI:

```
BigIron(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 **terminal monitor** command from the Privileged EXEC level of the session. See "terminal monitor" on page 5-48.

Possible values: N/A

Default value: Logging to the console is disabled by default.

mac-age-time

This parameter sets the aging period for ports on the device, defining how long a port address remains active in the address table.

EXAMPLE:

```
BigIron(config)# mac-age 600
```

Syntax: mac-age-time <age-time>

Possible values: The <age-time> can be 0 or a number from 67 – 65535. If you specify 0, the entries do not age.

Default value: 300 seconds

mac filter

Allows you to filter on MAC addresses.

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 the "IP Access Control Lists (ACLs) chapter in the *Foundry Enterprise Configuration and Management Guide*.

NOTE: You cannot use Layer 2 filters to filter Layer 4 information. To filter Layer 4 information, use IP access policies.

EXAMPLE:

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

```
BigIron(config)# mac filter 1 deny 3565.3475.3676 ffff.0000.0000 any etype eq 806
BigIron(config)# mac filter 1024 permit any any
BigIron(config)# int e 1/1
BigIron(config-if-1/1)# 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 define a MAC filter, the device drops Layer 2 traffic that does not match a MAC permit filter.

Syntax: mac filter <filter-num> permit | deny <src-mac> <mask> | any <dest-mac> <mask> | any etype | llc | snap eq | gt | lt | neq <frame-type>

Possible values:

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.

Use the **etype** | **llc** | **snap** argument if you want to filter on information beyond the source and destination address. The MAC filter allows for you to filter on the following encapsulation types:

- **etype** (Ethertype) – a two byte field indicating the protocol type of the frame. This can range from 0x0600 to 0xFFFF.
- **llc** (IEEE 802.3 LLC1 SSAP and DSAP) – a two byte sequence providing similar function as the EtherType but for an IEEE 802.3 frame.
- **snap** (IEEE 802.3 LLC1 SNAP) – a specific LLC1 type packet.

To determine which type of frame is used on your network, use a protocol analyzer. If byte 12 of an Ethernet packet is equal to or greater than 0600 (hex), it is an Ethernet framed packet. Any number below this indicates an IEEE 802.3 frame (byte 12 will now indicate the length of the data field). Some well-known Ethernet types are 0800 (TCP/IP), 0600 (XNS), and 8137 (Novell Netware). Refer to RFC 1042 for a complete listing of EtherTypes.

For IEEE 802.3 frame, you can further distinguish the SSAP and DSAP of LLC header. Some well-known SAPs include: FE (OSI), F0 (NetBIOS), 42 (Spanning Tree BPDU), and AA (SNAP). Usually the DSAP and SSAP are the same.

NOTE: You must type in both bytes, otherwise the software will fill the field, left justified with a 00. Refer to RFC 1042 for a complete listing of SAP numbers.

SNAP is defined as an IEEE 802.3 frame with the SSAP, DSAP, and control field set to AA, AA, and 03. Immediately following these is a five-byte SNAP header. The first three bytes in this header are not used by the MAC filters. However, the next two bytes usually are set to the EtherType, so you can define the EtherType inside the SNAP header that you want to filter on.

The eq | gt | lt | neq argument specifies the possible operator: eq (equal), gt (greater than), lt (less than) and neq (not equal).

The <frame-type> argument is a hexadecimal number for the frame type. For example, the hex number for ARP is 806.

Default value: N/A

Additional Examples of Layer 2 MAC Filter Definitions

```
BigIron(config)# mac filter 1 permit any any etype eq 0800
```

This filter configures the device to permit (forward) any inbound packet with the Ethertype field set to 0800 (IP).

```
BigIron(config)# mac filter 2 deny 0080.0020.000 ffff.ffff.0000 any etype eq 0800
```

This filter configures the device to deny an inbound packet with the first four bytes set to 0800.0020.xxxx and an EtherType field set to 0800 (IP). The destination field does not matter.

```
BigIron(config)# mac filter 3 deny any 00e0.5200.1234 ffff.ffff.ffff snap eq 0800
```

This filter configures the device to deny any inbound IEEE 802.3 packet with a destination set to 00e0.5200.1234 and a SNAP EtherType set to 0800. The source address does not matter.

```
BigIron(config)# mac filter 32 permit any any
```

This filter permits all packets. This filter is used as the last filter assigned in a filter-group that has previous deny filters in the group.

Abbreviating the Address or Mask

Address and Mask abbreviations are allowed. However, be careful when configuring them. The default fill character is a 0 and it will fill a byte range as left-justified. This applies only to the MAC address and mask. A range of frame types cannot be filtered. Each frame type must be entered. Here are some examples.

```
BigIron(config)# mac filter 1 deny 0800.0700 ffff.ff00 any
```

This command expands to the following: **mac filter 1 deny 0800.0700.0000 ffff.ff00.0000**

The filter shown above denied forwarding of an inbound frame that has the source address set to 080007 as the first three bytes. All other information is not significant.

Here is another example of the fill feature.

```
BigIron(config)# mac filter 2 deny 0260.8C00.0102 0.0.ffff any
```

This command expands to the following: **mac filter 1 deny 0260.8C00.0102 0000.0000.ffff any**

Since the fill character is 0's and the fill is left justified, certain filters will not allow for abbreviations. For example, suppose you want to deny an inbound packet that contained a broadcast destination address. Enter the following command:

```
BigIron(config)# mac filter 5 deny any ff ff
```

This command contains a destination of address all F's and mask of F's. The command expands to the following:

```
BigIron(config)# mac filter 1 deny any 00ff.0000.0000 00ff.0000.0000
```

Here is another example for DSAP and SSAP.

```
BigIron(config)# mac filter 10 deny any any llc eq F0
```

This command expands to the following: **mac filter 2 deny any any llc eq 00f0**

If you want to filter on both the SSAP and DSAP, then the following example shows this:

```
BigIron(config)# mac filter 4 deny any 0020.0010.1000 ffff.ffff.0000 llc eq e0e0
```


mac filter log-enable

Enables logging of packets that are denied by Layer 2 MAC filters. When you enable this feature, the device generates Syslog entries and SNMP traps for denied packets.

See “show logging” on page 43-93 for information about log entries generated by this feature.

EXAMPLE:

```
BigIron(config)# mac filter log-enable
```

Syntax: mac filter log-enable

Possible values: N/A

Default value: Disabled

mcast-hw-replic-disable

Disables hardware multicast forwarding for tagged ports.

NOTE: This command applies to Layer 3 multicast traffic on JetCore Layer 3 Switches only. All Layer 2 multicast traffic on JetCore or IronCore devices is forwarded by the CPU.

EXAMPLE:

```
BigIron(config)# mcast-hw-replic-disable
BigIron(config)# write memory
BigIron(config)# end
BigIron# reload
```

Syntax: [no] mcast-hw-replic-disable

NOTE: You must save the configuration change and reload the software to place the change into effect.

Possible values: N/A

Default value: Enabled on JetCore Layer 3 devices. Disabled on all other devices.

mcast-hw-replic-oar

Enables hardware forwarding of multicast traffic in one-armed-router configurations.

NOTE: This command applies to Layer 3 multicast traffic on JetCore Layer 3 Switches only. All Layer 2 multicast traffic on JetCore or IronCore devices is forwarded by the CPU. This command is supported in software release 07.6.01 and later.

By default, JetCore devices running software releases later than 07.5.05D forward multicast traffic destined to multiple VLANs on tagged ports, without the need to send the traffic to the CPU for forwarding. However, this default behavior does not apply to one-armed-router configurations, in which traffic received on a port is destined to another VLAN on the same port.

For example, assume that ports 1/1 and 1/2 are members of two port-based VLANs (10 and 20), and each VLAN has a virtual routing interface. If port 1/1 receives multicast traffic from VLAN 10 and needs to forward the traffic to the virtual routing interface on VLAN 20, the device forwards the traffic to port 1/2 in hardware but uses the CPU to process the same traffic for forwarding back onto port 1/1.

You can enable the device to forward multicast traffic in hardware even in one-armed-router configurations. When you enable this support, the devices still forward multicast traffic between ports in hardware.

NOTE: You cannot use sFlow or port monitoring and hardware forwarding of multicast traffic in one-armed-router configurations on the same device. If you plan to enable hardware forwarding of multicast traffic in one-armed-router configurations, you must first make sure that sFlow and port monitoring are disabled on all ports. If either of these features is enabled when you enable multicast traffic in one-armed-router configurations, you may get unexpected results.

EXAMPLE:

```
BigIron(config)# mcast-hw-replic-oar
BigIron(config)# write memory
BigIron(config)# end
BigIron# reload
```

Syntax: [no] mcast-hw-replic-oar

NOTE: You must save the configuration change and reload the software to place the change into effect.

NOTE: For hardware forwarding of multicast traffic in one-armed-router configurations to take effect, hardware forwarding of multicast traffic on tagged ports must be enabled. If you disable hardware forwarding of multicast traffic on tagged ports (by entering the **mcast-hw-replic-disable** command), the **mcast-hw-replic-oar** command does not take effect.

Possible values: N/A

Default value: Disabled

mirror-port

Enables and assigns a specific port to operate as a mirror port for other ports. After you enable the feature, you can connect an external traffic analyzer to the port for traffic analysis.

Use the following considerations when configuring mirroring for inbound traffic on a Chassis device or a FastIron 4802 (Stackable device). The guidelines are applicable whether you configure multiple mirror ports or just one mirror port.

Guidelines for Chassis devices:

- Configure only one mirror port to monitor input traffic on a given module. If you configure multiple mirror ports on the same module, the inbound traffic for all the monitored ports on the module is sent to all the mirror ports on the same module. For example, if you configure ports 1/1 and 1/13 as mirror ports, then enable monitoring of inbound traffic on ports 1/2 and 1/14, the traffic from both ports is mirrored to both the mirror ports, 1/1 and 1/13. This occurs regardless of the mirror ports you assign to the monitor ports.
- When inbound traffic on a monitored port on one module is switched normally to another module, the switched traffic will be mirrored to the mirror ports on the other module. For example, if inbound traffic on a monitored port on the module in slot 1 is switched to the module in slot 2, mirror ports on the module in slot 2 will receive copies of the traffic.These guidelines do not apply to outbound traffic.

These guidelines do not apply to outbound traffic.

Guidelines for the FastIron 4802:

- A mirror port and all the ports it is monitoring must be on the same IPC. IPC 1 manages ports 1 – 24 and port 49. IPC 2 manages ports 25 – 48 and port 50.
- In the current release, you cannot monitor inbound traffic on Gigabit ports 49 – 50 if they are configured as a trunk group. This restriction does not apply if the ports are not configured as a trunk group.
- If you configure more than one mirror port to monitor inbound traffic on the same IPC, each of the mirror ports on the IPC receives all the inbound traffic from all the monitored ports on the same IPC.
- When inbound traffic on a monitored port on one IPC is switched normally to another IPC, the switched traffic will be mirrored to the mirror ports on the other IPC. For example, if inbound traffic on a monitored port on IPC 1 is switched to IPC 2, mirror ports on IPC 2 will receive copies of the traffic.

These guidelines do not apply to monitoring outbound traffic. You can monitor traffic between IPCs and use multiple mirror ports.

EXAMPLE:

To assign port 1 on module 1 as the mirror port and port 5 on the same module as the port to be monitored, enter the following:


```
BigIron(config)# mirror-port ethernet 1/1
BigIron(config)# interface ethernet 1/5
BigIron(config-if-1/5)# monitor both
```

Syntax: [no] mirror-port ethernet <portnum>

The <portnum> parameter specifies the port. You can configure up to 64 mirror ports on a Chassis device and up to 50 mirror ports on a FastIron 4802.

Possible values: See above

Default value: Not configured

module

Adds a hardware module to a Chassis device.

EXAMPLE:

To add an 8-port Gigabit Ethernet management module to slot 3 in a BigIron 8000, enter the following command:

```
BigIron(config)# module 3 bi-8-port-gig-management-module
```

Syntax: module <slot-num> <module-type>

The <slot-num> parameter indicates the chassis slot number.

- Slots on a 4-slot chassis are numbered 1 – 4, from top to bottom.
- Slots on an 8-slot chassis are numbered 1 – 8, from left to right.
- Slots on a 15-slot chassis are numbered 1 – 15, from left to right.

The <module-type> parameter specifies the module. For a list of the valid module types, enter **module** <slot-num> **?** at the CLI prompt.

Possible values: see above

Default value: N/A

multicast filter

Configures a Layer 2 filter for multicast packets. You can filter on all multicast packets or on specific multicast groups.

EXAMPLE:

To configure a Layer 2 multicast filter to filter all multicast groups, then apply the filter to ports 2/4, 2/5, and 2/8, enter the following commands:

```
BigIron(config)# multicast filter 1 any
BigIron(config-mcast-filter-id-1)# exclude-ports ethernet 2/4 to 2/5 ethernet 2/8
BigIron(config-mcast-filter-id-1)# write memory
```

EXAMPLE:

To configure a multicast filter to block all multicast traffic destined for multicast addresses 0100.5e00.5200 – 0100.5e00.52ff on port 4/8, enter the following commands:

```
BigIron(config)# multicast filter 2 any 0100.5e00.5200 ffff.ffff.ff00
BigIron(config-mcast-filter-id-2)# exclude-ports ethernet 4/8
BigIron(config-mcast-filter-id-2)# write memory
```

The software calculates the range by combining the mask with the multicast address. In this example, all but the last two bits in the mask are “significant bits” (ones). The last two bits are zeros and thus match on any value.

Syntax: [no] multicast filter <filter-ID> any | ip udp mac <multicast-address> | **any** [mask <mask>] [vlan <vlan-id>]

The parameter values are the same as for the **broadcast filter** command (see “broadcast filter” on page 6-17). In addition, the **multicast filter** command requires the **mac** <multicast-address> | **any** parameter, which specifies the multicast address. Enter **mac any** to filter on all multicast addresses. Enter **mac** followed by a specific multicast address to filter only on that multicast address.

To filter on a range of multicast addresses, use the **mask** <mask> parameter. For example, to filter on multicast groups 0100.5e00.5200 – 0100.5e00.52ff, use mask ffff.ffff.ff00. The default mask matches all bits (is all Fs). You can leave the mask off if you want the filter to match on all bits in the multicast address.

Possible values: see above

Default value: N/A

multicast limit

Specifies the maximum number of multicast packets the device can forward each second. By default the device sends multicasts and all other traffic at wire speed and is limited only by the capacities of the hardware. However, if other devices in the network cannot handle unlimited multicast traffic, this command allows you to relieve those devices by throttling the multicasts at the Foundry device.

NOTE: The multicast limit does not affect broadcast or unicast traffic. However, you can use the **broadcast limit** and **unknown-unicast limit** commands to control these types of traffic. See "broadcast limit" on page 6-18 and "unknown-unicast limit" on page 6-138.

EXAMPLE:

```
BigIron(config)# multicast limit 30000
```

Syntax: multicast limit <num>

Possible values: 0 – 4294967295; if you specify 0, limiting is disabled.

Default value: N/A

no

Disables other commands. To disable a command, place the word **no** before the command.

password-change

Allows you to define those access points from which the system password can be defined. Options are **cli**, **console-cli**, **telnet-cli**, or **any**. The **any** option allows the password to be modified from a serial port, Telnet session, or through IronView, at any level of the user interface.

EXAMPLE:

To allow password changes from a serial port console connection only, enter the following command:

```
BigIron(config)# password-change console-cli
```

Syntax: password-change cli | console-cli | telnet-cli | any

Possible values: cli, console-cli, telnet-cli, or any

Default value: None

perf-mode

Allows you to define the performance mode as 'high' to allow flow control to activate at an earlier stage, when heavy congestion exists on the network. This feature must be saved to memory and the system reset before it becomes active.

EXAMPLE:

```
BigIron(config)# perf-mode hi
```

Syntax: perf-mode normal | hi

Possible values: normal | hi

Default value: normal

ping

Verifies connectivity to a Foundry Layer 2 Switch or Layer 3 Switch or other device. The command performs an ICMP echo test to confirm connectivity to the specified device.

NOTE: If you address the ping to the IP broadcast address, the device lists the first four responses to the ping.

EXAMPLE:

```
BigIron(config)# ping 192.22.2.33
```

Syntax: ping <ip addr> | <hostname> [source <ip addr>] [count <num>] [timeout <msec>] [ttl <num>] [size <byte>] [quiet] [numeric] [no-fragment] [verify] [data <1-to-4 byte hex>] [brief]

The only required parameter is the IP address or host name of the device.

NOTE: If the device is a Foundry Layer 2 Switch or Layer 3 Switch, you can use the host name only if you have already enabled the Domain Name Server (DNS) resolver feature on the device from which you are sending the ping. See the "Configuring IP" chapter of the *Foundry Enterprise Configuration and Management Guide*.

The **source** <ip addr> specifies an IP address to be used as the origin of the ping packets.

The **count** <num> parameter specifies how many ping packets the device sends. You can specify from 1 – 4294967296. The default is 1.

The **timeout** <msec> parameter specifies how many milliseconds the Foundry device waits for a reply from the pinged device. You can specify a timeout from 1 – 4294967296 milliseconds. The default is 5000 (5 seconds).

The **ttl** <num> parameter specifies the maximum number of hops. You can specify a TTL from 1 – 255. The default is 64.

The **size** <byte> parameter specifies the size of the ICMP data portion of the packet. This is the payload and does not include the header. You can specify from 0 – 4000. The default is 16.

The **no-fragment** parameter turns on the "don't fragment" bit in the IP header of the ping packet. This option is disabled by default.

The **quiet** parameter hides informational messages such as a summary of the ping parameters sent to the device and instead only displays messages indicating the success or failure of the ping. This option is disabled by default.

The **verify** parameter verifies that the data in the echo packet (the reply packet) is the same as the data in the echo request (the ping). By default the device does not verify the data.

The **data** <1 – 4 byte hex> parameter lets you specify a specific data pattern for the payload instead of the default data pattern, "abcd", in the packet's data payload. The pattern repeats itself throughout the ICMP message (payload) portion of the packet.

NOTE: For numeric parameter values, the CLI does not check that the value you enter is within the allowed range. Instead, if you do exceed the range for a numeric value, the software rounds the value to the nearest valid value.

The **brief** parameter causes ping test characters to be displayed. The following ping test characters are supported:

! Indicates that a reply was received.

. Indicates that the network server timed out while waiting for a reply.

U Indicates that a destination unreachable error PDU was received.

I Indicates that the user interrupted ping.

Possible values: see above

Default value: see above

port aps | bootp

Closes UDP port 67 or 1972.

The software on a Foundry device leaves some UDP ports open by default, even when they are not in use. To enhance security, you can close the following UDP ports:

- BootP server – well known port 67
- SONET Automatic Protection Switching (APS) – well known port 1972

NOTE: Closing one of these ports will prevent the device from receiving messages for the application associated with the closed port. For example, closing the BootP server port will prevent clients from being able to use BootP to boot over the network, if they use the Foundry device to reach the BootP server.

EXAMPLE:

To close a UDP port, enter a command such as the following at the global CONFIG level of the CLI:

```
BigIron(config)# no port aps
```

This command closes the APS port.

To re-open a UDP port, enter a command such as the following:

```
BigIron(config)# port aps
```

Syntax: [no] port aps | bootp

The **aps | bootp** parameter indicates the UDP port you are closing or re-opening.

Possible values: aps or **bootp**

Default value: both UDP ports are open

port security

Enables the MAC port security feature globally on all interfaces and launches you into the global port security configuration level.

EXAMPLE:

```
BigIron(config)# port security
BigIron(config-port-security)#
```

Syntax: [no] port security

Possible values: N/A

Default value: Disabled

port-priority

Enables ToS-based QoS on a JetCore device. For information about this feature, see the "JetCore Type of Service (ToS) Based QoS" chapter in the *Foundry Enterprise Configuration and Management Guide*.

NOTE: This command is supported in release 07.6.01 and later and applies only to JetCore devices.

EXAMPLE:

```
BigIron(config)# port-priority
```

Syntax: [no] port-priority

This command enables the feature on all interfaces.

Possible values: N/A

Default value: Disabled

pos boot

Changes the boot source for the POS module.

EXAMPLE:

To change the boot source from the POS module's primary flash to its secondary flash:

```
BigIron(config)# pos boot secondary
```

Syntax: pos boot interactive | primary | secondary

Possible values: The **primary** and **secondary** parameters identify either the primary or secondary flash on the POS module.

The **interactive** parameter enables you to enter a separate command after the module comes up to boot the module from a TFTP server. If you use this method, you also need to use the **pos boot tftp**... command at the Privileged EXEC level to boot the module after the module comes up.

Default value: N/A

privilege

Augments the default access privileges for an access level. When you configure a user account, you can give the account one of three privilege levels: full access, port-configuration access, and read-only access. Each privilege level provides access to specific areas of the CLI by default:

- Full access provides access to all commands and displays.
- Port-configuration access gives access to:
  - The User EXEC and Privileged EXEC levels, and the port-specific parts of the CONFIG level
  - All interface configuration levels
- Read-only access gives access to:
  - The User EXEC and Privileged EXEC levels

EXAMPLE:

To enhance the port-configuration privilege level so users also can enter **ip** commands at the global CONFIG level (useful for adding IP addresses for multinetting), enter the following command:

```
BigIron(config)# privilege configure level 4 ip
```

In this command, **configure** specifies that the enhanced access is for a command at the global CONFIG level of the CLI. The **level 4** parameter indicates that the enhanced access is for privilege level 4 (port-configuration). All users with port-configuration privileges will have the enhanced access. The **ip** parameter indicates that the enhanced access is for the IP commands. Users who log in with valid port-configuration level user names and passwords can enter commands that begin with "ip" at the global CONFIG level.

Syntax: [no] privilege <cli-level> level <privilege-level> <command-string>

The <cli-level> parameter specifies the CLI level and can be one of the following values:

- **exec** – EXEC level; for example, BigIron`>` or BigIron`#`
- **configure** – CONFIG level; for example, BigIron`(config)#`
- **interface** – interface level; for example, BigIron`(config-if-6)#`
- **virtual-interface** – virtual-interface level; for example, BigIron`(config-vif-6)#`
- **rip-router** – RIP router level; for example, BigIron`(config-rip-router)#`
- **ospf-router** – OSPF router level; for example, BigIron`(config-ospf-router)#`
- **dvmrp-router** – DVMRP router level; for example, BigIron`(config-dvmrp-router)#`
- **pim-router** – PIM router level; for example, BigIron`(config-pim-router)#`
- **bgp-router** – BGP4 router level; for example, BigIron`(config-bgp-router)#`
- **port-vlan** – Port-based VLAN level; for example, BigIron`(config-vlan)#`
- **protocol-vlan** – Protocol-based VLAN level

The <privilege-level> indicates the privilege level you are augmenting.

The **level** parameter specifies the privilege-level. You can specify one of the following:

- **0** – Full access (super-user)
- **4** – Port-configuration access
- **5** – Read-only access

The <command-string> parameter specifies the command you are allowing users with the specified privilege level to enter. To display a list of the commands at a CLI level, enter "?" at that level's command prompt and press Return.

pvlan-preference

Allows or restricts forwarding of broadcast or unknown unicast packets by a primary private VLAN to its community and isolated VLANs.

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.

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: 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.

EXAMPLE:

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:

```
BigIron(config)# pvlan-preference broadcast flood
BigIron(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:

```
BigIron(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

Possible values: See above

Default value: Forwarding is disabled

qd-flow sink

Specifies the threshold values for flow control.

NOTE: To use this enhancement, 802.3x flow control must be enabled globally on the device. By default, 802.3x flow control is enabled on Foundry devices, but can be disabled with the **no flow-control** command.

EXAMPLE:

```
BigIron(config)# qd-flow sink 75 sunk 50 slot 4 1 1 0 0
```

Syntax: qd-flow sink <sinking-threshold> sunk <sunk-threshold> slot <slot> <flag> <flag> <flag> <flag>

Possible values:

When the <sinking-threshold> is reached, the Foundry device sends out 802.3x PAUSE frames telling the sender to stop sending traffic for a period of time.

When the <sunk-threshold> is reached, the Foundry device drops traffic at the specified priority level.

The <slot> parameter specifies the location of the module where the thresholds are to take effect.

The <flag> parameters indicate whether to drop traffic at each priority if the number of available buffers falls below the <sunk-threshold>. Each <flag> parameter represents a priority level: 0, 1, 2, 3. To drop traffic at a priority level, set its <flag> parameter to 1. For example, to drop traffic at priority 0 and 1, enter 1 1 0 0 for the <flag> parameters. By default, the lowest priority traffic is dropped.

Default value: N/A

qos map dscp-priority

Configures QoS by changing the DSCP forwarding priority mappings for DSCP ranges.

NOTE: This command is supported only on the NetIron stackable Layer 3 Switch, and is available only in software release 07.1.16 or higher. The command is not supported in software release 07.2.x.

EXAMPLE:

To change the DSCP to forwarding priority mappings for all the DSCP ranges, enter commands such as the following at the global CONFIG level of the CLI:

```
NetIron(config)# qos map dscp-priority 0 2 3 4 to 1
NetIron(config)# qos map dscp-priority 8 to 5
NetIron(config)# qos map dscp-priority 16 to 4
NetIron(config)# qos map dscp-priority 24 to 2
NetIron(config)# qos map dscp-priority 32 to 0
NetIron(config)# qos map dscp-priority 40 to 7
NetIron(config)# qos map dscp-priority 48 to 3
NetIron(config)# qos map dscp-priority 56 to 6
```

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.

```
NetIron(config-if-1)# show qos
```

...portions of table omitted for simplicity...

```
DSCP-Priority map: (dscp = d1d2)
```

d1 \ d2	0	1	2	3	4	5	6	7	8	9
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 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 forwarding priority.

Possible values: See above

Default value: See the "DSCP-Priority map" section in the **show qos** output

qos map ip-prec-dscp

Configures QoS by changing IP precedence to DSCP mappings.

NOTE: This command is supported only on the NetIron stackable Layer 3 Switch, and is available only in software release 07.1.16 or higher. The command is not supported in software release 07.2.x.

EXAMPLE:

To change the IP precedence to DSCP mappings, enter a command such as the following at the global CONFIG level of the CLI:

```
NetIron(config)# qos map ip-prec-dscp 0 32 24 48 16 8 56 40
```

These commands configure the mappings displayed in the IP precedence to DSCP portion of the QoS information display.

```
NetIron(config-if-1)# show qos

...portions of table omitted for simplicity...

IP Precedence-DSCP map:

  ip-prec:     0     1     2     3     4     5     6     7
  ------------------------------------------------------------
     dscp:     0    32    24    48    16     8    56    40
```

Syntax: [no] qos map ip-prec-dscp <dscp1> <dscp2> <dscp3> <dscp4> <dscp5> <dscp6> <dscp7> <dscp8>

The <dscp1> ... <dscp8> parameters specify the DSCP values you are mapping to the IP precedence values. You must enter DSCP values for all eight IP precedence values, in order from IP precedence value 0 – 7.

Possible values: See above

Default value: See the "IP Precedence-DSCP map" section in the **show qos** output

qos mechanism

Configures the queuing method used for QoS. Two queuing methods are available:

- Weighted (the default) – A weighted fair queuing algorithm is used to rotate service among the four queues. The rotation is based on the weights you assign to each queue. This is the default queuing method and uses a default set of queue weights. This method rotates service among the four queues, forwarding a specific number of packets in one queue before moving on to the next one.

  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.

- Strict – 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.

EXAMPLE:

To change the queuing method from weighted fair queuing to strict queuing:

```
BigIron(config)# qos mechanism strict
```

Syntax: [no] qos mechanism strict | weighted

Possible values: See above

Default value: weighted

qos name

Changes the QoS queue names from their defaults. The default queue names are qosp3, qosp2, qosp1, and qosp0.

EXAMPLE:

To rename queue qosp3 (the premium queue) to “92-octane”:

```
BigIron(config)# qos name qosp3 92-octane
BigIron(config)# write memory
```

Syntax: qos name <old-name> <new-name>

Possible values: 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.

qos profile

Changes the minimum guaranteed bandwidth percentages of the queues. If you change the percentages for the queues, the software changes the weights, which changes the number of visits a queue receives during a full queue cycle and also the number of packets sent from each queue during each visit. For example, if you change the percentages so that queue qosp3 receives a weight of 5, then the system processes five packets in that queue during each visit to the queue.

NOTE: The weighted fair queuing method is based on packet-level scheduling. As a result, a queue’s bandwidth percentage does not necessarily reflect the exact bandwidth share the queue receives. This is due to the effects of variable size packets.

EXAMPLE:

To change the minimum guaranteed bandwidth percentages of the queues:

```
BigIron(config)# qos profile qosp3 75 qosp2 10 qosp1 10 qosp0 5
Profile qosp3     : PREMIUM     bandwidth requested  75% calculated  75%
Profile qosp2     : HIGH        bandwidth requested  10% calculated  13%
Profile qosp1     : NORMAL      bandwidth requested  10% calculated   8%
Profile qosp0     : BEST-EFFORT bandwidth requested   5% calculated   4%
BigIron(config)# write memory
```

Notice that the CLI displays the percentages you request and the percentages the device can provide based on your request. The values are not always the same, as explained below.

Syntax: [no] qos profile <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.

NOTE: The percentages you enter must equal 100. Also, the percentage for the premium queue (the highest priority queue) must be at least 50.

If you enter percentages that are less than the minimum percentages supported for a queue, the CLI recalculates the percentages to fall within the supported minimums. Here is an example. In this example, the values entered for all but the best-effort queue (the lowest priority queue) are much lower than the minimum values supported for those queues.

Possible values: See above.

Default value: The following table lists the default minimum guaranteed bandwidth percentages of the queues:

Queue	Default Minimum Percentage of Bandwidth	
	Chassis devices and TurboIron/8	**FastIron 4802**
qosp3	80%	75%
qosp2	15%	15%
qosp1	3.3%	5%
qosp0	1.7%	5%

qos tagged-priority

Allows you to reassign 802.1p priorities to different QoS queues. Tagged priority applies to tagged packets that come in from tagged ports. These packets have a tag in the header that specifies the packet's VLAN ID and its 802.1p priority tag value, which is 3 bits long.

You can specify how the Foundry device interprets the 3-bit priority information by reassigning the priority levels to other queues. For example, if you want the device to disregard the 802.1p priority and instead assign the priority based on other items (VLAN, port, and so on), you can configure the device to set all the 802.1p priorities to the best-effort queue (qosp0). If a tagged packet's 802.1p priority level is always in the qosp0 queue, then the packet's outbound queue is affected by other items such as incoming port, VLAN, and so on.

EXAMPLE:

To reassign all 802.1p priority levels 2 – 7 to the best-effort queue (qosp0), enter the following commands:

```
BigIron(config)# qos tagged-priority 2 qosp0
BigIron(config)# qos tagged-priority 3 qosp0
BigIron(config)# qos tagged-priority 4 qosp0
BigIron(config)# qos tagged-priority 5 qosp0
BigIron(config)# qos tagged-priority 6 qosp0
BigIron(config)# qos tagged-priority 7 qosp0
BigIron(config)# write memory
```

Syntax: [no] qos tagged-priority <num> <queue>

The <num> parameter can be from 0 – 7 and specifies the IEEE 802.1 equivalent to one of the four QoS queues.

The <queue> parameter specifies the queue to which you are reassigning the priority level. You must specify one of the named queues. The default names are qosp3, qosp2, qosp1, and qosp0. The example above reassigns the 802.1p levels to queue qosp0. (There is no need to reassign levels 0 and 1 in this case, because they are already assigned to qosp0 by default.)

Possible values: See above.

Default value: By default, a Foundry device interprets the prioritization information in the 3-bit priority tag as follows:

Priority Level	Queue
6, 7	qosp3
4, 5	qosp2
2, 3	qosp1
0, 1	qosp0

qos-tos map cos-dscp

Changes the Class of Service (CoS) to DiffServ Control Point (DSCP) mappings used by the JetCore IP ToS-based QoS feature. The CoS –> DSCP mappings are used if the trust level is CoS and DSCP marking is enabled.

NOTE: This command is supported in release 07.6.01 and later and applies only to JetCore devices.

NOTE: ToS-based QoS must be enabled. See “port-priority” on page 6-92.

NOTE: To place a mapping change into effect, you must enter the **ip rebind-acl all** command at the global CONFIG level of the CLI after making the mapping change.

EXAMPLE:

```
BigIron(config)# qos-tos map cos-dscp 0 33 25 49 17 7 55 41
BigIron(config)# ip rebind-acl all
```

This command configures the mappings displayed in the COS-DSCP map portion of the QoS information display.

```
BigIron(config-if-1/1)# show qos-tos

...portions of table omitted for simplicity...

COS-DSCP map:

     COS: 0 1 2 3 4 5 6 7
  ---------------------------------------------------------
     dscp: 0 33 25 49 17 7 55 41
```

For information about the rest of this display, see the "Displaying Configuration Information" section in the "JetCore Type of Service (ToS) Based QoS" chapter of the *Foundry Enterprise Configuration and Management Guide*.

Syntax: [no] qos-tos cos-dscp <dscp0> <dscp1> <dscp2> <dscp3> <dscp4> <dscp5> <dscp6> <dscp7>

The <dscp1> ... <dscp8> parameters specify the DSCP values you are mapping to the eight CoS values. You must enter DSCP values for all eight CoS values, in order from CoS value 0 – 7.

Possible values: See above

Default value:

CoS (802.1p)	0	1	2	3	4	5	6	7
DSCP value	0	8	16	24	32	40	48	56

qos-tos map dscp-dscp

Changes a DSCP to DSCP mapping used by the JetCore IP ToS-based QoS feature.

NOTE: This command is supported in release 07.6.01 and later and applies only to JetCore devices.

NOTE: ToS-based QoS must be enabled. See “port-priority” on page 6-92.

NOTE: To place a mapping change into effect, you must enter the **ip rebind-acl all** command at the global CONFIG level of the CLI after making the mapping change.

EXAMPLE:

```
BigIron(config)# qos-tos map dscp-dscp 0 10
BigIron(config)# ip rebind-acl all
```

This command changes the mapping of DSCP value 0 from 0 to 10.

Syntax: [no] qos-tos map dscp-dscp <old-dscp-value> [<old-dscp-value>...]
to <new-dscp-value> [<new-dscp-value>...]

You can change up to eight DSCP values in the same commend. Make sure you enter the old values and their new values in the same order.

Possible values: See above

Default value: The device maps a packet's DSCP value to the same DSCP value. For example, if the packet has DSCP value 63 when the packet is received, the packet still has DSCP value 63 when the packet is placed in the hardware forwarding queue.

qos-tos map dscp-priority

Change a DSCP to internal forwarding priority mapping used by the JetCore IP ToS-based QoS feature.

NOTE: This command is supported in release 07.6.01 and later and applies only to JetCore devices.

NOTE: ToS-based QoS must be enabled. See "port-priority" on page 6-92.

NOTE: To place a mapping change into effect, you must enter the **ip rebind-acl all** command at the global CONFIG level of the CLI after making the mapping change.

EXAMPLE:

```
BigIron(config)# qos-tos map dscp-priority 0 2 3 4 to 1
BigIron(config)# qos-tos map dscp-priority 8 to 5
BigIron(config)# qos-tos map dscp-priority 16 to 4
BigIron(config)# qos-tos map dscp-priority 24 to 2
BigIron(config)# qos-tos map dscp-priority 32 to 0
BigIron(config)# qos-tos map dscp-priority 40 to 7
BigIron(config)# qos-tos map dscp-priority 48 to 3
BigIron(config)# qos-tos map dscp-priority 56 to 6
BigIron(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.

```
BigIron(config-if-1/1)# 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
```

For information about the rest of this display, see the "Displaying Configuration Information" section in the "JetCore Type of Service (ToS) Based QoS" chapter of the *Foundry Enterprise Configuration and Management Guide*.

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.

Possible values: See above

Default value:

DSCP value	0 – 7	8 – 15	16 – 23	24 – 31	32 – 41	40 – 47	48 – 55	56 – 63
Internal Forwarding Priority	0	1	2	3	4	5	6	7

qos-tos map ip-prec-dscp

Changes the IP precedence to DSCP mappings used by the JetCore IP ToS-based QoS feature. The IP precedence –> DSCP mappings are used if the trust level is IP Precedence and DSCP marking is enabled.

NOTE: This command is supported in release 07.6.01 and later and applies only to JetCore devices.

NOTE: ToS-based QoS must be enabled. See “port-priority” on page 6-92.

NOTE: To place a mapping change into effect, you must enter the **ip rebind-acl all** command at the global CONFIG level of the CLI after making the mapping change.

EXAMPLE:

```
BigIron(config)# qos-tos map ip-prec-dscp 0 32 24 48 16 8 56 40
BigIron(config)# ip rebind-acl all
```

This command configures the mappings displayed in the IP Precedence-DSCP map portion of the QoS information display.

```
BigIron(config-if-1/1)# show qos-tos
```

...portions of table omitted for simplicity...

```
IP Precedence-DSCP map:

  ip-prec:     0     1     2     3     4     5     6     7
  ---------------------------------------------------------
     dscp:     0    32    24    48    16     8    56    40
```

For information about the rest of this display, see the "Displaying Configuration Information" section in the "JetCore Type of Service (ToS) Based QoS" chapter of the *Foundry Enterprise Configuration and Management Guide*.

Syntax: [no] qos-tos map ip-prec-dscp <dscp1> <dscp2> <dscp3> <dscp4> <dscp5> <dscp6> <dscp7> <dscp8>

The <dscp1> ... <dscp8> parameters specify the DSCP values you are mapping to the IP precedence values. You must enter DSCP values for all eight IP precedence values, in order from IP precedence value 0 – 7.

Possible values: See above

Default value:

IP precedence	0	1	2	3	4	5	6	7
DSCP value	0	8	16	24	32	40	48	56

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

radius-server

Identifies a RADIUS server and sets other RADIUS authentication parameters for authenticating access to the Foundry device.

EXAMPLE:

```
BigIron(config)# radius-server host 209.157.22.99
```

Syntax: radius-server host <ip-addr> | <server-name> [auth-port <number>] [acct-port <number>]

<ip-addr> | <server-name> is either an IP address or an ASCII text string.

<auth-port> is the Authentication port number; it is an optional parameter. The default is 1645.

<acct-port> is the Accounting port number; it is an optional parameter. The default is 1646.

Syntax: radius-server [key 0 | 1 <key-string>] [timeout <number>] [retransmit <number>] [dead-time <number>]

The **key** <key-string> parameter specifies the value that the Foundry device sends to the server when trying to authenticate user access. The RADIUS server uses the key to determine whether the Foundry device has authority to request authentication from the server. The key can be from 1 – 32 characters in length and cannot include any space characters.

Syntax: radius-server host <ip-addr> | <server-name> [auth-port <number> acct-port <number> default key <string> dot1x]

The **default key** <string> **dot1x** parameter indicates that this RADIUS server supports the 802.1X standard. A RADIUS server that supports the 802.1X standard can also be used to authenticate non-802.1X authentication requests.

NOTE: To implement 802.1X port security, at least one of the RADIUS servers identified to the Foundry device must support the 802.1X standard.

When you display the configuration of the Foundry device, the RADIUS key is encrypted. For example:

```
BigIron(config)# radius-server key 1 abc
BigIron(config)# write terminal
...
```



```
radius-server host 1.2.3.5
radius key 1 $!2d
```

NOTE: Encryption of the RADIUS keys is done by default. The **0** parameter disables encryption. The **1** parameter is not required; it is provided for backwards compatibility.

The timeout <number> is how many seconds to wait before declaring a RADIUS server timeout for the authentication request. The default timeout is 3 seconds. The range of possible timeout values is from 1 – 15.

The retransmit <number> is the maximum number of retransmission attempts. When an authentication request timeout, the Foundry software will retransmit the request up to the maximum number of retransmissions configured. The default retransmit value is 3 seconds. The possible retransmit value is from 1 – 5.

The **dead-time** parameter is not used in this software release. When the software allows multiple authentication servers, this parameter will specify how long the Foundry device waits for the primary authentication server to reply before deciding the server is dead and trying to authenticate using the next server. The dead-time value can be from 1 – 5 seconds. The default is 3.

You can designate a server to handle a specific AAA task. For example, you can designate one RADIUS server to handle authorization and another RADIUS server to handle accounting. You can specify individual servers for authentication and accounting, but not for authorization. You can set the RADIUS key for each server.

Syntax: radius-server host <ip-addr> | <server-name> [authentication-only | accounting-only | default] [key 0 | 1 <string>]

The **default** parameter causes the server to be used for all AAA functions.

Possible values: see above

Default value: see above

rarp

Enters a static IP RARP entry for static routes on a Foundry Layer 3 Switch.

EXAMPLE:

```
BigIron(config)# rarp 1 1245.7654.2348 192.53.4.2
BigIron(config)# exit
BigIron# write memory
```

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.

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.

Possible values: See above

Default value: N/A

rate-limit-arp

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.

EXAMPLE:

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:

```
BigIron(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.

Possible values: See above

Default value: No limit

redundancy

Changes the CLI to the configuration level for redundant management modules.

relative-utilization

Allows you to 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.

EXAMPLE:

To configure a link utilization list with port 1/1 as the uplink port and ports 1/2 and 1/3 as the downlink ports.

```
BigIron(config)# relative-utilization 1 uplink eth 1/1 downlink eth 1/2 to 1/3
```

Syntax: [no] relative-utilization <num> uplink ethernet <portnum> [to <portnum> | <portnum>…] downlink ethernet <portnum> [to <portnum> | <portnum>…]

Possible values: 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).

Default value: N/A

rmon alarm

Defines what MIB objects are monitored, the type of thresholds that will be monitored (falling, rising or both), the value of those thresholds, and the sample type (absolute or delta).

An **alarm** event will be reported each time that a threshold is exceeded. The alarm entry also defines the action (event) to take should the threshold be exceeded.

A sample CLI alarm entry and its syntax is shown below:

EXAMPLE:

```
BigIron(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-number> <sampling-time> <sample-type> <threshold-type> <threshold-value> <event-number> <threshold-type> <threshold-value> <event-number> owner <text>

Possible values:

- Threshold type: rising-threshold or falling threshold
- Sample type: delta or absolute

Default value: N/A

rmon event

There are two elements to the RMON event group 9, 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 displayed by entering the CLI command **show event**.

The event log table collects and stores reported events for retrieval by an RMON application.

EXAMPLE:

```
BigIron(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>

Possible values: `N/A`

Default value: N/A

rmon history

All active Foundry Layer 2 Switch and Layer 3 Switch ports by default will generate two RMON history (group 2) control data entries. If a port becomes inactive, then the two entries will automatically be deleted.

Two history entries are generated for each device by default:

- a sampling of statistics every 30 seconds
- a sampling of statistics every 30 minutes

You can modify how many of these historical entries are saved in an event log (buckets) as well as how often these intervals are taken. The station (owner) that collects these entries can also be defined.

To review the control data entry for each port or interface, enter the **show rmon history** command.

EXAMPLE:

```
BigIron(config)# rmon history 1 interface 1 buckets 10 interval 10 owner nyc02
```

Syntax: rmon history <entry-number> interface <portnum> buckets <number> interval <sampling-interval> owner <text-string>

Possible values: Buckets: 1 – 50 entries.

Default value: N/A

route-map

Creates a route map and places you in the Route Map CONFIG level of the CLI. 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. See the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

EXAMPLE:

To add instance 1 of a route map named "GET_ONE" with a permit action, enter the following command.

```
BigIron(config)# route-map GET_ONE permit 1
BigIron(config-routemap GET_ONE)#
```

Syntax: 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 “Route Map Commands” on page 19-1. Also see the “Configuring BGP4” chapter of the *Foundry Enterprise Configuration and Management Guide*.

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. Routes are compared to the instances in ascending numerical order. For example, a route is compared to instance 1, then instance 2, and so on.

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.

```
BigIron(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:

```
BigIron(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.

Possible values: N/A

Default value: N/A

route-only

Globally disables Layer 2 switching on a Foundry Layer 3 Switch.

NOTE: 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.

NOTE: As an alternative to disabling switching globally, you can disable it on individual interfaces. See “route-only” on page 7-72.

EXAMPLE:

```
BigIron(config)# route-only
BigIron(config)# exit
```



```
BigIron# write memory
BigIron# reload
```

Syntax: [no] route-only

Possible values: N/A

Default value: Enabled

router appletalk

This is a launch command that allows you to move to the AppleTalk configuration level.

EXAMPLE:

```
BigIron(config)# router appletalk
BigIron(config-atalk-router)# end
BigIron# write memory
BigIron# reload
```

NOTE: You must reset the system when AppleTalk is first enabled on the router using the **router appletalk** command. If you have previously reset the system and defined AppleTalk interface(s), and the interface configuration represents an addition, then no reset of the system is required.

Syntax: router appletalk

Possible values: N/A

Default value: disabled

router bgp

This is a launch command that allows you to move to the BGP configuration level.

NOTE: If you disable BGP4 by entering the **no router bgp** command, all BGP4 configuration information is deleted. To disable BGP4 without losing the configuration information, use the **no local-as** command to disable the local AS instead. See “local-as” on page 13-9.

EXAMPLE:

```
BigIron(config)# router bgp
BigIron(config-bgp-router)#
```

Syntax: [no] router bgp

Possible values: N/A

Default value: disabled

router dvmrp

This is a launch command that allows you to move to the DVMRP configuration level.

NOTE: You must reload the software after enabling this protocol to place the change into effect.

EXAMPLE:

```
BigIron(config)# router dvmrp
BigIron(config-dvmrp-router)# write memory
BigIron(config-dvmrp-router)# end
BigIron# reload
```

Syntax: router dvmrp

Possible values: N/A

Default value: disabled

router fsrp

This is a launch command that enables the FSRP feature. FSRP allows redundant paths to be assigned. Parameters for FSRP are set using the Interface level command **ip fsrp address** <ip-addr>…

EXAMPLE:

To enable FSRP on the Layer 3 Switch, enter the following:

```
BigIron(config)# router fsrp
BigIron(config-fsrp-router)# write memory
BigIron(config-fsrp-router)# end
BigIron# reload
```

Possible values: N/A

Default value: disabled

router ipx

Activates IPX routing on a Layer 3 Switch.

NOTE: You must reload the software after enabling this protocol to place the change into effect.

EXAMPLE:

```
BigIron(config)# router ipx
BigIron(config-ipx-router)# write memory
BigIron(config-ipx-router)# end
BigIron# reload
```

Syntax: router ipx

Possible values: N/A

Default value: disabled

router isis

Activates IS-IS routing on a Foundry Layer 3 Switch and launches you into the IS-IS configuration level.

NOTE: You must enable the protocol globally and also on individual interfaces. Globally enabling the protocol does not enable it on individual interfaces. To enable IS-IS on an interface, see “ip router isis” on page 7-44.

EXAMPLE:

```
NetIron(config)# router isis
ISIS: Please configure NET!
```

The command in this example globally enables IS-IS and changes the CLI to the IS-IS configuration level. If you have not already configured a NET for the IS-IS, the message shown in this example is displayed. To configure a NET, use the **net** command. See “net <area-id>.<system-id>.<sel>” on page 12-8.

Syntax: router isis

Possible values: N/A

Default value: disabled

router mpls

Activates MPLS on a Foundry NetIron and launches you into the MPLS configuration level.

NOTE: You must enable MPLS globally and also on individual interfaces. Globally enabling MPLS does not enable it on individual interfaces. To enable MPLS on an interface, see “mpls-interface” on page 22-2.

EXAMPLE:

```
BigIron(config)# router mpls
BigIron(config-mpls)
```

Syntax: router mpls

Possible values: N/A

Default value: disabled

router msdp

Activates Multicast Source Discovery Protocol (MSDP) on a Layer 3 Switch and places the CLI at the MSDP configuration level.

NOTE: You must reload the software after enabling this protocol to place the change into effect.

EXAMPLE:

```
BigIron(config)# router msdp
BigIron(config-msdp-router)# write memory
BigIron(config-msdp-router)# end
BigIron# reload
```

Syntax: router msdp

Possible values: N/A

Default value: disabled

router ospf

Activates OSPF routing on a Foundry Layer 3 Switch and launches you into the OSPF configuration level.

EXAMPLE:

```
BigIron(config)# router ospf
BigIron(config-ospf-router)#
```

Syntax: router ospf

Possible values: N/A

Default value: disabled

router pim

Activates PIM multicast on a Layer 3 Switch.

NOTE: You must reload the software after enabling this protocol to place the change into effect.

EXAMPLE:

```
BigIron(config)# router pim
BigIron(config-pim-router)# write memory
BigIron(config-pim-router)# end
BigIron# reload
```

Syntax: router pim

Possible values: N/A

Default value: disabled

router rip

Activates RIP routing on a Layer 3 Switch and launches you into that configuration level to assign or modify RIP parameters.

NOTE: You must enable the protocol globally and also on individual interfaces. Globally enabling the protocol does not enable it on individual interfaces. To enable RIP on an interface, see "ip rip" on page 7-42.

EXAMPLE:

```
BigIron(config)# router rip
BigIron(config-rip-router)# write memory
BigIron(config-rip-router)# end
BigIron# reload
```

Syntax: router rip

Possible values: N/A

Default value: disabled

router vrrp

Enables VRRP.

EXAMPLE:

```
BigIron(config)# router vrrp
```

Syntax: router vrrp

Possible values: N/A

Default value: disabled

router vrrp-extended

Enables VRRP Extended (VRRPE).

EXAMPLE:

```
BigIron(config)# router vrrp-extended
```

Syntax: router vrrp-extended

Possible values: N/A

Default value: disabled

router vsrp

Disables or re-enables the Virtual Switch Redundancy Protocol (VSRP) on a Layer 3 Switch. VSRP is an alternative to STP that provides Layer 2 and Layer 3 redundancy and sub-second failover in mesh topologies.

EXAMPLE:

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 disable VSRP:

```
BigIron(config)# no router vsrp
```

To re-enable VSRP:

```
BigIron(config)# router vsrp
```

Syntax: [no] router vsrp

NOTE: 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.

Possible values: N/A

Default value: VSRP is enabled by default on Layer 2 Switches and Layer 3 Switches. You can disable VSRP on Layer 3 Switches but not on Layer 2 Switches.

scale-timer

Changes 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.

To achieve sub-second failover times, you can shorten the duration of all VSRP timers by adjusting the timer scale. 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.

EXAMPLE:

To change the timer scale, enter a command such as the following at the global CONFIG level of the CLI:

```
BigIron(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.

Possible values: 1 – 10

Default value: 1

server port

Adds a profile for an application TCP or UDP port. This command applies only when you are using a Layer 3 Switch for the Globally-distributed Server Load Balancing (SLB) feature. See the "Route Health Injection" chapter of the *Foundry Enterprise Configuration and Management Guide*. When you add a profile for an application port, the health check for the port is automatically enabled.

EXAMPLE:

To add a profile for TCP port 80 and thus enable its health check, enter the following commands:

```
BigIron(config)# server port 80

BigIron(config-port-80)#
```

Syntax: server port <num>

See for “Application Port Commands” on page 42-1 for information about the commands you can enter at the Application Port level.

Possible values: TCP port number

Default value: N/A

server real-name

Identifies a Web server for Globally-distributed Server Load Balancing (SLB). Globally-distributed SLB allows the same web site (and same IP address) to reside on multiple servers, which usually are in geographically dispersed locations. See the "Route Health Injection" chapter of the *Foundry Enterprise Configuration and Management Guide*.

Use the server **real-name** command to identify the web sites for which the Foundry Layer 3 Switch is helping to provide geographically-distributed SLB.

EXAMPLE:

```
BigIron(config)# server real S2 209.157.22.249

BigIron(config-rs-S2)# port http keepalive
```

Syntax: [no] server real-name <name> <vip>

The <name> parameter identifies the ServerIron, third-party SLB, or real server. This value does not need to match a value on the ServerIron, third-party SLB, or real server. The value simply identifies the ServerIron, third-party SLB, or real server uniquely on the Layer 3 Switch.

The <vip> parameter is the IP address of the web site. If the web server is directly attached to the Layer 3 Switch, this is the IP address of the IP address on the web server. If the web server is attached to a ServerIron or third-party SLB, the VIP is the virtual IP address configured on the ServerIron or third-party SLB for the web site.

Possible values: see above

Default value: N/A

server session-vm-limit

Changes the maximum number of Layer 4 sessions the individual Velocity Switching Processors (VSPs) on the Velocity Management Module can have.

EXAMPLE:

```
BigIron(config)# server session-vm-limit 1000000
```

Syntax: [no] server session-vm-limit <num>

The <num> parameter indicates the maximum number of Layer 4 sessions (main cache flows) the VM can have. You can specify from 32768 – 5000000 (five million). The default is 2000000 (two million).

The command applies to all the VSPs.

You must reload the software to place the change into effect. Enter the following commands to reload the software.

```
BigIron(config)# exit
BigIron# reload
```

Possible values: 32768 – 5000000 (five million)

Default value: 2000000 (two million)

service password-encryption

Enables password encryption. When encryption is enabled, users cannot learn the device's passwords by viewing the configuration file. Password encryption is enabled by default.

NOTE: Password encryption does not encrypt the password in Telnet packets sent to the device. This feature applies only to the configuration file.

EXAMPLE:

```
BigIron(config)# no service password-encryption
```

Syntax: [no] service password-encryption

Possible values: N/A

Default value: Enabled

set pcmcia

Changes parameters for a 3Com Megahertz 10/100 LAN PC Card (model 3CCFE574BT) inserted in a PCMCIA flash card slot. You can use the PC Card as a management interface to the Foundry device.

NOTE: This feature supports 3Com Megahertz 10/100 LAN PC Card model 3CCFE574BT only.

NOTE: This command applies only to the Management IV module.

For PC Card information, see www.pc-card.com.

You can change the following parameter settings.

- Port speed (10 Mbps, 100 Mpbs, or auto-sensing) – The default is 10 Mbps.
- Mode (half-duplex or full-duplex) – The default is half-duplex.
- 802.3x Flow control (applies only to full-duplex mode) – Disabled by default.
- IP address of the management port (applies only to Layer 3 code) – None configured by default.

EXAMPLE:

To change the port speed, enter a command such as the following at the global CONFIG level of the CLI:

```
BigIron(config)# set pcmcia slot1 auto
```

Syntax: [no] set pcmcia slot1 | slot2 10 | 100 | auto

The **slot1** | **slot2** parameter specifies the PCMCIA slot. There is no default.

The **10** | **100** | **auto** parameter specifies the port speed. The default is 10 Mbps.

EXAMPLE:

To change the mode, enter a command such as the following at the global CONFIG level of the CLI:

```
BigIron(config)# set pcmcia slot1 full
```

Syntax: [no] set pcmcia slot1 | slot2 full | half

The **slot1** | **slot2** parameter specifies the PCMCIA slot. There is no default.

The **full** | **half** parameter specifies the mode. The default is **half** (half-duplex).

EXAMPLE:

To enable 802.3x flow control, enter the following command at the global CONFIG level of the CLI:

```
BigIron(config)# set pcmcia slot2 flow-control
```

Syntax: [no] set pcmcia slot1 | slot2 flow-control

EXAMPLE:

To configure an IP address on the management port, enter a command such as the following at the global CONFIG level of the CLI:

```
BigIron(config)# set pcmcia slot1 ip 10.10.10.2/24
```

Syntax: [no] set pcmcia slot1 | slot2 ip <ip-addr> <ip-mask>

or

Syntax: [no] set pcmcia slot1 | slot2 ip <ip-addr>/<mask-bits>

Possible values: See above

Default value: See above

sflow destination

Specifies the collector for sFlow data.

EXAMPLE:

```
BigIron(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.

NOTE: The default UDP port is present in release 07.5.06 and later only. Earlier releases do not have a default.

Possible values: See above

Default value: UDP port 6343. There is no default for the IP address.

sflow enable

Globally enables sFlow.

NOTE: You also must enable sFlow forwarding on individual interfaces. See "sflow forwarding" on page 7-73.

EXAMPLE:

```
BigIron(config)# sflow enable
```

Syntax: [no] sflow enable

Possible values: N/A

Default value: Disabled

sflow polling-interval

Changes the sFlow polling interval.

The polling interval defines how often the MIB objects for network interface statistics are polled. The interval value applies to all interfaces on which sFlow is enabled.

EXAMPLE:

To change the interval, enter a command such as the following at the global CONFIG level of the CLI:

```
BigIron(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.

Possible values: 1 to any higher value

Default value: 20 seconds

sflow sample

Globally changes the average packet sampling rate for sFlow.

NOTE: You also can specify the sampling rate for an individual interface. See “sflow sample” on page 7-73.

EXAMPLE:

```
BigIron(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. 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.

- 2
- 8
- 32
- 128
- 512
- 2048
- 8192
- 32768
- 131072
- 524288
- 2097152
- 8388608
- 33554432
- 134217728
- 536870912
- 2147483648

The default is 512 packets.

Possible values: See above

Default value: In releases prior to 07.6.03, the default sampling rate is 512 packets. In software releases 07.6.03 and later, the default sampling rate depends on the device being configured. The following are the new default sFlow sampling rates:

- Stackables: 128
- 4-slot chassis: 2048
- 8-slot chassis: 8192
- 15-slot chassis: 8192

sflow source

Specifies the source interface for exporting data to the sFlow collector.

NOTE: This command does not apply to release 07.6.00 or later.

By default, the Foundry device uses the port that is connected to a collector as the source interface for flows exported to that collector. You can specify an Ethernet port or POS port, a loopback interface, or the null interface as the source for sFlow export packets.

- Ethernet or POS port – sFlow sends the export packets out the specified interface. To load balance the export of flows from the main cache to a given collector, configure multiple source interfaces for the collector.
- Loopback interface – sFlow sends the export packets from the specified loopback address, using a physical port connected to the collector to transmit the packet.
- Null interface – sFlow continues to collect flows but does not export them to the collector. Use this type of interface when you want to administratively stop flow export without stopping flow collection and without removing configuration information.

EXAMPLE:

```
BigIron(config)# sflow source ethernet 1/1
```

This command configures port 1/1 to be the source interface for sFlow packets.

Syntax: [no] sflow source ethernet | pos | loopback <portnum>

Syntax: [no] sflow source null

The **ethernet | pos | loopback** <portnum> specifies a physical port or loopback interface.

The **null** parameter discards the export packets instead of sending them to a collector. However, the sFlow agent continues to collect samples.

Possible values: See above

Default value: The interface connected to the collector

show

Displays a variety of configuration and statistical information about the Layer 2 Switch or Layer 3 Switch. See “Show Commands” on page 43-1.

snmp disable

Disables SNMP management on the Foundry device.

EXAMPLE:

To disable SNMP management of the device:

```
BigIron(config)# snmp disable
```

To later re-enable SNMP management of the device:

```
BigIron(config)# no snmp disable
```

Syntax: [no] snmp disable

Possible values: N/A

Default value: N/A

snmp-client

Restricts SNMP management access to the Foundry device to the host whose IP address you specify. No other device except the one with the specified IP address can access the Foundry device through IronView or any other SNMP application.

If you want to restrict access from Telnet or the Web, use one or both of the following commands:

- **telnet-client** – restricts Telnet access. See “telnet-client” on page 6-135.
- **web-client** – restricts Web access. See “web-client” on page 6-143.

If you want to restrict all management access, you can use the commands above and the **snmp-client** command or you can use the following command: **all-client**. See “all-client” on page 6-11.

EXAMPLE:

To restrict SNMP access (which includes IronView) to the Foundry device to the host with IP address 209.157.22.26, enter the following command:

```
BigIron(config)# snmp-client 209.157.22.26
```

Syntax: [no] snmp-client <ip-addr>

Possible values: a valid IP address. You can enter one IP address with the command. You can use the command up to ten times for up to ten IP addresses.

Default value: N/A

snmp-server community

Assigns an SNMP community string for the system:

- read-only (public)
- read-write (private)

EXAMPLE:

```
BigIron(config)# snmp-server community planet1 ro view admin 2
```

Syntax: snmp-server community [0 | 1] <string> ro | rw [view <viewname>] [<standard-acl-name> | <standard-acl-id>]

The <string> parameter specifies the community string name.

The **ro** | **rw** parameter specifies whether the string has read-only (ro) or read-write (rw) privileges to the assigned view.

The **0** | **1** parameter affects encryption for display of the string in the running-config and the startup-config file. Encryption is enabled by default.

When encryption is enabled, the community string is encrypted in the CLI regardless of the access level you are using. In the Web management interface, the community string is encrypted at the read-only access level but is 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 community string you specify with the command. The community 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 community string to be encrypted.
- **1** – Assumes that the community 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 community string. In this case, the software decrypts the community string you enter before using the value for authentication. If you accidentally enter option **1** followed by the clear-text version of the community string, authentication will fail because the value used by the software will not match the value you intended to use.

The **view** <viewstring> parameter is optional. It allows you to associate a view to the members of this community string. If no view is specified, access to the full MIB is granted.

The <standard-acl-name> | <standard-acl-id> parameter is optional. It allows you to specify which ACL group will be used to filter incoming SNMP packets. You can enter either the ACL name or its ID.

Possible values: See above

Default value: The default read-only (**ro**) community string is "public". Foundry devices do not have a default read-write (**rw**) community string.

snmp-server contact

Identifies a system contact. You can designate a contact name for the Layer 2 Switch or Layer 3 Switch and save it in the configuration file for later reference. You can later access contact information using the **show snmp server** command.

EXAMPLE:

```
BigIron(config)# snmp-server contact Noi Lampa
```

Syntax: snmp-server contact <text>

Possible values: up to 32 alphanumeric characters for the system contact text string

Default value: N/A

snmp-server enable ethernet

Allows SNMP access only to clients on a specific port.

EXAMPLE:

The following example configures the device to allow SNMP access only to clients connected to Ethernet port 7/11.

```
BigIron(config)# snmp-server enable ethernet 7/11
```

Syntax: [no] snmp-server enable ethernet <portnum>

Possible values: An Ethernet port

Default value: N/A

snmp-server enable traps

When the command is preceded with **no**, the command is used to stop certain traps from being generated by a system. The following SNMP traps are collected by default:

- authentication key
- cold-start
- link-up
- link-down
- new-root
- topology-change
- power-supply-failure
- locked-address-violation

EXAMPLE:

To stop reporting incidences of links that are down, enter the following commands:

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

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

Possible values: trap type (for example, cold-start, new-root, and so on)

Default value: All of the following SNMP traps are enabled and will be generated by default for a system:

- authentication key
- cold-start
- link-up

- link-down
- new-root
- topology-change
- power-supply-failure
- locked-address-violation

snmp-server enable traps holddown-time

Changes the holddown time for SNMP traps.

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 link up trap remembers which ports are up during the holddown them and the device sends the traps, including traps such as “cold start” or “warm start” that occur before the holddown time expires.

EXAMPLE:

```
BigIron(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.

Possible values: 1 – 600 seconds

Default value: 60 seconds

snmp-server enable vlan

Allows SNMP access only to clients in a specific VLAN.

EXAMPLE:

The following example configures the device to allow SNMP access only to clients connected to ports within port-based VLAN 40. Clients connected to ports that are not in VLAN 40 are denied access.

```
BigIron(config)# snmp-server enable vlan 40
```

Syntax: [no] snmp-server enable vlan <vlan-id>

Possible values: A VLAN ID

Default value: N/A

snmp-server engineid

Changes the default engine ID to a user-defined one. (For SNMP version 3.) An SNMP engine ID identifies an SNMP management entity.

EXAMPLE:

```
BigIron(config)# snmp-server engineid local 800007c70300e05290ab60
```

Syntax: [no] snmp-server engineid local <hex-string>

The **local** parameter indicates that engine ID to be entered is the ID of this device.

NOTE: Since the current implementation of SNMP version 3 does not support Notification, remote engine IDs cannot be configured at this time.

The <hex-string> variable consists of 11 octets, entered as hexadecimal values. There are two hexadecimal characters in each octet. There should be an even number of hexadecimal characters in an engine ID.

The default engine ID has a maximum of 11 octets:

- Octets 1 through 4 represent the agent's SNMP management private enterprise number as assigned by the Internet Assigned Numbers Authority (IANA). The most significant bit of Octet 1 is "1". For example, “000007c7” is the ID for Foundry Networks in hexadecimal. With Octet 1 always equal to "1", the first four octets in the default engine ID is always “800007c7” (which is 1991 in decimal).
- Octet 5 is always 03 in hexadecimal and indicates that the next set of values represent a MAC address.
- Octets 6 through 11 form the MAC address of the lowest port in the management module.

NOTE: Engine ID must be a unique number among the various SNMP engines in the management domain. Using the default engine ID ensures the uniqueness of the numbers.

Possible values: See above.

Default value: Default engine ID.

snmp-server group

Maps SNMP users to SNMP views. For each SNMP group, you can configure a read view, a write view, or both. Users who are mapped to a group will use its views for access control. (For SNMP version 3.)

EXAMPLE:

```
BigIron(config)# snmp-server group admin v3 auth read v1default write v1default
```

Syntax: [no] snmp-server group <groupname>
v1 | v2 | v3
auth | noauth
[access <standard-acl-id>] [read <viewstring> | write <viewstring>]

NOTE: This command is not used for SNMP version 1 and SNMP version 2. In these versions, groups and group views are created internally using community strings. When a community string is created, two groups are created, based on the community string name. One group is for SNMP version 1 packets, while the other is for SNMP version 2 packets.

The **group** <groupname> parameter defines the name of the SNMP group to be created.

The **v1**, **v2**, or **v3** parameter indicates which version of SNMP is used. In most cases, you will be using v3.

The **auth** | **noauth** parameter determines whether or not authentication will be required to access the supported views. If auth is selected, then only authenticated packets are allowed to access the view specified for the user group. Selecting **noauth** means that no authentication is not required to access the specified view.

The **access** <standard-acl-id> parameter is optional. It allows incoming SNMP packets to be filtered based on the standard ACL attached to the group.

The **read** <viewstring> | **write** <viewstring> parameter is optional. It indicates that users who belong to this group have either read or write access to the portion of the MIB specified by the <viewstring>.

The <viewstring> variable is the name of the view to which the SNMP group members have access. If no view is specified, then the group has no access to the MIB.

The value of <viewstring> is defined using the **snmp-server view** command. The SNMP agent comes with the "v1default" view; however, it must be specified when defining a group. The "v1default" view provides access to the entire MIB. The "v1default" view also allows SNMP version 3 to be backwards compatibility with SNMP version 1 and version 2.

NOTE: If you will be using a view other than the "v1default" view, that view must be exist before creating the user group. See the section “snmp-server view” on page 6-123.

To delete a group, use the no parameter before the command.

Possible values: See above.

Default value: N/A

snmp-server host

Assigns or removes a station as an SNMP trap receiver. To assign the trap receiver, use the command **snmp-server host**. To later remove the trap receiver feature, enter **no snmp-server host**.

EXAMPLE:

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:

```
BigIron(config)# # snmp-server host 2.2.2.2 0 mypublic port 200
BigIron(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:

```
BigIron(config)# snmp-server host 2.2.2.2 0 BigIron-12
BigIron(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 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.

Possible values: IP address of trap receiver station, community string

Default value: no system default

snmp-server location

Identifies a system location for the Layer 2 Switch or Layer 3 Switch. This information is saved in the configuration file for later reference. You can later access system location information using the **show snmp server** command.

EXAMPLE:

```
BigIron(config)# snmp-server location pulchritude_lane
```

Syntax: snmp-server location <text>

Possible values: up to 32 alphanumeric characters for the snmp-server location text string

Default value: N/A

snmp-server pw-check

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. You can disable this password checking with the **no snmp-server pw-check** command.

EXAMPLE:

```
BigIron(config)# no snmp-server pw-check
```

Syntax: [no] snmp-server pw-check

Possible values: N/A

Default value: N/A

snmp-server trap-source

Specifies 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.

EXAMPLE:

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

```
BigIron(config)# int loopback 1
BigIron(config-lbif-1)# ip address 10.0.0.1/24
BigIron(config-lbif-1)# exit
BigIron(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 Layer 3 Switch. Regardless of the port the Foundry device uses to send traps to the receiver, the traps always arrive from the same source IP address.

Syntax: snmp-server trap-source loopback <num> | ethernet <portnum> | pos <portnum> | ve <num>

The <num> parameter is a loopback interface or virtual interface number. If you specify an Ethernet or POS port, the <portnum> is the port's number (including the slot number, if you are configuring a Chassis device). The lowest-numbered address on the specified interface is used as the trap source.

Possible values: Valid Ethernet port, POS port, loopback interface, or virtual interface

Default value: N/A

snmp-server user

Creates an SNMP user, defines the group to which the user will be associated, defines the type of authentication to be used for SNMP access by the user.

EXAMPLE:

```
BigIron(config)# snmp-s user bob admin v3 access 2 encrypted auth md5 md5authstring
```

Syntax: [no] snmp-server user <name> <groupname> v3
[[access <standard-acl-id>] [encrypted] [auth md5 <md5-password> | sha <sha-password>]
[priv [encrypted] des <des-password>]]]

The <name> parameter defines the SNMP user name or security name used to access the management module.

The <groupname> parameter identifies the SNMP group to which this user is associated or mapped. All users must be mapped to an SNMP group. Groups are defined using the **snmp-server group** command.

NOTE: The SNMP group to which the user account will be mapped should be configured before creating the user accounts; otherwise, the group will be created without any views. Also, ACL groups must be configured before configuring user accounts.

The **v3** parameter is required.

The **access** <standard-acl-id> parameter is optional. It indicates that incoming SNMP packets are filtered based on the ACL attached to the user account.

NOTE: The ACL specified in a user account overrides the ACL assigned to the group to which the user is mapped. If no ACL is entered for the user account, then the ACL configured for the group will be used to filter packets.

The **encrypted** parameter means that the MD5 or SHA password will be a digest value. MD5 has 16 octets in the digest. SHA has 20. The digest string has to be entered as a hexadecimal string. In this case, the agent need not generate any explicit digest. If the **encrypted** parameter is not used, the user is expected to enter the authentication password string for MD5 or SHA. The agent will convert the password string to a digest, as described in RFC 2574.

The **auth md5 | sha** parameter is optional. It defines the type of encryption that the user must have to be authenticated. Choose between MD5 or SHA encryption. MD5 and SHA are two authentication protocols used in SNMP version 3.

The <md5-password> and <sha-password> define the password the user must use to be authenticated. These password must have a minimum of 8 characters. If the encrypted parameter is used, then the digest has 16 octets for MD5 or 20 octets for SHA.

NOTE: Once a password string is entered, the generated configuration displays the digest (for security reasons), not the actual password.

The **priv [encrypted] des** <des-password> parameter is optional. It defines the type of encryption that will be used to encrypt the privacy password. If the "encryption" keyword is used, enter a 16-octet DES key in hexadecimal format for the des-password. If the "encryption" keyword is not used enter a password string. The agent will generate a suitable 16-octet DES key from the password string.

Currently, DES is the only encryption type supported for priv password.

To delete a user account, use the no parameter before the command.

Possible values: See above.

Default value: N/A

snmp-server view

Configures an SNMP view. You can use an SNMP view as an argument with other commands.

SNMP views are named groups of MIB objects that can be associated with user accounts to allow limited access for viewing and modification of SNMP statistics and system configuration. SNMP views can also be used with other commands that take SNMP views as an argument. SNMP views reference MIB objects using object names, numbers, wildcards, or a combination of the three. The numbers represent the hierarchical location of the object in the MIB tree. You can reference individual objects in the MIB tree or a subset of objects from the MIB tree.

NOTE: The **snmp-server view** command supports the MIB objects as defined in RFC 1445.

EXAMPLE:

To add an SNMP view, use the following CLI method:

```
BigIron(config)# snmp-server view Maynes system included
BigIron(config)# snmp-server view Maynes system.2 excluded
BigIron(config)# snmp-server view Maynes 2.3.*.6
BigIron(config)# write mem
```

Syntax: [no] snmp-server view <name> <mib_tree> included | excluded

The <name> parameter can be any alphanumeric name you choose to identify the view. The names cannot contain spaces.

The <mib_tree> parameter is the name of the MIB object or family. MIB objects and MIB sub-trees can be identified by name or by the numbers representing the position of the object or sub-tree in the MIB hierarchy. You can use a wildcard (*) in the numbers to specify a sub-tree family.

The **included** | **excluded** parameter specifies whether the MIB objects identified by the <mib_family> parameter are included in the view or excluded from the view.

NOTE: All MIB objects are automatically excluded from any view unless they are explicitly included; therefore, when creating views using the **snmp-server view** command, indicate which portion of the MIB you want users to access.

For example, you may want to allow a community string called “admin” to view the Foundry MIBs area, which starts with the 1.3.6.1.4.1.1991 object identifier. Enter the following command:

```
BigIron(config)# snmp-server view admin 1.3.6.1.4.1.1991 included
```

Within an inclusion scope, you can exclude portions of the MIB. For example, if you want to exclude the snAgentSys objects, which begin with 1.3.6.1.4.1.1991.1.1.2 object identifier from the admin view, enter a second command such as the following:

```
BigIron(config)# snmp-server view admin 1.3.6.1.4.1.1991.1.1.2 excluded
```

Note that the exclusion is within the scope of the inclusion.

To delete a view, use the no parameter before the command.

Possible values: See above

Default value: N/A

sntp poll-interval

This parameter sets how often clock updates are requested from an SNTP server.

EXAMPLE:

To configure the Layer 2 Switch or Layer 3 Switch to poll for clock updates from an SNTP server every 15 minutes, enter the following:

```
BigIron(config)# sntp poll-interval 900
```

Syntax: sntp poll-interval <1 – 65535>

Possible values: 1 – 65535 seconds

Default value: 1800 seconds

sntp server

Allows you to define the SNTP server that will be used for clock synchronization for the Foundry device. You can enter the SNTP server’s IP address or its host name.

Up to three SNTP server entries can be defined.

EXAMPLE:

To define the SNTP server (IP address 192.1.4.69) that will be polled by the Layer 2 Switch or Layer 3 Switch for time updates, enter:

```
BigIron(config)# sntp server 192.1.4.69
```

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.

Possible values: See above.

Default value: N/A

spanning-tree

Enables or disables (no) Spanning Tree on the device. This change can be viewed by the **show spanning tree** command.

- For Layer 2 Switches, this feature is enabled by default.
- For Layer 3 Switches, this feature is disabled by default.

To disable this feature, enter **no spanning-tree**. To later re-enable spanning tree on the router, enter **spanning-tree**.

Also, once 802.1W has been enabled on a device (see the "spanning-tree 802-1w" on page 6-126 command), use this command to disable and re-enable 802.1W on a port.

EXAMPLE:

To disable spanning tree, enter the following:

```
BigIron(config)# no span
```

EXAMPLE:

To enable spanning tree, enter the following:

```
BigIron(config)# spanning-tree
```

EXAMPLE:

To disable 802.1W on a port, enter the following:

```
BigIron(config)# interface 1/1
BigIron(config-if-1/1)# no spanning-tree
```

Syntax: [no] spanning-tree

Possible values: N/A

Default value: Enabled on Layer 2 Switches. Disabled on Layer 3 Switches.

spanning-tree <parameter>

Spanning Tree bridge and port parameters are configurable using one CLI command. When no port-based VLANs are active on the system, spanning tree parameters are set at the Global CONFIG Level.

When port-based VLANs are active on the system, spanning tree protocol bridge and port parameters can be configured at the VLAN Level (see "spanning-tree" on page 33-11). Additionally, you can disable or enable STP on an interface basis.

NOTE: If VLANs are active on a Layer 2 Switch or Layer 3 Switch, spanning-tree will not be seen as an option at the Global CONFIG Level of the CLI but will be an option of the VLAN Level.

All bridge and port parameters have default values and do not need to be modified unless required to match network needs. Additionally, all values will be globally applied to the Layer 2 Switch or Layer 3 Switch. By default this feature is enabled on Layer 2 Switches and disabled on Layer 3 Switches.

You can modify the following STP Parameters:

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

EXAMPLE:

To enable spanning tree on a system in which no port-based VLANs are active and change the hello-time from the default value of 2 to 8 seconds, enter the following commands.

```
BigIron(config)# span hello-time 8
BigIron(config)# span ethernet 1/5 path-cost 15 priority 64
```

Here is the syntax for global STP parameters.

Syntax: spanning-tree [forward-delay <value>] | [hello-time <value>] | [maximum-age <value>] | [priority <value>]

Here is the syntax for port STP parameters.

Syntax: spanning-tree ethernet | pos <portnum> path-cost <value> | priority <value>

Possible values: see below

Bridge Parameters:

- **forward-delay**: **Possible values:** 4 – 30 seconds. Default is 15 seconds.
- **max-age**: **Possible values:** 6 – 40 seconds. Default is 20 seconds.
- **hello-time**: **Possible values:** 1 – 10 seconds. Default is 2 seconds.
- **priority**: **Possible values:** 1 – 65535. Default is 32768. A higher numerical value means a lower priority; thus, the highest priority is 0.

Port Parameters:

- **path**: **Possible values:** 1 – 65535. Default: The default depends on the port type:
  - 10 Mbps – 100
  - 100 Mbps – 19
  - Gigabit – 4
  - 10 Gigabit – 2
  - OC-3c – 200
  - OC-12c – 80
  - OC-48c – 20
- **priority**: possible values are 8 – 252, in increments of 4. Default is 128. A higher numerical value means a lower priority; thus, the highest priority is 8.

NOTE: The range in software releases earlier than 07.5.01 is 0 – 255. 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.

spanning-tree 802-1w

Enables 802.1W on all ports of a device running the Spanning Tree Protocol. The 802.1W feature is Foundry's implementation of the IEEE 802.1W standard Rapid Spanning Tree Protocol (RSTP). Foundry's earlier implementation of RSTP, which was 802.1W Draft 3, provided only a subset of the IEEE 802.1W standard; whereas, this implementation provides the full standard. This new implementation of the protocol is referred to as 802.1W. The implementation of the 802.1W Draft 3 is referred to as RSTP Draft 3. (See the command "spanning-tree single rstp" on page 6-129 to enable RSTP Draft 3.)

EXAMPLE:

To enable 802.1W on a device, enter commands such as the following:

```
BigIron(config)# spanning-tree 802-1w
```

Syntax: [no] spanning-tree 802-1w

Possible Values: N/A

Default values: Disabled

spanning-tree 802-1w <parameter>

Configures 802.1W bridge and port parameters once 802.1W has been enabled on the device. There are two versions of this command. The first one configures bridge parameters; the other configures port parameters.

Bridge Parameters:

Configuration changes using these parameters are applied to all ports on the bridge.

EXAMPLE:

To change 802.1W bridge parameters, enter a commands such as the following:

```
BigIron(config)# spanning-tree 802-1w forward-delay 20 hello-time 3 max-age 10
priority 10
```

The command in this example changes the parameter 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. (See "spanning-tree 802-1w" on page 33-12 for an example.)

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 from non-root bridge.

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.

Beginning with software release 07.6.03, 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.

Possible values: See above

Default values: See above

Port Parameters:

These commands can be entered on individual ports or on multiple ports, such as all ports that belong to a VLAN.

EXAMPLE:

You can change the following 802.1W port parameters using the following methods.

```
BigIron(config)# vlan 10
BigIron(config-vlan-10)# spanning-tree 802-1w ethernet 1/5 path-cost 15 priority 64
```

Syntax: spanning-tree 802-1w ethernet <portnum> path-cost <value> | priority <value> | [admin-edge-port] | [admin-pt2pt-mac] | [force-migration-check]

The **ethernet | pos** <portnum> parameter specifies the interface used.

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 6.2 shows the recommended path cost values from the IEEE standards.

Table 6.2: 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
OC-3c	128,000	12,800 – 1,280,000
OC-12c	32,000	3,200 – 320,000
OC-48c	8,000	800 – 80,000
OC-192c	2,000	200 – 20,000

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 – 255. The default is 128. A higher numerical value means a lower priority. 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.

Possible Values: See above

Default values: See above

spanning-tree single 802-1w

Enables 802.1W on all ports of a single spanning tree.

EXAMPLE:

Enter a command such as the following:

```
BigIron(config)# spanning-tree single 802-1w
```

Syntax: [no] spanning-tree single 802-1w

Possible values: N/A

Default value: Disabled

spanning-tree single <parameter>

Configures single spanning tree. Single spanning tree enables you to configure a single instance of the Spanning Tree Protocol (SSTP) to run on all the port-based VLANs on a device.

SSTP uses the same parameters, with the same value ranges and defaults, as the default STP on Foundry devices (multiple-instance STP or "MSTP"), which is described in the previous section.

When you enable SSTP, all VLANs in which STP is enabled are added to the single spanning tree. VLANs in which STP is disabled are excluded from the single spanning tree.

spanning-tree single rstp

Enables 802.1W Draft 3 Rapid Spanning Tree on a device that is running Single Spanning Tree.

NOTE: To enable 802.1W Draft 3 on a device that is not running Single Spanning Tree, enter the **spanning-tree rstp** command at the VLAN configuration level. See “spanning-tree rstp” on page 33-15.

802.1W Draft 3 enhances STP by providing a fast failover mechanism for a root port that fails on a non-root bridge. 802.1W Draft 3 provides a subset of the capabilities described in the 802.1W STP specification.

EXAMPLE:

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:

```
BigIron(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 RSTP.

To disable 802.1W Draft 3 on a device that is running single STP, enter the following command:

```
BigIron(config)# no spanning-tree single rstp
```

Possible values: N/A

Default value: Disabled

ssh access-group

Specifies an ACL that restricts SSH access to management functions on the device.

EXAMPLE:

To configure an ACL that restricts SSH access the device:

```
BigIron(config)# access-list 12 deny host 209.157.22.98 log
BigIron(config)# access-list 12 deny 209.157.23.0 0.0.0.255 log
BigIron(config)# access-list 12 deny 209.157.24.0/24 log
BigIron(config)# access-list 12 permit any
BigIron(config)# ssh access-group 12
BigIron(config)# write memory
```

Syntax: ssh access-group <num>

The <num> parameter specifies the number of a standard ACL and must be from 1 – 99.

These commands configure ACL 12, then apply the ACL as the access list for SSH access. The device denies SSH access from the IP addresses listed in ACL 12 and permits SSH access from all other IP addresses. Without the last ACL entry for permitting all packets, this ACL would deny SSH access from all IP addresses.

Possible values: see above

Default value: N/A

static-mac-address

Defines a static MAC address on an individual Layer 2 Switch or switching port to ensure it is not aged out.

NOTE: Foundry recommends that you configure a static ARP entry to match the static MAC entry. In fact, the software automatically creates a static MAC entry when you create a static ARP entry. See "arp" on page 6-13.

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.

EXAMPLE:

```
BigIron(config)# static 1145.5563.67FF e12 7 router-type
```

Here is the syntax for Chassis devices, the FastIron 4802, and the TurboIron/8.

Syntax: [no] static-mac-address <mac-addr> ethernet <portnum> [to <portnum> ethernet <portnum>] [normal-priority | high-priority] [host-type | router-type | fixed-host]

Here is the syntax for Stackable devices.

Syntax: static-mac-address <mac-addr> ethernet <portnum> [normal-priority | high-priority] [host-type | router-type]

Possible values: The priority can be 0 – 7 (0 is lowest and 7 is highest) for Chassis devices, the FastIron 4802, or the TurboIron/8, and either normal-priority or high-priority for Stackable devices.

Default value: host-type; 0 or normal priority

stp-group

Begins configuration of an STP group. An STP group enables you to manage multiple port-based VLANs in the same spanning tree, without using the Single Span feature. When you enter this command, the CLI changes to the STP group configuration level. (See "STP Group Commands" on page 37-1.)

EXAMPLE:

```
BigIron(config)# stp-group 1
BigIron(config-stp-group-1)# master-vlan 2
BigIron(config-stp-group-1)# member-vlan 3 to 4
BigIron(config-stp-group-1)# exit
BigIron(config)# stp-group 2
BigIron(config-stp-group-2)# master-vlan 12
BigIron(config-stp-group-2)# member-vlan 13 to 14
```

These commands configure two STP groups and add VLANs to those groups. All the VLANs in an STP group are managed in the same spanning tree. For information about the commands at the STP group configuration level, see "STP Group Commands" on page 37-1.

Syntax: [no] stp-group <num>

The <num> parameter specifies the STP group ID and can be from 1 – 32.

Possible values: 1 – 32

Default value: N/A

super-span-global

Globally enables the SuperSpan™ feature.

Use this command after you configure the SuperSpan boundary interfaces. (See "stp-boundary" on page 7-75.) You can enable SuperSpan globally or on an individual VLAN level. If you enable the feature globally, the feature is enabled on all VLANs. To enable or disable SuperSpan in an individual VLAN, see "super-span" on page 33-16.

NOTE: If you enable the feature globally, then create a new VLAN, the new VLAN inherits the global SuperSpan state. For example, if SuperSpan is globally enabled when you create a VLAN, SuperSpan also is enabled in the new VLAN.

For information about this feature, see the "SuperSpan™" section in the "Configuring Spanning Tree Protocol (STP) and IronSpan" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

EXAMPLE:

```
BigIron(config)# super-span-global
```

Syntax: [no] super-span-global [preforward-delay <secs>]

The <secs> parameter specifies the length of the Preforwarding state. You can specify from 3 – 30 seconds. The default is 5 seconds.

Possible values: 3 – 30 seconds for the Preforwarding state

Default value: Disabled; when SuperSpan is enabled, the default length for the Preforwarding state is 5 seconds.

system-max

Allows you to modify the default settings for parameters that use system memory. The configurable parameters and their defaults and maximums differ depending on the device. To display the configurable parameters, their defaults, and the maximum configurable values for each, enter the following command at any level of the CLI: **show default values**. See "show default" on page 43-21.

NOTE: You must save the configuration (**write memory**), then reload the software to place this command into effect.

NOTE: You do not need to reload the software for the **pim-max-int-group** or **dvmrp-max-int-group** option.

NOTE: Beginning with software release 07.6.02, there is no longer a limit to the number of interface groups that can be configured for PIM and DVMRP; therefore, the **system-max pim-max-int-group** and the **system-max dvmrp-max-int-group** commands have been removed from the CLI.

EXAMPLE:

To define the maximum number of repeated DVMRP traffic being sent from the same source address and being received by the same destination address, enter a command such as the following:

```
BigIron(config)# system-max dvmrp-mcache 500
```

Syntax: system-max dvmrp-mcache <num>

Possible values: The <num> parameter specifies the maximum number of multicast cache entries for DVMRP. Enter a number from 128 – 2048.

Default value: The default is 512.

EXAMPLE:

To increase the system capacity of a BigIron for IP routes from the default 10000 to 50000, enter the following command:


```
BigIron(config)# system-max ip-route 50000
```

Syntax: system-max <parameter> <value>

Possible values: These depend on the device you are configuring. See the System Parameters section in the show default values display. The CLI will display the acceptable range if you enter a value that is outside the range.

Default value: These depend on the device you are configuring. See the System Parameters section in the show default values display. The CLI will display the acceptable range if you enter a value that is outside the range.

EXAMPLE:

To define the maximum number of entries for the Multicast Flow table, enter a command such as the following:

```
BigIron(config)# system-max multicast-flow 2048
```

Syntax: system-max multicast-flow <num>

Possible values: 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.

Default value: The default is 1024.

NOTE: Do not set this maximum too high since you may run out of resources in the CAM.

EXAMPLE:

To define the maximum number of repeated PIM traffic being sent from the same source address and being received by the same destination address, enter a command such as the following:

```
BigIron(config)# system-max pim-mcache 999
```

Syntax: system-max pim-mcache <num>

Possible values: The <num> parameter specifies the maximum number of multicast cache entries for PIM. Enter a number from 256 – 4096.

Default: The default is 1024.

EXAMPLE:

To increase the number of SNMP views available on a Foundry device:

```
BigIron(config)# system-max view 15
```

Syntax: system-max view <number-of-views>

This command specifies the maximum number of SNMPv2 and v3 views that can be configured on a device.

Possible values: The number of views can be from 10 – 65536.

Default value: The default is 10 views.

A view can be configured using command “snmp-server view” on page 6-123.

tacacs-server

Identifies a TACACS or TACACS+ server and sets other TACACS/TACACS+ parameters for authenticating access to the Foundry device.

EXAMPLE:

```
BigIron(config)# tacacs-server host 209.157.22.99
```

Syntax: tacacs-server host <ip-addr> | <server-name> [auth-port <number>]

The only required parameter is the IP address or host name of the server. You can enter this command up to three times, to add up to three servers. During authentication, the device tries to reach the servers in the order you add them.

NOTE: To specify the server's host name instead of its IP address, you must first identify a DNS server using the **ip dns server-address** <ip-addr> command at the global CONFIG level. See the "Configuring IP" chapter of the *Foundry Enterprise Configuration and Management Guide*.

The **auth-port** parameter specifies the UDP port number of the authentication port on the server. The default port number is 49.

Syntax: tacacs-server [key 0 | 1 <string>] [timeout <number>] [retransmit <number>] [dead-time <number>]

The **key** parameter specifies the value that the Foundry device sends to the server when trying to authenticate user access. The TACACS+ server uses the key to determine whether the Foundry device has authority to request authentication from the server. The key can be from 1 – 32 characters in length and cannot include any space characters.

NOTE: Encryption of the TACACS+ keys is done by default. The **0** parameter disables encryption. The **1** parameter is not required; it is provided for backwards compatibility.

The **timeout** parameter specifies how many seconds the Foundry device waits for a response from the TACACS/TACACS+ server before either retrying the authentication request or determining that the TACACS/TACACS+ server is unavailable and moving on to the next authentication method in the authentication-method list. The timeout can be from 1 – 15 seconds. The default is 3 seconds.

The **retransmit** parameter specifies how many times the Foundry device will resend an authentication request when the TACACS/TACACS+ server does not respond. The retransmit value can be from 1 – 5 times. The default is 3 times.

The dead-time parameter specifies how long the Foundry device waits for the primary authentication server to reply before deciding the server is dead and trying to authenticate using the next server. The dead-time value can be from 1 – 5 seconds. The default is 3.

In a TACACS+ configuration, you can designate a server to handle a specific AAA task. For example, you can designate one TACACS+ server to handle authorization and another TACACS+ server to handle accounting. You can set the TACACS+ key for each server.

For example, to specify different TACACS+ servers for authentication, authorization, and accounting:

```
BigIron(config)# tacacs-server host 1.2.3.4 auth-port 49 authentication-only key abc
BigIron(config)# tacacs-server host 1.2.3.5 auth-port 49 authorization-only key def
BigIron(config)# tacacs-server host 1.2.3.6 auth-port 49 accounting-only key ghi
```

Syntax: tacacs-server host <ip-addr> | <server-name> [authentication-only | authorization-only | accounting-only | default] [key <string>]

The **default** parameter causes the server to be used for all AAA functions.

After authentication takes place, the server that performed the authentication is used for authorization and/or accounting. If the authenticating server cannot perform the requested function, then the next server in the configured list of servers is tried; this process repeats until a server that can perform the requested function is found, or every server in the configured list has been tried.

Possible values: see above

Default value: see above

tag-type

This parameter defines the value that will be sent out on a packet to indicate it is part of a tagged VLAN port. The 802.1p/q standard recognizes the value of 8100 for this purpose. Other values can be assigned to this parameter but are not recommended.

EXAMPLE:

```
BigIron(config)# tag-type 8100
```

Syntax: tag-type <hex-value>

Possible values: A hexadecimal value from 0 – ffff.

Default value: 8100

telnet access-group

Specifies an ACL that restricts Telnet access to management functions on the device.

EXAMPLE:

To configure an ACL that restricts Telnet access the device:

```
BigIron(config)# access-list 12 deny host 209.157.22.98 log
BigIron(config)# access-list 12 deny 209.157.23.0 0.0.0.255 log
BigIron(config)# access-list 12 deny 209.157.24.0/24 log
BigIron(config)# access-list 12 permit any
BigIron(config)# telnet access-group 12
BigIron(config)# write memory
```

Syntax: telnet access-group <num>

The <num> parameter specifies the number of a standard ACL and must be from 1 – 99.

These commands configure ACL 12, then apply the ACL as the access list for Telnet access. The device denies Telnet access from the IP addresses listed in ACL 12 and permits Telnet access from all other IP addresses. Without the last ACL entry for permitting all packets, this ACL would deny Telnet access from all IP addresses.

Possible values: see above

Default value: N/A

telnet login-timeout

Changes the login timeout period for Telnet sessions.

EXAMPLE:

To change the login timeout period for Telnet sessions to 5 minutes:

```
BigIron(config)# telnet login-timeout 5
```

Syntax: [no] telnet login-timeout <minutes>

Possible values: 1 – 10 minutes

Default value: 1 minute

telnet server enable vlan

Allows Telnet access only to clients in a specific VLAN.

EXAMPLE:

The following command configures the device to allow Telnet management access only to clients connected to ports within port-based VLAN 10. Clients connected to ports that are not in VLAN 10 are denied management access.

```
BigIron(config)# telnet server enable vlan 10
```

Syntax: [no] telnet server enable vlan <vlan-id>

Possible values: N/A

Default value: N/A

telnet server suppress-reject-message

Suppresses the rejection message the device sends in response to a denied Telnet client.

If you enable suppression of the connection rejection message, a denied Telnet client does not receive a message from the Foundry device. Instead, the denied client simply does not gain access.

EXAMPLE:

To suppress the connection rejection message sent by the device to a denied Telnet client, enter the following command at the global CONFIG level of the CLI:

```
BigIron(config)# telnet server suppress-reject-message
```

Syntax: [no] telnet server suppress-reject-message

Possible values: N/A

Default value: Disabled

telnet-client

Restricts Telnet management access to the Foundry device to the host whose IP address you specify. No other device except the one with the specified IP address can access the Foundry device's CLI through Telnet.

If you want to restrict access from SNMP or the Web, use one or two of the following commands:

- **snmp-client** – restricts SNMP access (including IronView). See "snmp-client" on page 6-116.
- **web-client** – restricts web access. See "web-client" on page 6-143.

If you want to restrict all management access, you can use the commands above and the **telnet-client** command or you can use the following command: **all-client**. See "all-client" on page 6-11.

EXAMPLE:

To restrict Telnet access (which includes IronView) to the Foundry device to the host with IP address 209.157.22.26, enter the following command:

```
BigIron(config)# telnet-client 209.157.22.26
```

Syntax: [no] telnet-client <ip-addr>

Possible values: a valid IP address. You can enter one IP address with the command. You can use the command up to ten times for up to ten IP addresses.

Default value: N/A

telnet-server

Enables or disables Telnet access to a Foundry device. By default, Telnet access is allowed on a system.

EXAMPLE:

To disable Telnet access to a Foundry device, enter the following:

```
BigIron(config)# no telnet-server
```

Syntax: [no] telnet-server

Possible values: Enabled or disabled

Default value: Enabled

telnet-timeout

Defines how many minutes a Telnet session can remain idle before it is timed out. An idle Telnet session is a session that is still sending TCP ACKs in response to keepalive messages from the Foundry device, but is not being used to send data.

By default, the Telnet timeout is zero (which means Telnet sessions do not time out).

NOTE: Foundry devices also have another, non-configurable Telnet timer used to close sessions that have ended abnormally. This mechanism is enabled regardless of the setting of the Telnet timeout. The Foundry device sends TCP keepalive messages to the Telnet client once a minute. If the client fails to respond to two consecutive keepalive messages, the Foundry device concludes that the TCP session has ended abnormally and immediately ends the session. A typical cause of a session ending abnormally is the client rebooting during the TCP session.

EXAMPLE:

```
BigIron(config)# telnet-timeout 120
```

Syntax: telnet-timeout <0 – 240>

Possible values: 0 – 240 minutes

Default value: 0 minutes (no timeout)

terminal length

Specifies how many lines to display on the screen during the current CLI session. This command is useful when reading multiple lines of displayed information, especially those that do not fit on one screen.

EXAMPLE:

To specify the maximum number of lines displayed on one page, enter a command such as the following:

```
BigIron(config)# terminal length 15
```

Syntax: terminal length <number-of-lines>

Possible values: The <number-of-lines> parameter indicates the maximum number of lines that will display on a full screen of text during the current session. If the displayed information requires more than one page, the terminal pauses. Pressing the space bar displays the next page.

Default: The default <number-of-lines> is 24. Entering a value of 0 prevents the terminal from pausing between multiple output pages:

tftp client enable vlan

Allows TFTP access only to clients in a specific VLAN.

EXAMPLE:

The following example configures the device to allow TFTP access only to clients connected to ports within port-based VLAN 40. Clients connected to ports that are not in VLAN 40 are denied access.

```
BigIron(config)# tftp client enable vlan 40
```

Syntax: [no] tftp client enable vlan <vlan-id>

Possible values: N/A

Default value: N/A

topology-group

Creates a topology group.

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

EXAMPLE:

```
BigIron(config)# topology-group 2
BigIron(config-topo-group-2)# master-vlan 2
BigIron(config-topo-group-2)# member-vlan 3
BigIron(config-topo-group-2)# member-vlan 4
BigIron(config-topo-group-2)# member-vlan 5
BigIron(config-topo-group-2)# member-group 2
```


Syntax: [no] topology-group <group-id>

The <group-id> parameter specifies the topology group ID and can be from 1 – 256.

For information about the other commands in this example, see “Topology Group Commands” on page 36-1.

Possible values: See above

Default value: N/A

trunk

Allows you to add a trunk group and connect the ports in the group to a switch, router, or server for high-speed connections.

See the "Configuring Trunk Groups and Dynamic Link Aggregation" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide* for trunk configuration rules and more examples.

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.

NOTE: Trunking is supported on POS OC-3 and OC-12 ports but not on OC-48 ports. Server trunking of POS ports is supported only for Layer 2 and requires software release 07.6.01 or later.

NOTE: You can trunk two 10 Gigabit Ethernet ports together. The first port must be in an odd-numbered chassis slot and the second port must be in the following even-numbered slot.

NOTE: If you are running a software release earlier than 07.5.00, you must save the configuration (**write memory**), then reload the software to place this command into effect. On devices running 07.5.00 or later, you can dynamically place trunk configuration changes into effect by entering the **trunk deploy** command at the global CONFIG level of the CLI.

EXAMPLE:

To assign ports 1 and 2 to a trunk group, enter the following command:

```
BigIron(config)# trunk switch e 1/1 to 1/2
```

A trunk group must then also be configured on the connecting Layer 2 Switch or Layer 3 Switch at the other end of the trunk group. The **switch** parameter in the above command can refer to another Foundry Layer 2 Switch or Layer 3 Switch.

If you are going to connect to a server, then enter the following command:

```
BigIron(config)# trunk server e 1/1 to 1/2
```

This will connect a trunk group of ports 1 and 2 to a server.

Syntax: [no] trunk [server | switch] ethernet | pos <primary-portnum> to <portnum>
ethernet | pos <primary-portnum> to <portnum>

The **server** | **switch** parameter specifies whether the trunk ports will be connected to a server or to another Layer 2 Switch or Layer 3 Switch. This parameter affects the type of load balancing performed by the device. See the “Configuring Trunk Groups and Dynamic Link Aggregation” chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*. The default is **switch**.

Each **ethernet** or **pos** parameter introduces a port group.

The <primary-portnum> **to** <portnum> parameters specify the ports. The first port must be a primary port and the remaining ports must be the ports that follow it. The primary port is always the lowest number in the port range.

EXAMPLE:

To configure a trunk group consisting of two groups of ports, 1/1 – 1/4 on module 1 and 4/5 – 5/8 on module 4, enter the following commands:

```
BigIron(config)# trunk ethernet 1/1 to 1/4 ethernet 4/5 to 4/8
BigIron(config)# write memory
BigIron(config)# trunk deploy
```

Syntax: trunk [server | switch] ethernet <primary-portnum> to <portnum> ethernet <primary-portnum> to <portnum>

The **server** | **switch** parameter specifies whether the trunk ports will be connected to a server or to another Layer 2 Switch or Layer 3 Switch. This parameter affects the type of load balancing performed by the Foundry device. See the "Configuring Trunk Groups and Dynamic Link Aggregation" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*. The default is **switch**.

Each **ethernet** or **pos** parameter introduces a port group.

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. For Gigabit Ethernet modules, the primary ports are 1, 3, 5, and 7.

Possible values: see above

Default value: N/A

trunk deploy

Dynamically places trunk configuration changes into effect.

NOTE: You still need to save the trunk configuration changes to the startup-config file in order for the changes to be retained following a software reload.

EXAMPLE:

```
BigIron(config)# trunk ethernet 1/1 to 1/8
BigIron(config-trunk-1/1-1/8)# write memory
BigIron(config-trunk-1/1-1/8)# exit
BigIron(config)# trunk deploy
```

Syntax: trunk deploy

Possible values: N/A

Default value: N/A

unknown-unicast limit

Specifies the maximum number of unknown-unicast packets the device can forward each second. By default the device sends unknown unicasts and all other traffic at wire speed and is limited only by the capacities of the hardware. However, if other devices in the network cannot handle unlimited unknown-unicast traffic, this command allows you to relieve those devices by throttling the unknown unicasts at the Foundry device.

NOTE: The unknown-unicast limit does not affect broadcast or multicast traffic. However, you can use the **broadcast limit** and **multicast limit** commands to control these types of traffic. See "broadcast limit" on page 6-18 and "multicast limit" on page 6-90.

EXAMPLE:

```
BigIron(config)# unknown-unicast limit 30000
```

Syntax: unknown-unicast limit <num>

Possible values: 0 – 4294967295; if you specify 0, limiting is disabled.

Default value: N/A

username

Configures a local user account. For each user account, you specify the user name. You also can specify the following parameters:

- A password
- The privilege level, which can be one of the following:
- Full access (super-user). This is the default.
- Port-configuration access
- Read-only access

EXAMPLE:

To configure a user account, enter a command such as the following at the global CONFIG level of the CLI.

```
BigIron(config)# username wonka password willy
```

This command adds a user account for a super-user with the user name "wonka" and the password "willy", with privilege level super-user. This user has full access to all configuration and display features.

NOTE: If you configure user accounts, you must add a user account for super-user access before you can add accounts for other access levels. You will need the super-user account to make further administrative changes.

```
BigIron(config)# username waldo privilege 5 password whereis
```

This command adds a user account for user name "waldo", password "whereis", with privilege level read-only. Waldo can look for information but cannot make configuration changes.

Syntax: [no] username <user-string> privilege <privilege-level> password | nopassword <password-string>

The **privilege** parameter specifies the privilege-level. You can specify one of the following:

- **0** – Full access (super-user)
- **4** – Port-configuration access
- **5** – Read-only access

The default privilege level is 0. If you want to assign full access to the user account, you can enter the command without "**privilege 0**", as shown in the command example above.

The **password** | **nopassword** parameter indicates whether the user must enter a password. If you specify **password**, enter the string for the user's password.

NOTE: You must be logged on with super-user access (privilege level 0, or with a valid Enable password for super-user access) to add user accounts or configure other access parameters.

virtual-interface-mac

Specifies the MAC address for virtual routing interfaces.

By default, a Foundry device uses the MAC address of the first port (1 or 1/1) as the MAC address for all virtual routing interfaces configured on the device. You can specify a different MAC address for the virtual routing interfaces. If you specify another MAC address for the virtual routing interfaces, the address applies to all the virtual routing interfaces configured on the device.

EXAMPLE:

```
BigIron(config)# virtual-interface-mac aaaa.bbbb.cccc
BigIron(config)# write memory
BigIron(config)# end
BigIron# reload
```

Syntax: [no] virtual-interface-mac <mac-addr>

Enter the MAC address in the following format: HHHH.HHHH.HHHH

NOTE: You must save the configuration and reload the software to place the change into effect.

Possible values: A MAC address belonging to one of the device's ports

Default value: The MAC address of the first port on the device

vlan

Creates or changes the CLI focus to a port-based VLAN.

EXAMPLE:

```
BigIron(config)# vlan 200 by port
BigIron(config)# vlan 200 name Prod Marketing
```

Syntax: vlan <num> by port

Syntax: vlan <num> name <string>

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.

Possible values: VLAN ID 1 – 4096; VLAN name can be a string up to 16 characters. You can use blank spaces in the name if you enclose the name in double quotes (for example, "Prod Marketing".)

Default value: n/a

vlan-dynamic-discovery

Disables or re-enables dynamic discovery of protocol VLANs on switch-to-switch links. This feature enables switch-to-switch links to be automatically included in protocol VLANs that have dynamic port membership.

EXAMPLE:

To disable the feature, enter the following command:

```
BigIron(config)# no vlan-dynamic-discovery
```

Syntax: [no] vlan-dynamic-discovery

Possible values: Enabled or disabled

Default value: Enabled

vlan-group

Configures a VLAN group. A VLAN group enables you to easily configure multiple VLANs that have identical parameters.

You can add a virtual interface group to each VLAN group. See "interface group-ve" on page 6-34.

EXAMPLE:

To configure a VLAN group, enter commands such as the following:

```
BigIron(config)# vlan-group 1 vlan 2 to 1000
BigIron(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 | pos <portnum> [to <portnum> | ethernet <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 interfaces, before you configure the VLAN groups. This is true regardless of whether you use the virtual interface groups. The memory allocation is required because the VLAN groups and virtual interface groups have a one-to-one mapping.

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:

```
BigIron(config-vlan-group-1)# add-vlan 1001 to 1002
BigIron(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>]

Possible values: See above

Default value: n/a

vlan max-vlans

Allows you to assign a set number of VLANs to be supported on a Layer 2 Switch or Layer 3 Switch. This allows you to set a smaller value than the default to preserve memory on the system.

EXAMPLE:

```
BigIron(config)# vlan max-vlans 200
```

Syntax: vlan max-vlans <value>

Possible values: 1 – 1,024

Default value: 32

vm boot

Changes the default boot source for a Velocity Management Module.

EXAMPLE:

```
BigIron(config)# vm boot secondary
BigIron(config)# write memory
```

This command configures the module to boot from the secondary flash by default.

To configure the module to pause during booting to allow you to specify the boot source, enter the following command:

```
BigIron(config)# vm boot interactive
```

NOTE: The **write memory** command saves the change to the startup-config file. You must save the configuration change for the change to remain in effect after you reboot.

Syntax: vm boot primary | secondary | interactive

The **primary** and **secondary** parameters specify a flash memory location. The **interactive** parameter causes the device to pause during bootup to allow you to specify the boot source for the VSPs. You must use this method if you want to boot the VSPs from a TFTP server. Otherwise, the **interactive** parameter is used for troubleshooting.

Possible values: See above

Default value: primary

vm vm-map

Maps a forwarding module to a specific Velocity Switching Processor (VSP) on the Velocity Management Module.

EXAMPLE:

```
BigIron(config)# vm vm-map slot 3 vm-slot 2 vm-cpu 1
```

This command remaps processing for the modules in slot 3 to VSP CPU 1 on the VM1 in slot 2.

Syntax: vm vm-map <from-slotnum> vm-slot <to-slotnum> vm-cpu <cpunum>

The <from-slotnum> parameter specifies the slot that contains the forwarding module.

The <to-slotnum> parameter specifies the slot that contains the VM1.

The <cpunum> parameter specifies the VSP CPU on <to-slotnum> that will perform the processing. The VSM CPUs are numbered from 1 – 3.

Possible values: See above

Default value: Allocations are based on module weight and occur during software reload.

vm vm-map per-port-dma

Configures the VM1 to use per-DMA VSP load sharing. This command allows the VM1 to assign DMAs to VSPs dynamically. When the device is started or reset, the VM1 load balances processing by assigning DMAs to the VSPs according to the total bandwidth of the DMAs. See "VSP Load Sharing on a Per-DMA Basis" in the *Foundry Switch and Router Installation and Basic Configuration Guide* for more information.

EXAMPLE:

```
BigIron(config)# vm vm-map per-port-dma
```

Syntax: [no] vm vm-map per-port-dma

Possible values: N/A

Default value: If the **vm vm-map per-port-dma** command is in the Foundry device's configuration when the device is started or reset, the VM1 uses per-DMA VSP load sharing. Otherwise, the VM1 uses per-module VSP load sharing to balance forwarding among the VSPs.

If any ports or modules are statically assigned to VSPs, then those assignments are made prior to any dynamic assignments. You can have both per-module static assignments and per-DMA static assignments in a configuration.

vm vm-map port-dma

Assigns ports to individual VSPs statically. Forwarding for all of the ports controlled by the specified port's DMA are handled by the specified VSP. See "VSP Load Sharing on a Per-DMA Basis" in the *Foundry Switch and Router Installation and Basic Configuration Guide* for more information.

EXAMPLE:

```
BigIron(config)# vm vm-map port-dma 2/1 vm-slot 1 vm-cpu 1
```

This command assigns the DMA that controls port 2/1 to VSP 1 on the VM1 in slot 1:

Syntax: [no] vm vm-map port-dma <port> vm-slot <slot> vm-cpu <vsp-cpu>

Possible values:

The <port> parameter is one of the ports controlled by the DMA that you want to assign to a VSP. To assign a DMA to a VSP, you specify any of the ports controlled by the DMA as the <port> parameter. Forwarding for all of the ports controlled by the DMA is then handled by the specified VSP.

The <slot> parameter specifies the slot that contains the VM1.

The <vsp-cpu> parameter specifies the VSP CPU on <slot> that will perform the processing. The VSM CPUs are numbered from 1 – 3.

Default value: Allocations are based on module weight and occur during software reload.

web access-group

Specifies an ACL that restricts Web management access to management functions on the device.

EXAMPLE:

To configure an ACL that restricts Web management access the device:

```
BigIron(config)# access-list 12 deny host 209.157.22.98 log
BigIron(config)# access-list 12 deny 209.157.23.0 0.0.0.255 log
BigIron(config)# access-list 12 deny 209.157.24.0/24 log
BigIron(config)# access-list 12 permit any
BigIron(config)# web access-group 12
BigIron(config)# write memory
```

Syntax: web access-group <num>

The <num> parameter specifies the number of a standard ACL and must be from 1 – 99.

These commands configure ACL 12, then apply the ACL as the access list for Web management access. The device denies Web management access from the IP addresses listed in ACL 12 and permits Web management access from all other IP addresses. Without the last ACL entry for permitting all packets, this ACL would deny Web management access from all IP addresses.

Possible values: see above

Default value: N/A

web-client

Restricts Web management access to the Foundry device to the host whose IP address you specify. No other device except the one with the specified IP address can access the Foundry device's Web management interface.

If you want to restrict access from SNMP or Telnet, use one or two of the following commands:

- **snmp-client** – restricts SNMP access (including IronView). See "snmp-client" on page 6-116.
- **telnet-client** – restricts Telnet access to the CLI. See "telnet-client" on page 6-135.

If you want to restrict all management access, you can use the commands above and the **web-client** command or you can use the following command: **all-client**. See "all-client" on page 6-11.

EXAMPLE:

To restrict Web access to the Foundry device to the host with IP address 209.157.22.26, enter the following command:

```
BigIron(config)# web-client 209.157.22.26
```

Syntax: [no] web-client <ip-addr>

Possible values: a valid IP address. You can enter one IP address with the command. You can use the command up to ten times for up to ten IP addresses.

Default value: N/A

web-management

Sets configuration options on the Web management interface. By default the Web management interface is enabled.

EXAMPLE:

To disable the Web management interface on a Foundry device, enter the following:

```
BigIron(config)# no web-management
```

Syntax: [no] web-management [allow-no-password | enable [vlan <vlan-id>] | front-panel | hp-top-tools | list-menu]

Possible values:

The **allow-no-password** option disables password authentication for the Web management interface.

The **enable** option enables the Web management interface on the Foundry device.

The **front-panel** option causes the front panel frame, which contains a graphic depicting the Layer 2 Switch or Layer 3 Switch, to be displayed on the Web management interface.

The **hp-top-tools** parameter disables TCP port 280. By default, TCP ports 80 and 280 are enabled on the Foundry device. TCP port 80 (HTTP) allows access to the device's Web management interface. TCP port 280 allows access to the device by HP TOP-TOOLS.

The **list-menu** option causes the List (pre-06.0.00) menu to be displayed on the Web management interface, instead of the Tree menu.

Default value: Password authentication and the front panel are enabled by default. The List menu is disabled by default. (This means the Tree menu is enabled by default.)

web-management enable vlan

Allows Web management access only to clients in a specific VLAN.

EXAMPLE:

The following example configures the device to allow Web management access only to clients connected to ports within port-based VLAN 10. Clients connected to ports that are not in VLAN 10 are denied management access.

```
BigIron(config)# web-management enable vlan 10
```

Syntax: [no] web-management enable vlan <vlan-id>

Possible values: N/A

Default value: N/A

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config)# write memory
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen. In software releases 07.6.03 and later, this command also displays the size of the running-config file.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config)# write terminal
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 7
Interface Commands

appletalk address

Assigns AppleTalk addresses to a seed router.

EXAMPLE:

To assign an AppleTalk address of 10.5 to interface 3, module 2, enter the following:

```
BigIron(config)# int e 2/3
BigIron(config-if-2/3)# appletalk address 10.5
```

Syntax: appletalk address <node.network>

Possible values: N/A

Default value: N/A

appletalk cable-range

Assigns network numbers (cable range) to an AppleTalk seed router.

EXAMPLE:

To support network numbers from 10 – 50 on interface 3 (module 2):

```
BigIron(config)# int e 2/3
BigIron(config-if-2/3)# appletalk cable-range 10 - 50
```

Syntax: appletalk cable-range <network-number> | <network-number - network-number>

Possible values: N/A

Default value: N/A

appletalk deny

Restricts access to an AppleTalk zone.

EXAMPLE:

To deny Marketing (interface 2/1) and Field Service (interface e 2/3) zones access to the Finance server, enter the following:

```
BigIron(config)# int e 2/1
BigIron(config-if-2/1)# appletalk deny zone finance
BigIron(config-if-2/1)# exit
```

```
BigIron(config)# int e 2/3
BigIron(config-if-2/3)# appletalk deny zone finance
```

Syntax: appletalk deny zone <name> | additional-zones rtmp-filtering | no-rtmp-filtering

Possible values: N/A

Default value: N/A

appletalk deny additional-zones

Denies access to Appletalk zones not specifically addressed in permit zone filters.

EXAMPLE:

```
BigIron(config)# int e 2/1
BigIron(config-if-2/1)# appletalk permit zone HR
BigIron(config-if-2/1)# appletalk deny additional-zones
```

Syntax: appletalk deny additional-zones [rtmp-filtering | no-rtmp-filtering]

Possible values: The **rtmp-filtering** option causes the denied network numbers of the filtered zone to be removed from the RTMP packets.

Default value: N/A

appletalk permit

Allows access to an AppleTalk zone.

EXAMPLE:

To allow the Marketing (interface 2/1) and Field Service (interface e 2/3) zones access to the Finance server, enter the following:

```
BigIron(config)# int e 2/1
BigIron(config-if-2/1)# appletalk permit zone finance
BigIron(config-if-2/1)# exit
BigIron(config)# int e 2/3
BigIron(config-if-2/3)# appletalk permit zone finance
```

Syntax: appletalk permit zone <name>

Possible values: N/A

Default value: N/A

appletalk routing

Enables AppleTalk routing on a seed router.

You also can use this command, when preceded by **no** (**no appletalk routing**) to disable routing on an interface. Disable routing when you need to make configuration changes to the seed router. After all the changes are made, re-enable routing on the interface using the **appletalk routing** command.

EXAMPLE:

To enable AppleTalk routing on interface 2/1, enter the following:

```
BigIron(config)# int e 2/1
BigIron(config-if-2/1)# appletalk routing
```

Syntax: [no] appletalk routing

Possible values: N/A

Default value: N/A

appletalk zone-name

Assigns AppleTalk zones to a seed router.

EXAMPLE:

To assign Marketing and Sales zones to interface 2/1, enter the following:

```
BigIron(config)# int e 2/1
BigIron(config-if-2/1)# appletalk zone sales
BigIron(config-if-2/1)# appletalk zone marketing
```

Syntax: appletalk zone-name <name>

Possible values: N/A

Default value: N/A

aps authenticate

Specifies an authentication string that must be part of each packet sent between the process controlling a POS APS working interface and the process controlling a POS APS protect interface.

EXAMPLE:

```
BigIronA(config)# interface pos 2/1
BigIronA(config-posif-2/1)# aps working 1
BigIronA(config-posif-2/1)# aps authenticate mulvaney
BigIronA(config-posif-2/1)# exit
BigIronB(config)# interface pos 3/1
BigIronB(config-posif-3/1)# aps protect 1 10.0.0.1
BigIronB(config-posif-3/1)# aps authenticate mulvaney
BigIronB(config-posif-3/1)# exit
```

NOTE: The same authentication string must be configured on both the working and protect interfaces.

Syntax: [no] aps authenticate <string>

Possible values: N/A

Default value: N/A

aps force

Manually forces a POS APS protect interface to take over as a working interface. This command is useful when you want to bring down a working interface for maintenance purposes.

EXAMPLE:

```
BigIron(config)# interface pos 3/1
BigIron(config-posif-3/1)# aps protect 1 10.0.0.1
BigIron(config-posif-3/1)# aps force 1
BigIron(config-posif-3/1)# exit
```

Syntax: [no] aps force <circuit-number>

The <circuit-number> is a valid POS APS circuit number. In addition, you can specify 0 as the <circuit-number> (**aps force 0**) to manually force traffic from the protect interface to the working interface.

The switchover takes place immediately after you enter the command on the protect interface. The **aps force** command is not saved if you write the active configuration to memory.

Possible values: N/A

Default value: N/A

aps group

Allows more than one APS working/protect interface pair to be configured on a device.

EXAMPLE:

To configure more than one protect or working interface on a router, you assign each interface to a group using the **aps group** command. For example, suppose interface 3/1 on BigIronB serves as the protect interface to working interface 2/1 on BigIronA, and interface 2/2 on BigIronA serves as the protect interface to working interface 3/2 on BigIronB. To implement this configuration, you place each APS working/protect circuit pair in a separate APS group.

The following commands configure the working interface for APS group 50 and protect interface for APS group 60 on BigIronA:

```
BigIronA(config)# interface loopback 1
BigIronA(config-lbif-1)# ip address 10.0.0.1/24
BigIronA(config-lbif-1)# exit

BigIronA(config)# interface pos 2/1
BigIronA(config-posif-2/1)# aps group 50
BigIronA(config-posif-2/1)# aps working 1
BigIronA(config-posif-2/1)# exit

BigIronA(config)# interface pos 2/2
BigIronA(config-posif-2/2)# aps group 60
BigIronA(config-posif-2/2)# aps protect 1 10.0.0.2
BigIronA(config-posif-2/2)# exit
```

The following commands configure the working and protect interfaces on BigIronB:

```
BigIronB(config)# interface loopback 2
BigIronB(config-lbif-2)# ip address 10.0.0.2/24
BigIronB(config-lbif-2)# exit

BigIronB(config)# interface pos 3/1
BigIronB(config-posif-3/1)# aps group 50
BigIronB(config-posif-3/1)# aps protect 1 10.0.0.1
BigIronB(config-posif-3/1)# exit

BigIronB(config)# interface pos 3/2
BigIronB(config-posif-3/2)# aps group 60
BigIronB(config-posif-3/2)# aps working 1
BigIronB(config-posif-3/2)# exit
```

Syntax: [no] aps group <group-number>

Possible values: N/A

Default value: The default APS group number is 0.

aps lockout

Prevents an APS protect interface from taking over from a working interface. If the working interface malfunctions, no switchover to the protect interface takes place.

EXAMPLE:

To prevent a protect interface from becoming a working interface, enter commands such as the following on the protect interface:

```
BigIron(config)# interface pos 3/1
BigIron(config-posif-3/1)# aps protect 1 10.0.0.1
BigIron(config-posif-3/1)# aps lockout
BigIron(config-posif-3/1)# exit
```

Syntax: [no] aps lockout

Possible values: N/A

Default value: N/A

aps manual

Manually causes a switchover from a working interface to a protect interface. This command can be used when you want to bring a working interface down for maintenance, or if want to bring an interface back up without waiting for the WTR (wait-to-restore) interval specified by the **aps revert** command.

EXAMPLE:

To cause traffic on a working interface to switch over to the protect interface, enter commands such as the following:

```
BigIron(config)# interface pos 2/1
BigIron(config-posif-2/1)# aps working 1
BigIron(config-posif-2/1)# aps manual 1
BigIron(config-posif-2/1)# exit
```

Syntax: [no] aps manual <circuit-number>

Possible values: The <circuit-number> is a valid POS APS circuit number. In addition, you can specify 0 as the <circuit-number> (**aps manual 0**) to manually force traffic from the protect interface to the working interface.

Default value: N/A

aps protect

Specifies the circuit this interface is protecting, as well as the IP address of the router where the working interface resides.

EXAMPLE:

```
BigIronB(config)# interface pos 3/1
BigIronB(config-posif-3/1)# aps group 1
BigIronB(config-posif-3/1)# aps protect 1 10.0.0.1
BigIronB(config-posif-3/1)# exit
```

Syntax: [no] aps protect <circuit-number> <ip-addr>

Possible values: The <circuit-number> is the circuit this interface is protecting; the <ip-addr> is the IP address of the router where the working interface resides. The <ip-addr> is normally the loopback address of the router. If the device is a Layer 2 Switch, this is the management IP address.

Default value: N/A

aps revert

Configures an APS interface to automatically revert to being a working interface after it has been available for a specified amount of time.

EXAMPLE:

To cause an interface to switch back to being a working interface 3 minutes after becoming available, enter commands such as the following:

```
BigIron(config)# interface pos 2/1
BigIron(config-posif-2/1)# aps protect 1
BigIron(config-posif-2/1)# aps revert 3
BigIron(config-posif-2/1)# exit
```

Syntax: [no] aps revert <minutes>

Possible values: N/A

Default value: N/A

aps timers

Configures the interval at which the process controlling the POS APS protect interface sends hello packets to the process controlling the working interface, as well as how long the process controlling the protect interface waits for a response before declaring the working interface down.

EXAMPLE:

To configure the protect interface process to send hello packets every 3 seconds and wait a maximum of 6 seconds for a response, enter commands such as the following on the protect interface:

```
BigIron(config)# interface pos 3/1
BigIron(config-posif-3/1)# aps protect 1 10.0.0.1
BigIron(config-posif-3/1)# aps timers 3 6
BigIron(config-posif-3/1)# exit
```

Syntax: [no] aps timers <hello-timer> <response-timer>

Possible values:

The <hello-timer> is the interval between hello packets. The default is 1 second.

The <response-timer> is the amount of time the protect interface process waits for a response from the working interface process before declaring the working interface down. The default is 3 seconds.

Default value: See above.

aps working

Configures a POS APS working interface

EXAMPLE:

```
BigIronA(config)# interface pos 2/1
BigIronA(config-posif-2/1)# aps working 1
BigIronA(config-posif-2/1)# exit
```

Syntax: [no] aps working <circuit-number>

Possible values: The <circuit-number> on the working interface corresponds to a circuit number on a protect interface.

Default value: N/A

atm-hcsadd

Disables or re-enable the CRC for header checksums in ATM cells.

By default, Foundry ATM ports add a CRC for the header checksum in an ATM cell. Some ATM switches add the CRC by default while others do not. Check the documentation for your ATM switch to determine the setting for your switch. The setting of the CRC mode affects both send and receive traffic on the port.

EXAMPLE:

To disable the CRC for ATM interface 4/1, enter the following commands:

```
BigIron(config)# interface atm 4/1
BigIron(config-atmif-4/1)# no atm-hcsadd
```

Syntax: [no] atm-hcsadd

To re-enable the feature, enter the following command:

```
BigIron(config-atmif-4/1)# atm-hcsadd
```

Possible values: N/A

Default value: Enabled

atm pvc

Configures a Virtual Circuit (VC). You can configure the following parameters:

- The link type (point-to-point or point-to-multipoint)
- The VC's VPI and VCI
- The VC's CoS type and the values for the applicable CoS parameters
- For point-to-multipoint links, the source of the remote IP address (user-configured or learned through Inverse ARP)

Configure the parameters to match the VC and CoS settings on the ATM switch at the other end of the link.

EXAMPLE:

To configure a PVC and specify CoS parameters for the PVC, first enter a command such as the following to change the CLI to the ATM sub-interface configuration level:

```
BigIron(config)# interface atm 4/1.1
BigIron(config-subif-4/1.1)#
```

The command in this example changes the CLI to the configuration level for sub-interface 1 on port 1 of the ATM module in chassis slot 4. The CLI prompt changes to indicate the configuration level.

Syntax: interface atm <slot>/<port>.<subif>

The <slot>/<port> parameter specifies the chassis slot and the port number on the ATM module in the specified slot. The <subif> parameter specifies a sub-interface on the ATM port. You can configure up to 4095 sub-interfaces on an ATM port, with a per chassis maximum of 4095 total. Specify a number from 1 – 4095 for the sub-interface.

After changing to the configuration level for a sub-interface, enter a command such as the following:

```
BigIron(config-subif-4/1.1)# atm pvc 1 200 cbr 10000
```

This command adds a PVC with virtual path 1 and virtual channel 200, and configures CoS parameters for CBR. The **cbr** parameter indicates that the CoS method is CBR. The value following **cbr** indicates the Peak Cell Rate (PCR), in this case 10,000 kilobits per second (10 Mbps).

The following command configures a PVC for VBR:

```
BigIron(config-subif-4/1.1)# atm pvc 1 300 vbr 10000 5000 200
```

This example specifies that the maximum data rate on ATM VC 1, 300 is 10000 kilobits, while the average data rate is 5000 kilobits. The VC can accommodate a maximum burst size of 200 cells.

Syntax: [no] atm pvc <vpi> <vci> cbr <pcr>

or

Syntax: [no] atm pvc <vpi> <vci> vbr <pcr> <scr> <mbs>

or

Syntax: [no] atm pvc <vpi> <vci> ubr

The **pvc** <vpi> <vci> parameter specifies the virtual path and virtual channel of the PVC. Each PVC has a unique combination of virtual path and virtual channel. You can specify a number from 0 – 3 for the VPI. You can specify a number from 0 – 1023 for the VCI. You can use a given VPI and VCI combination only once on a given ATM port. However, you can use the same combination on other ATM ports on the same module or in the same chassis.

The **cbr** <pcr> parameter specifies CBR as the CoS method and specifies the PCR. The <pcr> can be from 1 – 155000 kilobits.

The **vbr** <pcr> <scr> <mbs> parameter specifies VBR as the CoS method, and specifies the PCR, Sustained Cell Rate (SCR), and Maximum Burst Size (MBS).

- The <pcr> can be from 1 – 155000 kilobits.
- The <scr> can be from 1 – 155000 kilobits.
- The <mbs> can be from 2 – 255.

The **ubr** parameter specifies UBR as the CoS method.

NOTE: The CLI checks the values you enter for the <pcr> and <scr> parameters to make sure that the combined cell rates on all the PVCs configured on the port do not exceed the bandwidth of the port.

NOTE: ABR is not supported and no support is planned.

EXAMPLE:

To configure a point-to-multipoint ATM interface, enter commands such as the following:

```
BigIron(config)# interface atm 4/1.1 multipoint
BigIron(config-subif-4/1.1)# atm pvc 1 1 cbr 10000 ip inarp
BigIron(config-subif-4/1.1)# atm pvc 1 2 cbr 20000 ip inarp
BigIron(config-subif-4/1.1)# atm pvc 1 3 cbr 25000 ip inarp
BigIron(config-subif-4/1.1)# ip address 10.10.10.4
```

The first command configures a point-to-multipoint ATM interface. The remaining commands configure PVCs on the interface. In this example, all three PVCs use Inverse ARP to resolve the remote IP addresses.

Syntax: [no] interface atm <slot>/<port>.<subif> [multipoint | point-to-point]

Syntax: [no] atm pvc <vpi>< vci> cbr | ubr | vbr ip <remote-ip-addr> | inarp [<mins>]

The **multipoint** | **point-to-point** parameter specifies the type of ATM interface.

- **multipoint** – The sub-interface can have more than one PVC.
- **point-to-point** – The sub-interface can have only one PVC. This is this default.

NOTE: Once you create the sub-interface, you cannot change the interface type.

The **ip** <remote-ip-addr> | **inarp** <mins> parameter specifies how the PVC's remote IP address is obtained.

- <remote-ip-addr> – The address is obtained statically, when you specify the address using this parameter.
- **inarp** [<mins>] – The address is obtained dynamically using Inverse ARP. The <mins> parameter specifies how often the software sends a new Inverse ARP to refresh the remote IP address mapped to the PVC. You can specify from 1 – 60 minutes. The default is 10 minutes.

You must specify an IP address or **inarp**. There is no default.

To configure a point-to-multipoint interface with PVCs that use remote IP address that you specify, enter commands such as the following:

```
BigIron(config)# interface atm 4/1.1 multipoint
BigIron(config-subif-4/1.1)# atm pvc 1 1 ubr ip 10.10.10.1
BigIron(config-subif-4/1.1)# atm pvc 1 2 ubr ip 10.10.10.2
BigIron(config-subif-4/1.1)# atm pvc 1 3 ubr ip 10.10.10.3
BigIron(config-subif-4/1.1)# ip address 10.10.10.4
```

To display the point-to-multipoint mappings, see "show atm map" on page 43-14.

To clear the mappings, see "clear atm map" on page 5-8.

Possible values: See above.

Default value: None configured.

atm-scram

Disables or re-enable the ATM scramble mode.

By default, Foundry ATM ports perform bit scrambling at the ATM cell level. Most ATM scramble mode affects both send and receive traffic on the port.

EXAMPLE:

To disable the ATM scramble mode for ATM interface 4/1, enter the following commands:

```
BigIron(config)# interface atm 4/1
BigIron(config-atmif-4/1)# no atm-scram
```

Syntax: [no] atm-scram

To re-enable the feature, enter the following command:

```
BigIron(config-atmif-4/1)# atm-scram
```

Possible values: N/A

Default value: Enabled

atm vc-per-vp

Changes the maximum number of VCs per VP on an ATM port.

By default, an ATM port can have up to four VPs and each VP can have up to 1024 VCs. You can change, on an individual port basis, the maximum number of VCs allowed in a VP to one of the following values:

- 4096
- 2048
- 1024 (the default)
- 256

When you change the maximum number of VCs a VP can have, the maximum number of VPs the port can have also is changed.

Maximum Number of VCs per VP	Maximum Number of VPs
4096	1
2048	2
1024	4
256	16

NOTE: The total number of VCs an ATM port can have is still 4096.

NOTE: When you change the maximum number of VCs per VP, the software deletes all configured PVCs from the port.

EXAMPLE:

To change the maximum number of VCs per VP, enter commands such as the following:

```
BigIron(config)# interface atm 3/1
BigIron(config-atmif-3/1)# atm vc-per-vp 256
Changing the number of VCs per VP will remove all current PVCs configured on this
atm port.
Do you want to continue?(enter 'y' or 'n'):
```

These commands change the CLI to the configuration level for ATM port 3/1, and change the maximum number of VCs per VP to 256, which increases the maximum number of VPs to 16. The software asks you to verify that you want to make the change, since all PVCs already configured on the port will be removed.

Syntax: [no] atm vc-per-vp 256 | 1024 | 2048 | 4096

To display how many VCs a VP can have on a port, enter a command such as the following:

```
BigIron(config)# show interface atm 3/1
No port name
  Hardware is ATM
  Encapsulation llcsnap, clock is internal
  Framing is SONET, BW 155000Kbit
  Loopback not set, keepalive not set, scramble enabled
  Each virtual path contains 4096 virtual channels
  300 second input rate: 0 bits/sec, 0 packets/sec
  300 second output rate: 0 bits/sec, 0 packets/sec
  0 packets input, 0 bytes, 0 no buffer
  Received 0 CRCs, 0 shorts, 0 giants, 0 alignments
  0 packets output, 0 bytes, 0 underruns
```

The line shown in bold type indicates how many VCs each VP on the port can have.

Possible values: See above.

Default value: 4 VCs per VP and up to 1024 VPs

auto-gig

Enables the auto-Gigabit 802.3x negotiation mode on a Stackable Gigabit interface. Both sides of the circuit need to be configured with this feature.

NOTE: This command applies only to Stackable devices. To configure the negotiation mode on a Chassis device, see "gig-default" on page 7-16.

EXAMPLE:

```
BigIron(config)# int e 1/4
```

```
BigIron(config-if-1/4)# auto-gig
```

Syntax: [no] auto-gig

Possible values: on or off

Default value: disabled

bandwidth

Allows you to change the bandwidth of a POS port. Depending on the POS module you have installed, the interfaces operate at 155 Mbps or 622 Mbps by default.

- The ports on model number B2P622 run at 622 Mbps by default. If needed, you can reduce the port bandwidth to 155 Mbps on an individual port basis.
- The ports on model number B2P155 run at 155 Mpbs and cannot be configured to run at 622 Mbps.

EXAMPLE:

To change the bandwidth of POS interface 2/1 from 622 Mbps to 155 Mbps, enter the following commands:

```
BigIron(config)# interface pos 2/1
BigIron(config-posif-2/1)# bandwidth 155
```

Syntax: bandwidth 155 | 622

Possible values: The **155** and **622** parameters specify how many megabits per second the port can transmit.

Default value: See above.

broadcast limit

Specifies the maximum number of broadcast packets the port can forward each second.

EXAMPLE:

```
BigIron(config-if-1/3)# broadcast limit 80000
```

Syntax: broadcast limit <num>

Possible values: 0 – 4294967295; if you specify 0, limiting is disabled.

Default value: N/A

cdp enable

Enables or disables CDP at the interface level. This command is available in software releases 07.6.03 and later.

EXAMPLE:

To enable CDP on port 2/1, enter commands such as the following:

```
BigIron(config)# int e 2/1
BigIron(config-if-2/1)# cdp enable
```

For an ATM interface, you can enable or disable CDP at the sub-interface level by entering commands such as the following:

```
BigIron(config)# int atm 2/1.1
BigIron(config-subif-2/1.1)# cdp enable
```

Syntax: [no] cdp enable

Possible value: N/A

Default value: By default, the feature is enabled on an interface once CDP is enabled on the device.

clock

Changes the clock source for a Foundry ATM port or POS module.

EXAMPLE:

To change the clock source for POS interface 2/1 to internal (the POS module itself):

```
BigIron(config)# interface pos 2/1
BigIron(config-posif-2/1)# clock internal
```

EXAMPLE:

To change the clock source for ATM interface 4/1 to line (the network), enter the following commands:

```
BigIron(config)# interface atm 4/1
BigIron(config-atmif-4/1)# clock line
```

Syntax: [no] clock internal | line

The **internal** and **line** parameters specify whether the clock source is on the module (internal) or on the network (line).

Possible values: See above

Default value: internal for ATM; line for POS

crc

Specifies whether the CRC (Cyclic Redundancy Check) portion of each frame transmitted on a POS interface is 16 bits or 32 bits long. The default is 32 bits.

EXAMPLE:

To change the CRC length for POS interface 2/1 to 16 bits:

```
BigIron(config)# interface pos 2/1
BigIron(config-posif-2/1)# crc 16
```

Syntax: crc 16 | 32

Possible values: The **16** and **32** parameters specify how many bits in each frame transmitted on the interface contain the CRC data.

Default value: 32

dhcp-gateway-list

This parameter assigns a defined DHCP gateway list to a specific interface on a Foundry Layer 2 Switch. DHCP gateway lists must be defined at the Global CONFIG level and the DHCP Assist feature enabled to support assignment of this feature on switches.

This feature is not supported on Foundry Layer 3 Switches.

NOTE: For more details on this command and the DHCP Assist feature, see the "Configuring IP" chapter of the *Foundry Enterprise Configuration and Management Guide*.

EXAMPLE:

To assign a defined DHCP gateway list (1) to interface 2/5, enter the following:

```
BigIron(config)# int e 2/5
BigIron(config-if-2/5)# dhcp-gateway-list 1
```

Syntax: dhcp-gateway-list <number>

Possible values: N/A

Default value: N/A

disable

Disables a specific interface.

EXAMPLE:

```
BigIron(config)# interface e 1/5
BigIron(config-if-1/5)# disable
```

EXAMPLE:

```
BigIron(config)# interface v 6
BigIron(config-vif-6)# disable
```

Syntax: disable

Possible values: N/A

Default value: N/A

dot1x multiple-hosts

Allows multiple 802.1X Clients to connect on the same port.

NOTE: When the **port-control** parameter on an 802.1X-enabled interface is set to **force-authorized**, the Foundry device allows connections from multiple Clients, regardless of whether the **multiple-hosts** parameter is used in the interface's configuration.

EXAMPLE:

```
BigIron(config-if-3/1)# dot1x multiple-hosts
```

Syntax: [no] dot1x multiple-hosts

Possible values: N/A

Default value: disabled

dot1x port-control

Activates authentication on an 802.1X-enabled interface

EXAMPLE:

```
BigIron(config)# interface e 3/1
BigIron(config-if-3/1)# dot1x port-control auto
```

Syntax: [no] dot1x port-control [force-authorized | force-unauthorized | auto]

Possible values: The port control type can be one of the following:

force-authorized – The port's controlled port is placed unconditionally in the authorized state, allowing all traffic. This is the default state for ports on the Foundry device.

force-unauthorized – The controlled port is placed unconditionally in the unauthorized state.

auto – The controlled port is unauthorized until authentication takes place between the Client and Authentication Server. Once the Client passes authentication, the port becomes authorized. This has the effect of activating authentication on an 802.1X-enabled interface.

Default value: force-authorized

dual-mode

Configures a tagged VLAN port as a dual-mode port. 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).

NOTE: If you plan to use dual-mode ports, do not configure any of the ports in the default VLAN in a trunk group and do not configure the dual-mode ports in a trunk group.

EXAMPLE:

```
BigIron(config)# vlan 20
BigIron(config-vlan-20)# tagged e 2/11
BigIron(config-vlan-20)# tagged e 2/9
BigIron(config-vlan-20)# int e 2/11
BigIron(config-if-e100-2/11)# dual-mode
BigIron(config-if-e100-2/11)# exit
```

Syntax: [no] dual-mode [<vlan-id>]

Possible values: The <vlan-id> parameter causes the port to transmit traffic for the specified VLAN as untagged, while transmitting traffic for other VLANs as tagged. If this parameter is not specified, the port transmits untagged traffic on the DEFAULT-VLAN.

Default value: Disabled

enable

Enables a specific interface. All interfaces are enabled at initial startup. This command is necessary only if an interface has been disabled.

EXAMPLE:

```
BigIron(config)# interface e 1/5
BigIron(config-if-1/5)# enable
```

EXAMPLE:

```
BigIron(config)# interface v 6
BigIron(config-vif-6)# enable
```

Syntax: enable

Possible values: N/A

Default value: All ports are enabled at system startup.

encapsulation

Changes the encapsulation type of a Foundry POS interface. Foundry POS interfaces use the PPP encapsulation type by default.

NOTE: Both ends of the POS link must use the same encapsulation type.

NOTE: HDLC encapsulation is not supported on Layer 2 POS. You must use the default encapsulation, PPP.

EXAMPLE:

To configure POS interface 2/1 to use HDLC:

```
BigIron(config)# interface pos 2/1
BigIron(config-posif-2/1)# encapsulation hdlc
```

Syntax: [no] encapsulation hdlc | ppp | frame-relay [ietf]

The **ietf** parameter applies only to Frame Relay and enables the IETF (RFC 1490) Frame Relay type instead of the default (Cisco-compatible). If you enter the **encapsulation frame-relay** command without the **ietf** parameter, the software uses the default Frame Relay type, Cisco-compatible.

Possible values: See above

Default value: ppp

end

Moves activity to the privileged level from any level of the CLI except the User EXEC level.

EXAMPLE:

To move to the privileged level, enter the following:

```
BigIron(config-if-5/3)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level of the CLI. This command is available at all levels.

EXAMPLE:

To move from the Interface level back to the global CONFIG level, enter the following:

```
BigIron(config-if-4/3)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

fdp enable

Enables or disables FDP at the interface level. This command is available in software releases 07.6.03 and later.

EXAMPLE:

To enable FDP on ethernet interface 2/1, enter the following:

```
BigIron(config)# int e 2/1
BigIron(config-if-2/1)# fdp enable
```


For an ATM interface, you can enable or disable FDP at the sub-interface level by entering commands such as the following:

```
BigIron(config)# int atm 2/1.1
BigIron(config-subif-2/1.1)# fdp enable
```

Syntax: [no] fdp enable

Possible values: N/A

Default value: By default, the feature is enabled on an interface once FDP is enabled on the device.

flow-control

Allows you to turn flow control (802.3x) for full-duplex ports on or off (no). Flow control is on by default.

EXAMPLE:

To turn the feature off, enter the following:

```
BigIron(config)# int e 1/5

BigIron(config-if-1/5)# no flow control
```

To turn the feature on after being turned off, enter the following:

```
BigIron(config-if-1/5)# flow-control
```

Syntax: [no] flow-control

Possible values: N/A

Default value: on

frame-relay

Configures parameters for a POS Frame Relay interface. You can configure the following parameters:

- Frame Relay interface type.

  NOTE: The current software release supports Data Terminal Equipment (DTE) only. The other end of the link must be configured as a DCE link.

- Data-Link Connection Identifier (DLCI). This is the circuit ID for the link and can be a number in the range from 1 – 1023. The circuit ID must be the same on both ends of the link.
- Local Management Interface (LMI) type. You can specify ANSI, CCITT, or LMI (Cisco-compatible). The default is LMI.

EXAMPLE:

```
BigIron(config-posif-2/1)# frame-relay interface-dlci 69
BigIron(config-posif-2/1)# frame-relay lmi-type ansi
```

Syntax: [no] frame-relay intf-type dte

NOTE: The current software release supports Data Terminal Equipment (DTE) only. The other end of the link must be configured as a DCE link.

Syntax: [no] frame-relay interface-dlci <num>

The <num> parameter specifies the DLCI and can be a number from 1 – 1023.

Syntax: [no] frame-relay lmi-type ansi | ccitt | lmi

The default is **lmi**.

Possible values: See above

Default value: See above

gig-default

Overrides the global default setting for Gigabit negotiation mode. You can configure the Gigabit negotiation mode for a port to be one of the following:

- Default – The port uses the negotiation mode that was set at the global level.
- 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 for Chassis devices (including the TurboIron/8).
- 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.

See the "Configuring Basic Features" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide* for more information.

NOTE: This command does not apply to Stackable devices. To change the negotiation mode for a Stackable Gigabit Ethernet port, use the **[no] auto-gig** command at the Interface level. See “auto-gig” on page 7-10.

EXAMPLE:

To override the global setting and set the negotiation mode to auto-Gigabit for ports 4/1 – 4/4, enter the following commands:

```
BigIron(config)# int ethernet 4/1 to 4/4
BigIron(config-mif-4/1-4/4)# gig-default auto-gig
```

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

Possible values: see above

Default value: neg-full-auto

hello padding

Disables IS-IS padding.

By default, the Layer 3 Switch adds extra data to the end of a hello packet to make the packet the same size as the maximum length of PDU the Layer 3 Switch supports.

The padding applies to the following types of hello packets:

- ES hello (ESH PDU)
- IS hello (ISH PDU)
- IS to IS hello (IIH PDU)

The padding consists of arbitrarily valued octets. A padded hello PDU indicates the largest PDU that the Layer 3 Switch can receive. Other ISs that receive a padded hello PDU from the Layer 3 Switch can therefore ensure that the IS-IS PDUs they send the Layer 3 Switch. Similarly, if the Layer 3 Switch receives a padded hello PDU from a neighbor IS, the Layer 3 Switch knows the maximum size PDU that the Layer 3 Switch can send to the neighbor.

When padding is enabled, the maximum length of a Hello PDU sent by the Layer 3 Switch is 1514 bytes.

If you need to disable padding, you can do so globally or on individual interfaces. Generally, you do not need to disable padding unless a link is experiencing slow performance, for example due to point-to-point interoperability issues. If you enable or disable padding on an interface, the interface setting overrides the global setting.

By default, disabling or re-enabling padding affects hello PDUs sent on point-to-point circuits and to an IS-IS broadcast address. You can specify an option to enable or disable the padding for point-to-point or broadcast PDUs.

NOTE: To disable or re-enable padding globally, see “hello padding” on page 12-4.

EXAMPLE:

To globally disable padding of IS-IS hello PDUs, enter the following command:

```
NetIron(config-isis-router)# no hello padding
```

This command disables all hello PDU padding on the Layer 3 Switch. To re-enable padding, enter the following command:

```
NetIron(config-isis-router)# hello padding
```

Syntax: [no] hello padding [point-to-point]

The **point-to-point** parameter disables or re-enables the padding only for point-to-point connections.

Possible values: See above

Default value: Enabled

ip access-group

Applies an ACL to an interface.

EXAMPLE:

To configure a standard ACL and apply it to outgoing traffic on port 1/1, enter the following commands.

```
BigIron(config)# access-list 1 deny host 209.157.22.26 log
BigIron(config)# access-list 1 deny 209.157.29.12 log
BigIron(config)# access-list 1 deny host IPHost1 log
BigIron(config)# access-list 1 permit any
BigIron(config)# int eth 1/1
BigIron(config-if-1/1)# ip access-group 1 out
BigIron(config)# write memory
```

The commands in this example configure an ACL to deny packets from three source IP addresses from being forwarded 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.

Syntax: [no] ip access-group <num> in | out

The <num> parameter is the access list number and can be from 1 – 99.

The **in | out** parameter specifies whether the ACL applies to incoming traffic or outgoing traffic on the sub-interface.

EXAMPLE:

To configure an IP ACL and apply it to an ATM sub-interface, enter commands such as the following:

```
BigIron(config)# access-list 1 deny host 209.157.22.26 log
BigIron(config)# access-list 1 permit any
BigIron(config)# int atm 4/1.1
BigIron(config-atmif-4/1.1)# ip access-group 1 out
```

Syntax: [no] ip access-group <num> in | out

EXAMPLE:

To apply an ACL to a subset of ports within a virtual interface, enter commands such as the following:

```
BigIron(config)# vlan 10 name IP-subnet-vlan
BigIron(config-vlan-10)# untag ethernet 1/1 to 2/12
BigIron(config-vlan-10)# router-interface ve 1
BigIron(config-vlan-10)# exit
BigIron(config)# access-list 1 deny host 209.157.22.26 log
BigIron(config)# access-list 1 deny 209.157.29.12 log
BigIron(config)# access-list 1 deny host IPHost1 log
```



```
BigIron(config)# access-list 1 permit any
BigIron(config)# interface ve 1
BigIron(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 <num> in ethernet <portnum> [<portnum>...] to <portnum>

Possible values: see above

Default value: N/A

ip access-group flow-mode

Enables or disables flow-based ACL mode on a JetCore interface.

NOTE: This command applies only to JetCore devices running software release 07.6.01 or later.

By default, the device enables hardware-based ACLs on all ports.

Disabling hardware-based ACLs on some of the ports is useful if some ports have large ACLs (ACLs with many entries) while other ports have few ACL entries. By disabling hardware-based ACLs on the ports that have few or no ACL entries, you can ensure that the ports that do have ACL entries will have enough CAM space for the ACL entries.

You also might want to disable hardware-based ACLs if the ACL entries on a port are used infrequently. In this case, you can conserve CAM entries for other features or other ports with minimal performance impact, since the ACL activity is low.

NOTE: You can determine the ports that have high ACL usage by disabling hardware-based ACLs on all the ports, enabling ACL accounting, and observing the results. After you determine the ports that use ACLs the most, you can selectively re-enable hardware-based ACLs on those ports.

EXAMPLE:

To disable hardware-based ACLs on a port, enter the following command at the configuration level for the port:

```
BigIron(config-if-1/1)# ip access-group flow-mode
```

Syntax: [no] ip access-group flow-mode

To re-enable hardware-based ACLs on the port, enter the following command:

```
BigIron(config-if-1/1)# no ip access-group flow-mode
```

Possible values: N/A

Default value: Flow-based ACLs are disabled

ip access-group frag deny

Configures an interface to automatically drop a fragment whose source and destination IP addresses exactly match an ACL entry that has Layer 4 information, even if that ACL entry's action is permit.

NOTE: This section applies only to IronCore devices.

For a packet fragment that is sent to the CPU for processing, the device compares the fragment's source and destination IP addresses against the interface's ACL entries. By default, if the fragment's source and destination IP addresses exactly match an ACL entry that also has Layer 4 information (source and destination TCP or UDP application ports), the device permits or denies the fragment according to the ACL.

EXAMPLE:

```
BigIron(config)# interface ethernet 1/1
```



```
BigIron(config-if-1/1)# ip access-group frag deny
```

Syntax: [no] ip access-group frag deny

Possible values: N/A

Default value: The interface will forward subsequent fragments of the same packet in hardware if there is a CAM entry for the fragment, or will permit or deny the fragment if its source and destination IP addresses exactly match an ACL entry that has Layer 4 information.

ip access-group frag deny

Configures an interface to automatically drop all fragmented packets.

NOTE: This section applies only to JetCore devices.

EXAMPLE:

```
BigIron(config)# interface ethernet 1/1
BigIron(config-if-1/1)# ip access-group frag deny
```

Syntax: [no] ip access-group frag deny

Possible values: N/A

Default value: The interface will forward subsequent fragments of the same packet in hardware if there is a CAM entry for the fragment, or will permit or deny the fragment if its source and destination IP addresses exactly match an ACL entry that has Layer 4 information.

ip access-group frag inspect

Enables CPU filtering of all packet fragments on the interface. When you enable CPU filtering, the interface sends all the fragments of a fragmented packet to the CPU. The CPU then permits or denies each fragment according to the ACL applied to the interface.

NOTE: The fragmentation support described in this section applies only to JetCore devices and only to hardware-based ACLs.

By default, when a hardware-based ACL is applied to an interface, the interface will use the ACL to permit or deny the first fragment of a fragmented packet, but forward subsequent fragments of the same packet in hardware.

NOTE: By default, 10 Gigabit Ethernet modules also forward the first fragment instead of using the ACLs to permit or deny the fragment.

EXAMPLE:

```
BigIron(config)# interface ethernet 1/1
BigIron(config-if-1/1)# ip access-group frag inspect
```

Syntax: [no] ip access-group frag inspect

The **inspect** parameter sends all fragments to the CPU.

ip access-group max-l4-cam

Changes the number of CAM entries allocated for hardware-based ACLs applied to the interface.

NOTE: This command applies only to JetCore devices.

You can adjust the allocation of Layer 4 CAM space for use by ACLs, on an IPC or IGC basis or on 10 Gigabit Ethernet modules. The new allocation applies to all the ports managed by the IPC or IGC.

Most ACLs require one CAM entry for each ACL entry (rule). The exception is an ACL entry that matches on more than one TCP or UDP application port. In this case, the ACL entry requires a separate Layer 4 CAM entry for each application port on which the ACL entry matches.

Make sure you specify a maximum that is equal to or greater than the largest number of entries required by an ACL applied to any of the ports managed by the same IPC or IGC or 10 Gigabit Ethernet module. For example, if port 1 on a FastIron 4802 will have an ACL that requires 250 entries, make sure 250 is the lowest number of entries you specify for any port on IPC 1 (the IPC that manages ports 1 – 24).

Hardware-based ACLs use CAM partitions 1 and 2. The default number of entries that are allocated in each pool differs depending on the device.

EXAMPLE:

To specify the maximum number of CAM entries the device can allocate for hardware-based ACLs, enter commands such as the following:

```
BigIron(config)# interface ethernet 1/1
BigIron(config-if-1/1)# ip access-group max-l4-cam 50
```

This command allows up to 50 ACL entries on each port managed by the IPC or IGC that manages port 1/1.

Syntax: [no] ip access-group max-l4-cam <num>

The <num> parameter specifies the number of CAM entries and can be from 10 – 8192. The default depends on the device.

The command is valid at the interface configuration level. However, the device applies the change to all ports managed by the same IPC or IGC or 10 Gigabit Ethernet module. Regardless of the port number, when you save the change to the startup-config file, the CLI applies the command to the first port managed by the IPC or IGC. For example, if you enter the command on port 3 of a FastIron 4802, when you save the configuration change, the CLI enters the **ip access-group max-l4-cam** command under port 1 in the startup-config file.

NOTE: If you enter the command on more than one port managed by the same IPC or IGC, the CLI uses the value entered with the most-recent command for all the ports on the ICP or IGC.

Possible values: 10 – 8192

Default value: Depends on the device

ip access-group redirect-deny-to-interf

Copies traffic denied by a JetCore hardware-based ACL to a mirror port for monitoring.

Although hardware-based ACLs do not support ACL logging, you nonetheless can monitor the traffic denied by hardware-based ACLs. To do so, attach a protocol analyzer to a port and enable the device to redirect traffic denied by ACLs to that port.

When you enable redirection, the deny action of the ACL entry is still honored. Traffic that matches the ACL is not forwarded on the port that has the ACL.

EXAMPLE:

```
BigIron(config-if-1/1)# ip access-group redirect-deny-to-interf
```

Syntax: [no] ip access-group redirect-deny-to-interf

Enter the command on the port to which you want the denied traffic to be copied.

Possible values: N/A

Default value: Disabled

ip access-group ve-traffic

Enables filtering of traffic switched within a virtual routing interface.

By default, the device does not filter traffic that is switched from one port to another within the same virtual routing interface, even if an ACL is applied to the interface. You can enable the device to filter switched traffic within a virtual routing interface. When you enable the filtering, the device uses the ACLs applied to inbound traffic to filter traffic received by a port from another port in the same virtual routing interface. This feature does not apply to ACLs applied to outbound traffic.

EXAMPLE:

```
BigIron(config-vif-1)# ip access-group ve-traffic
```

Syntax: [no] ip access-group ve-traffic

Possible values: N/A

Default value: Disabled

ip access-policy-group

Applies an IP access policy group to an interface on a Layer 3 Switch and defines whether the policies are applied to incoming packets, outgoing packets, both. You must configure the access policies using the **ip access-policy** command before you can apply them using this command. See "ip access-policy" on page 6-39.

This command is not supported on Foundry Layer 2 Switches.

NOTE: For backward compatibility, the Layer 3 Switches also support the **ip filter-group** and **ip policy-group** commands. The parameters are the same as those for the **ip access-policy-group** command.

EXAMPLE:

To apply IP access policies 2, 3, and 4 to interface 1 (module 4), enter the following commands:

```
BigIron(config)# int e 4/1
BigIron(config-if-4/1)# ip filter-gr in 2 3 4
```

EXAMPLE:

You also can specify policy ranges. For example, to apply policies 1 – 3, policy 9, and policies 11 – 25 to port 2/4's outbound policy group, enter the following commands:

```
BigIron(config)# int ethernet 2/4
BigIron(config-if-2/4)# ip access-policy-group out 1 to 3 9 11 to 25
```

Syntax: ip access-policy-group in | out <policy-list>

Possible values: access policy numbers; enter all the policies you want to apply on the same command.

Default value: N/A

ip address

Configures an IP interface. You can configure multiple IP addresses as routing interfaces on a Layer 3 Switch.

NOTE: You can increase the total number of IP sub-net interfaces that you can configure on the Layer 3 Switch. See "system-max" on page 6-131.

EXAMPLE:

```
BigIron(config)# int e 2/3
BigIron(config-if-2/3)# ip address 192.55.6.54 255.255.0.0
```

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.

EXAMPLE:

To configure an IP address on an ATM sub-interface, enter a command such as the following:

```
BigIron(config)# interface atm 4/1.1
BigIron(config-atmif-4/1.1)# ip address 192.168.2.9 255.255.255.0
```

EXAMPLE:

To use the OSPF options when configuring IP addresses, enter commands such as the following:

```
BigIron(config)# interface ethernet 1/1
BigIron(config-if-1/1)# ip address 1.1.1.1/24
BigIron(config-if-1/1)# ip address 1.1.2.1/24 ospf-passive
BigIron(config-if-1/1)# ip address 1.1.3.1/24 ospf-ignore
```

These commands configure three IP addresses on port 1/1. The first address does not use the OSPF options, and thus is eligible to form OSPF adjacencies and be advertised into OSPF. The second address uses the **ospf-passive** option, which prevents the address from forming OSPF adjacencies. However, the address still can be advertised into OSPF. The third address cannot form OSPF adjacencies and cannot be advertised into OSPF.

EXAMPLE:

To delete an IP address, enter a command such as the following:

```
BigIron(config-if-1/1)# 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:

```
BigIron(config-if-1/1)# no ip address *
```

Syntax: no ip address <ip-addr> | *

Possible values: Valid IP address

Default value: N/A

ip arp-age

Overrides the globally configured IP ARP age on an individual interface.

EXAMPLE:

```
BigIron(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.

Possible values: 0 – 240 minutes

Default value: the globally configured value, which is 10 minutes by default

ip bootp-gateway

Specifies the interface address the router should use for stamping BootP/DHCP packets. Use this command when the interface has multiple IP sub-net addresses. By default, the router uses the lowest numbered IP address for stamping BootP/DHCP requests.

EXAMPLE:

```
BigIron(config)# int e 2/3

BigIron(config-if-2/3)# ip bootp-gateway 192.55.6.54
```

Syntax: ip bootp-gateway <ip-addr>

Possible values: Valid IP address

Default value: N/A

ip directed-broadcast

Enables or disables forwarding of directed IP broadcasts on an individual interface on a Layer 3 Switch.

EXAMPLE:

```
BigIron(config)# interface ethernet 1/1
BigIron(config-if-1/1)# ip directed-broadcast
```

Syntax: [no] ip directed-broadcast

Possible values: N/A

Default value: disabled

ip dont-advertise

Configures the Layer 3 Switch to block advertisement of the attached network on the interface. If you do not block advertisement of the network, the Layer 3 Switch will advertise a route to the network containing the host even if the host itself is unavailable.

Use this command when configuring a Layer 3 Switch to assist ServerIrons, third-party SLBs, or web servers with Geographically-distributed SLB. Globally-distributed SLB allows the same web site (and same IP address) to reside on multiple servers, which usually are in geographically dispersed locations. See the "Route Health Injection" chapter of the *Foundry Enterprise Configuration and Management Guide*.

After you enter the **ip dont-advertise** command, the Layer 3 Switch advertises only a host route to the IP address. Thus, if the web site fails the HTTP health check, the Layer 3 Switch removes the static host route for the web site's IP address and also does not advertise a network route for the network containing the IP address.

NOTE: An IP address within the sub-net you want to block must already be configured on the interface.

EXAMPLE:

To block advertisement of a network route for a Class-C host with IP address 209.157.22.1, enter the following commands.

```
BigIron(config-if-1/9) ip address 209.157.22.1/24

BigIron(config-if-1/9) ip dont-advertise 209.157.22.1/24
```

Syntax: [no] ip dont-advertise <ip-addr> <ip-mask>

Or

Syntax: [no] ip dont-advertise <ip-addr>/<mask-bits>

Possible values: see above

Default value: network routes are advertised

ip dvmrp advertise-local

Enables (on) or disables (off) advertisement of a local route on an interface with DVMRP enabled. DVMRP must be enabled on the router for this command to be operational.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron(config)# int e 1/4
BigIron(config-if-1/4)# ip dvmrp advertise-local on
```

Syntax: advertise-local on | off

Possible values: on, off

Default value: off

ip dvmrp metric

Sets the default metric for a directly connected interface, when operating with DVMRP multicast.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron(config)# interface 3/5
BigIron(config-if-3/5)# ip dvmrp metric 10
```

Syntax: ip dvmrp metric <value>

Possible values: 1 – 31 hops

Default value: 1 hop

ip dvmrp ttl-threshold

Specifies how long a packet is considered viable on an interface configured for DVMRP multicast.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

To modify the default TTL value for interface 1 that is configured to operate with DVMRP, enter the following:

```
BigIron(config)# int e 1/4
BigIron(config-if-1/4)# ip dvmrp ttl 60
```

Syntax: ttl-threshold <value>

Possible values: 1 – 254

Default value: 1

ip encapsulation

Enables IP encapsulation and defines the type of encapsulation to be used on a given port.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron(config)# int e 1/6
BigIron(config-if-1/6)# ip dvmrp encap ethernet-2
```

Syntax: ip encapsulation ethernet-2 | snap

Possible values: ethernet-2, snap

Default value: ethernet-2

ip follow

Configures a virtual interface to "follow" the IP address configured on another virtual interface. Thus, you can use this command to conserve your IP address space by configuring multiple virtual interfaces with the same IP address.

EXAMPLE:

To configure an IP sub-net address on virtual interface 1, then configure virtual interfaces 2 and 3 to “follow” the IP sub-net address configured on virtual interface 1, enter the following commands.

```
BigIron(config-vlan-3)# interface ve 1
BigIron(config-vif-1)# ip address 10.0.0.1/24
BigIron(config-vif-1)# interface ve 2
BigIron(config-vif-2)# ip follow ve 1
BigIron(config-vif-2)# interface ve 3
BigIron(config-vif-3)# ip follow ve 1
```

NOTE: Since virtual interfaces 2 and 3 do not have their own IP sub-net addresses but instead are "following" virtual interface 1’s IP address, you still can configure an IPX or AppleTalk interface on virtual interfaces 2 and 3.

Syntax: ip follow ve <num>

Possible values: a configured virtual interface

Default value: N/A

ip follow acl

Enables 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 shared IP address. For example, you can enable virtual routing interface 2 to use its own ACLs instead of using interface 1’s ACLs.

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.

NOTE: This command applies to flow-based ACLs only.

NOTE: This command applies only to virtual routing interfaces configured to follow the IP address of another virtual routing interface. See “ip follow” on page 7-25.

EXAMPLE:

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.

```
BigIron(config)# vlan 1 name primary_vlan
BigIron(config-vlan-1)# untag ethernet 1/1
BigIron(config-vlan-1)# tag ethernet 1/8
BigIron(config-vlan-1)# router-interface ve 1
BigIron(config-vlan-1)# exit
BigIron(config)# interface ve 1
BigIron(config-ve-1)# ip address 10.0.0.1/24
```



```
BigIron(config-ve-1)# ip access-group 1 in
BigIron(config-ve-1)# exit

BigIron(config)# vlan 2 name followerA
BigIron(config-vlan-2)# untag ethernet 1/2
BigIron(config-vlan-2)# tag ethernet 1/8
BigIron(config-vlan-2)# router-interface ve 2
BigIron(config-vlan-2)# exit
BigIron(config)# interface ve 2
BigIron(config-ve-2)# ip follow ve 1
BigIron(config-v2-2)# no ip follow acl
BigIron(config-ve-2)# ip access-group 2 in
BigIron(config-ve-2)# exit

BigIron(config)# vlan 3 name followerB
BigIron(config-vlan-3)# untag ethernet 1/5 to 1/6
BigIron(config-vlan-3)# tag ethernet 1/8
BigIron(config-vlan-3)# router-interface ve 3
BigIron(config-vlan-3)# exit
BigIron(config)# interface ve 3
BigIron(config-ve-3)# ip follow ve 1
BigIron(config-ve-3)# no ip follow acl
BigIron(config-ve-3)# ip access-group 3 out
BigIron(config-ve-3)# exit
```

Syntax: [no] ip follow acl

Possible values: N/A

Default value: Enabled to follow the ACLs of the interface that owns the address

ip fsrp address preference

Modifies the priority for a router interface configured for FSRP operation. The router in the network with the highest value will be the ***active*** (master) router.

FSRP must be active on the router for this command to be operational. FSRP is enabled at the global CONFIG level.

EXAMPLE:

To modify the preference (priority) of a router interface, enter the following command:

```
BigIron(config)# inter e 1/1

BigIron(config-if-1/1)# ip fsrp address 192.33.52.5 preference 200
```

Syntax: ip fsrp address <ip-addr> preference <value>

Possible values: 1 – 255

Default value: 60

ip fsrp address track-port

Assigns a track port for use by the FSRP protocol. The ***track port*** feature is used to track the status of those ports that provide redundant paths. If change in state occurs (up or down), the track port will detect this and the priority of the FSRP Group Interface will be increased or decreased.

FSRP must be active on the router for this command to be operational. FSRP is enabled at the global CONFIG level.

EXAMPLE:

```
BigIron(config)# inter e 2/1

BigIron(config-if-2/1)# ip fsrp address 192.33.52.5 track-port 1
```

Syntax: ip fsrp address <ip-addr> track-port <port>

Possible values: 1 – 26; range is determine by port capacity of the device

Default value: Disabled

ip fsrp address vir-rtr-ip

Defines the virtual router and its address for the specified interface. The virtual router IP address needs to be configured on at least one router in the FSRP group.

NOTE: The virtual router is what arbitrates the redundant path management under the FSRP protocol.

FSRP must be active on the router for this command to be operational. FSRP is enabled at the global CONFIG level.

Syntax: ip fsrp address <ip-addr> vir-rtr-ip <ip-addr>

NOTE: The virtual IP router must belong to the same sub-net and FSRP group as the defined FSRP interface.

EXAMPLE:

```
BigIron(config)# inter e 1/5
```

```
BigIron(config-if-1/5)# ip fsrp add 192.33.52.5 vir-rtr-ip 195.45.5.1
```

Possible values: N/A

Default value: 0.0.0.0

ip fsrp address vir-rtr-ip other-rtr-ip

Defines the partner router interface address. FSRP must be active on the router for this command to be operational.

EXAMPLE:

```
BigIron(config)# inter e 1/3
```

```
BigIron(config-if-1/3)# ip fsrp add 192.33.52.5 vir-rtr-ip 195.45.5.1 other-rtr-ip
195.55.2.1
```

Syntax: ip fsrp address <ip-addr> vir-rtr-ip <ip-addr> other-rtr-ip <ip-addr>

Possible values: N/A

Default value: 0.0.0.0

ip fsrp address keep-alive-time

The **keep-alive-time** parameter allows you to modify how often the FSRP hello message will be sent on a router's interface on which the keep alive time is being configured.

NOTE: The keep-alive-time value must be set to the same value on both the active and standby router when both routers are connected to the same sub-net.

EXAMPLE:

```
BigIron(config)# int 2
```

```
BigIron(config-if-2)# ip fsrp address 192.55.4.3 keep-alive-time 15
```

Syntax: ip fsrp address <ip-addr> keep-alive-time <value>

Possible values: 1 – 120 seconds

Default value: 3 seconds

ip fsrp address router-dead-time

The **router-dead-time** parameter allows you to define the period of time (hold time) that the standby router will wait before determining the active router unavailable (dead). When the configured period of time expires, the standby router will become active.

NOTE: The router-dead-time value must be set to the same value on both the active and standby router when both routers are connected to the same sub-net.

EXAMPLE:

```
BigIron(config)# int 4/2

BigIron(config-if-4/2)# ip fsrp address 192.55.4.3 router-dead-time 30
```

Syntax: ip fsrp address <ip-addr> router-dead-time <value>

Possible values: 3 – 255

Default value: 9 seconds

ip helper-address

Foundry Layer 3 Switches support the relay of UDP/DHCP packets to a destination for a specific application (for example; bootps, domain, tftp), when the destination server is not on the local LAN segment.

To aid in relaying packets to a specific application on a server (for example; bootps, bootpc, domain, TFTP, NetBIOS, time) on a remote network, the router is configured with the destination address of the remote server.

To enter the address of the remote server, enter the commands below. Note that the interface entered is the interface on which the originating host is attached. The value '1' in the example is the identifier of that UDP address. The valid range for identifiers for each interface is 1 – 16.

EXAMPLE:

To support relaying of UDP/DHCP packets to a remote server with an IP address of 207.95.7.6, enter the following:

```
BigIron(config)# interface e 5/2

BigIron(config-if-5/2)# ip helper-address 1 207.95.7.6
```

Syntax: ip helper-address <value> <ip-addr>

Possible values: 1 – 16

Default value: N/A

ip icmp

Causes the interface to drop ICMP packets when excessive numbers are encountered, as is the case when the device is the victim of a Smurf attack. This command allows you to set threshold values for ICMP packets targeted at the device and drop them when the thresholds are exceeded.

EXAMPLE:

You can set threshold values for ICMP packets received on an interface and drop them when the thresholds are exceeded. For example:

```
BigIron(config)# int e 3/11
BigIron(config-if-e100-3/11)# ip icmp burst-normal 5000 burst-max 10000 lockup 300
```

Syntax: ip icmp burst-normal <value> burst-max <value> lockup <seconds>

The burst-normal value can be from 1 – 100000.

The burst-max value can be from 1 – 100000.

The lockup value can be from 1 – 10000.

The number of incoming ICMP packets per second are measured and compared to the threshold values as follows:

- If the number of ICMP packets exceeds the burst-normal value, the excess ICMP packets are dropped.
- If the number of ICMP packets exceeds the burst-max value, all ICMP packets are 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.

Possible values: The burst-normal and burst-max values can be between 1 – 100000 packets. The burst-normal value must be smaller than the burst-max value. The lockup value can be between 1 – 10000 seconds.

Default value: N/A

ip icmp redirects

Disables ICMP redirect messages.

NOTE: The interface forwards misdirected traffic to the appropriate router, even if you disable the redirect messages.

EXAMPLE:

```
BigIron(config)# int e 3/11
BigIron(config-if-e100-3/11)# no ip icmp redirects
```

Syntax: [no] ip icmp redirects

Possible values: N/A

Default value: Redirect messages are enabled

ip igmp static-group

Adds 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.

EXAMPLE:

To manually add a port to a multicast group, enter a command such as the following at the configuration level for the port:

```
BigIron(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:

```
BigIron(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.

Possible values: See above

Default value: N/A

ip irdp

Enables IRDP on an individual interface. You also can change individual IRDP parameters using this command.

EXAMPLE:

```
BigIron(config)# interface ethernet 1/3
BigIron(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 m**inadvertinterval** 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.

Possible values: See above

Default value: Disabled

ip metric

Defines the cost applied to all IP routes by default.

EXAMPLE:

```
BigIron(config)# int e 3/1
BigIron(config-if-3/1)# ip metric 15
```

Syntax: ip metric <value>

Possible values: 1 – 16

Default value: 1

NOTE: RIP considers the metric 16 to be unreachable.

ip mtu

Defines the maximum transmission unit (MTU) for IP packets on a port on an IronCore device.

EXAMPLE:

To change the MTU for an interface to 1000, enter the following:

```
BigIron(config)# int e 4/11
BigIron(config-if-4/11)# ip mtu 1000
```

Syntax: ip mtu <572 – 1492> (Ethernet SNAP); ip mtu <572 – 1500> (Ethernet II);

Possible values: Ethernet type: 572 – 1500; SNAP type: 572 – 1492

Default value: Ethernet type: 1500; SNAP type: 1492

ip multicast boundary

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:

```
BigIron(config)# access-list 1 deny 239.0.0.0 0.255.255.255
BigIron (config)# access-list 1 permit 234.00.0.0 15.255.255.255
BigIron(config)# interface ethernet 1/1
BigIron(config-if-1/1)# ip pim-sparse
BigIron(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.

ip-multicast-disable

Disables transmission and receipt of IGMP queries on a port. Transmission and receipt of IGMP queries is enabled by default once you enable the IP Multicast Traffic Reduction feature.

Disable IGMP on individual ports if you want to block all IP multicast traffic on those ports. When you disable IGMP on an individual port, the Layer 2 Switch does not forward any multicast traffic out the port, but other ports can still send and receive multicast traffic.

NOTE: This command applies only to the IP Multicast Traffic Reduction feature on Layer 2 Switches.

EXAMPLE:

To disable the queries on an interface, enter commands such as the following:

```
FastIron(config)# int e5
FastIron(config-if-5)# ip-multicast-disable
```

To re-enable the queries, enter the following:

```
FastIron(config-if-5)# no ip-multicast-disable
```


Syntax: [no] ip-multicast-disable

Possible values: N/A

Default value: on

ip nat inside

Enables inside NAT on an interface.

EXAMPLE:

To enable inside NAT on an interface, enter commands such as the following:

```
BigIron(config)# interface ethernet 1/1
BigIron(config-if-1/1)# ip nat inside
```

Syntax: [no] ip nat inside

To enable inside NAT on a virtual interface, enter commands such as the following:

```
BigIron(config)# interface virtual 1
BigIron(config-vif-1)# ip nat inside
```

This command enables inside NAT on virtual interface 4.

Possible values: N/A

Default value: Disabled

ip nat outside

Enables outside NAT on the interface attached to public addresses.

EXAMPLE:

To enable outside NAT on an interface, enter commands such as the following:

```
BigIron(config)# interface ethernet 1/2
BigIron(config-if-1/2)# ip nat outside
```

This command enables outside NAT on Ethernet port 1/2.

Syntax: [no] ip nat outside

To enable outside NAT on a virtual interface, enter commands such as the following:

```
BigIron(config)# interface virtual 2
BigIron(config-vif-2)# ip nat outside
```

This command enables outside NAT on virtual interface 4.

Possible values: N/A

Default value: Disabled

ip ospf area

Assigns interfaces to an OSPF area. OSPF must be active on the router and reference the area IP address to which the router will be attached, for this command to be operational. OSPF is enabled at the global CONFIG level.

EXAMPLE:

To assign interface to area ID 192.45.1.0, enter the following commands:

```
BigIron(config)# int e 5/1

BigIron(config-if-5/1)# ip ospf area 192.45.1.0
```

NOTE: Each port supports eight interfaces. All eight interfaces can be assigned to a port at one time using this command.

Syntax: ip ospf area <ip-addr> | <area-number>

Possible values: N/A

Default value: N/A

ip ospf auth-change-wait-time

Changes the authentication-change interval.

After you make a change to OSPF authentication, the software continues to use the old (changed) authentication key for sending packets and accepts packets that contain either the new or the old authentication key. The amount of time during which the software supports both the old and new authentication keys is determined by the authentication-change timer.

The interval applies to all the following types of changes:

- 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

When you make any of the OSPF authentication changes listed above, 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.

EXAMPLE:

To change the authentication-change interval, enter a command such as the following at the interface configuration level of the CLI:

```
BigIron(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.

Possible values: 0 – 14400 seconds

Default: 300 seconds (5 minutes)

ip ospf authentication-key

Assigns a password for managed interface access when operating with OPSF.

OSPF must be active, and the areas to which the router will be attached assigned on the router, for this command to be operational.

EXAMPLE:

To assign an authentication key (password) of 'passkey' for access to interface 1 (module 4), enter the following:

```
BigIron(config)# int e 4/1
```



```
BigIron(config-if-4/1)# ip ospf authentication-key passkey
BigIron(config-if-4/1)# end
BigIron# write memory
```

Syntax: [no] ip ospf authentication-key [0 | 1] <string>

The <string> parameter specifies the password and can be up to eight alphanumeric characters.

The optional **0 | 1** parameter affects encryption. For added security, software release 07.1.10 and later encrypts 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 pasword 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.

Possible values: See above

Default: None.

ip ospf cost

Represents the cost that will be advertised for an interface for various types of service (for example; low delay, high bandwidth (108/ Interface Speed), or link reliability).

Use this command to assign higher or lower costs than the default. This allows you to bias traffic to or from links. The higher the cost on the link, the less desirable the path.

EXAMPLE:

To assign a cost of 10 to interface 8 (module 2), enter the following:

```
BigIron(config)# int e 2/8
BigIron(config-if-2/8)# ip ospf cost 10
```

Syntax: ip ospf cost <num>

Possible values: 1 – 65,535

Default: 1 for 100 or 1000 Mbps links; 10 for 10Mbps links

ip ospf database-filter

Blocks flooding of outbound OSPF LSAs on the interface.

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.

EXAMPLE:

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.

```
BigIron(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:

```
BigIron(config-if-1/1)# no ip ospf database-filter all out
```

Possible values: see above

Default: Outbound LSAs are not blocked

ip ospf dead-interval

Defines the number of seconds that a neighbor OSPF router will wait for receipt of a hello packet, before declaring the router down.

EXAMPLE:

To change the dead interval time for interface 5 (module 3) from the default of 40 seconds, enter the following:

```
BigIron(config)# int e 3/5
BigIron(config-if-3/5)# ip ospf dead-interval
```

Syntax: ip ospf dead-interval <value>

Possible values: 1 – 65,535 seconds.

Default: 40 seconds

ip ospf hello-interval

Defines the length of time between the transmission of OSPF hello packets.

EXAMPLE:

To change the hello interval for interface 5 to 20 seconds from the default value of 10 seconds, enter the following:

```
BigIron(config)# int e 3/5
BigIron(config-if-3/5)# ip ospf hello-interval 20
```

Syntax: ip ospf hello-interval <value>

Possible values: 1 – 65,535 seconds

Default: 10 seconds

ip ospf md5-authentication

Configures MD5 authentication parameters for OSPF. You can configure the following parameters using this command:

- MD5 key-activation wait time – specifies how many seconds the Layer 3 Switch waits before 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).
- Key ID and key string – specifies the MD5 key ID and the string value for the key. The key ID can be from 1 – 255. The string can be up to 16 alphanumeric characters long. The Layer 3 Switch encrypts the key in each OSPF packet sent on this interface.

EXAMPLE:

To change the key activation wait time from 300 seconds to 45 seconds, enter the following:

```
BigIron(config)# int e 2/5
BigIron(config-if-2/5)# ip ospf md5-authentication key-activation-wait-time 30
```

EXAMPLE:

To configure key ID 35 with the string value “UR2crusty”, enter the following:

```
BigIron(config)# int e 2/5
BigIron(config-if-2/5)# ip ospf md5-authentication key-id 35 key UR2crusty
```

Syntax: [no] ip ospf md5-authentication key-activation-wait-time <num> | key-id <num> [0 | 1] key <string>

The **key-activation-wait-time** <num> parameter can be a value from 0 – 14400 seconds. The default is 300 seconds (5 minutes).

The **key-id** <num> specifies the key and can be a value from 1 – 255.

The **key** <string> parameter specifies the authentication string and can be up to 16 alphanumeric characters long.

The optional **0 | 1** parameter affects encryption. For added security, software release 07.1.10 and later encrypts 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 pasword 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.

Possible values: See above

Default: See above

ip ospf network

Configures an OSPF interface on an ATM subinterface.

EXAMPLE:

To configure an IS-IS interface on an ATM point-to-multipoint interface, enter commands such as the following:

```
BigIron(config)# interface atm 1/2.10 multipoint
BigIron(config-subif-1/2.10)# atm pvc 1 1 ubr ip 10.10.10.1
BigIron(config-subif-1/2.10)# atm pvc 1 2 ubr ip 10.10.10.2
BigIron(config-subif-1/2.10)# atm pvc 1 3 ubr ip 10.10.10.3
BigIron(config-subif-1/2.10)# ip address 10.10.10.9/28
BigIron(config-subif-1/2.10)# ip ospf area 0
BigIron(config-subif-1/2.10)# disable
BigIron(config-subif-1/2.10)# ip ospf network point-to-multipoint
BigIron(config-subif-1/2.10)# enable
```

The first four commands configure a point-to-multipoint ATM subinterface with three PVCs. The last five commands configure an IP address on the subinterface, disable the interface, enable OSPF on the subinterface, configure the OSPF network type for the interface to point-to-multipoint, then re-enable the interface. You must disable the interface to change the interface type.

Syntax: [no] ip ospf network [point-to-multipoint]

The default OSPF network type is point-to-point.

NOTE: You must enter the **ip ospf network point-to-multipoint** command before the interface forms OSPF adjacencies. The command is not valid once an adjacency is formed.

EXAMPLE:

To configure an IS-IS interface on an ATM point-to-point interface, enter commands such as the following:

```
BigIron(config)# interface atm 1/2.10
BigIron(config-subif-1/2.10)# atm pvc 3 34 ubr
BigIron(config-subif-1/2.10)# ip address 10.10.10.9/28
BigIron(config-subif-1/2.10)# ip ospf area 0
BigIron(config-subif-1/2.10)# disable
BigIron(config-subif-1/2.10)# ip ospf network
BigIron(config-subif-1/2.10)# enable
```

Possible values: See above

Default: None configured

ip ospf passive

Configures an OSPF network interface to be 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.

NOTE: This command 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 “ip address” on page 7-21.

EXAMPLE:

```
BigIron(config)# int eth 1/1
BigIron(config-if-1/1)# ip ospf passive
```

Syntax: ip ospf passive

Possible values: N/A

Default: N/A

ip ospf priority

Indicates the priority of an interface. Priority is used to select the designated router. The higher the number, the greater the priority. In the case of equal priority, the router with the highest IP Address will be the designated router. A value of zero signifies that the router is not eligible to become the designated router on its network.

EXAMPLE:

To modify the priority of interface 5 (module 2) to 15 from the default of 1, enter the following:

```
BigIron(config)# int e 2/5
BigIron(config-if-2/5)# ip ospf priority 15
```

Syntax: ip ospf priority <value>

Possible values: 0 – 255. If you set the priority to 0, the Layer 3 Switch does not participate in DR and BDR election.

Default: 1

ip ospf retransmit-interval

Defines the time between retransmits of link state advertisements to router adjacencies for an interface.

EXAMPLE:

To modify the retransmit interval of interface 5 (module 2) to 15 from the default of 5 seconds, enter the following:

```
BigIron(config)# int e 2/5
BigIron(config-if-2/5)# ip ospf retransmit-interval 15
```

Syntax: ip ospf retransmit-interval <value>

Possible values: 0 – 3600 seconds

Default: 5 seconds

ip ospf transmit-delay

Indicates the time it takes to transmit Link State Update packets on an interface.

EXAMPLE:

To modify the transit delay of interface 5 (module 2) to 10, from the default of 1 second, enter the following:

```
BigIron(config)# int e 2/5
BigIron(config-if-2/5)# ip ospf transmit-delay 10
```

Syntax: ip ospf transmit-delay <value>

Possible values: 0 – 3600 seconds

Default: 1 second

ip pim

Enables IP PIM DM on the interface. You can enable PIM DM version 1 or version 2. 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 the 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 in software release 07.2.05 and higher. Previous releases support only version 1. 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 software release 07.2.05 or higher and also 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).

EXAMPLE:

To enable PIM DM V2 globally and on an interface, enter commands such as the following:

```
BigIron(config)# router pim
BigIron(config-pim-router)# interface ethernet 1/1
BigIron(config-if-1/1)# ip pim
```

The commands in this example globally enable PIM DM, then enable PIM DM V2 on interface 1/1. Since the default version is 2, you do not need to specify the version.

Syntax: [no] ip pim [version 1 | 2]

The **version 1 | 2** parameter specifies the PIM DM version. The default version is 2.

To enable PIM version 1 on interface, enter the following command at the configuration level for the interface:

```
BigIron(config-if-1/1)# ip pim version 1
```

If you have enabled PIM version 1 but need to enable version 2 instead, enter the following command at the configuration level for the interface:

```
BigIron(config-if-1/1)# ip pim version 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:

```
BigIron(config-if-1/1)# ip pim version 2
```

```
BigIron(config-if-1/1)# no ip pim version 1
```

To disable PIM DM on the interface, enter the following command:

```
BigIron(config-if-1/1)# no ip pim
```

Possible values: version 1 or **version 2**

Default: Version 2, when PIM DM is enabled

ip pim-sparse

Enables PIM Sparse on an interface. 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 this command.

EXAMPLE:

To enable PIM Sparse mode on an interface, enter commands such as the following:

```
BigIron(config)# interface ethernet 2/2
BigIron(config-if-2/2)# ip address 207.95.7.1 255.255.255.0
BigIron(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:


```
BigIron(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.

Possible values: N/A

Default: Disabled

ip pim ttl

Specifies 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.

EXAMPLE:

To configure a TTL of 45, enter the following:

```
BigIron(config-if-3/24)# ip pim ttl 45
```

Syntax: ip pim ttl <1-64>

Possible values: 1 – 64

Default: 1

ip-policy

Locally applies a Layer 4 QoS policy to a port on a Layer 2 Switch. If you use the **local** parameter instead of the **global** parameter when you configure the policy using the **ip policy** command, you need to use the **ip-policy** command at the Interface level to apply the policy to a specific port.

NOTE: You must use the **ip access policy** command to configure the policy before using the **ip-policy** command. See “ip access-policy” on page 6-39.

EXAMPLE:

To configure a Layer 4 QoS policy to give the highest priority to HTTP packets on ports 1/1 and 1/2 on a FastIron II, enter the following commands:

```
FastIronII(config)# ip access policy 1 priority 7 tcp http local
FastIronII(config)# int e 1/1
FastIronII(config-if-1/1)# ip-policy 1
FastIronII(config-if-1/1)# int e 1/2
FastIronII(config-if-1/2)# ip-policy 1
```

EXAMPLE:

To configure a Layer 4 QoS policy to give high priority to DNS packets on ports 5 and 7 on a Stackable Layer 2 Switch, enter the following commands:

```
FastIron(config)# ip access policy 1 high tcp dns local
FastIron(config)# int e 5
FastIron(config-if-5)# ip-policy 1
FastIron(config-if-5)# int e 7
FastIron(config-if-7)# ip-policy 1
```

Syntax: ip-policy <num>

Possible values: See above

Default value: N/A

ip policy frag-match-dst

Enables PBR for fragmented packets, based on the destination TCP or UDP port of the fragments.

By default, PBR policies apply at Layer 3 only. The device matches traffic against the Layer 3 information in a PBR policy's ACLs, and applies the policy if the traffic matches the ACL. The device does not apply a PBR policy to a packet fragment even if the fragment's IP addresses match an ACL in the policy. Instead, the device forwards the fragment using a non-PBR route. This is true even if an ACL in a PBR policy contains Layer 4 information.

NOTE: This command applies only to JetCore devices running software release 07.6.01 or later.

EXAMPLE:

```
BigIron(config)# access-list 111 permit udp any host 2.3.3.5 eq 2049
BigIron(config)# route-map slbmap permit 1
BigIron(config-routemap slbmap)# match ip address 111
BigIron(config-routemap slbmap)# set next-hop 1.2.3.4
BigIron(config-routemap slbmap)# exit
BigIron(config)# interface ethernet 1/1
BigIron(config-if-1/1)# ip policy route-map slbmap
BigIron(config-if-1/1)# ip policy frag-match-dest
```

Syntax: [no] ip policy frag-match-dest

Possible values: N/A

Default value: The PBR policy is not applied to the fragment.

ip policy frag-match-src

Enables PBR for fragmented packets, based on the source TCP or UDP port of the fragments.

By default, PBR policies apply at Layer 3 only. The device matches traffic against the Layer 3 information in a PBR policy's ACLs, and applies the policy if the traffic matches the ACL. The device does not apply a PBR policy to a packet fragment even if the fragment's IP addresses match an ACL in the policy. Instead, the device forwards the fragment using a non-PBR route. This is true even if an ACL in a PBR policy contains Layer 4 information.

NOTE: This command applies only to JetCore devices running software release 07.6.01 or later.

EXAMPLE:

```
BigIron(config-if-1/1)# ip policy frag-match-src
```

Syntax: [no] ip policy frag-match-src

Possible values: N/A

Default value: The PBR policy is not applied to the fragment.

ip policy frag-match-src-dest

Enables PBR for fragmented packets, based on both the source and destination TCP or UDP ports of the fragments.

By default, PBR policies apply at Layer 3 only. The device matches traffic against the Layer 3 information in a PBR policy's ACLs, and applies the policy if the traffic matches the ACL. The device does not apply a PBR policy to a packet fragment even if the fragment's IP addresses match an ACL in the policy. Instead, the device forwards the fragment using a non-PBR route. This is true even if an ACL in a PBR policy contains Layer 4 information.

NOTE: This command applies only to JetCore devices running software release 07.6.01 or later.

EXAMPLE:

```
BigIron(config-if-1/1)# ip policy frag-match-src-dest
```

Syntax: [no] ip policy frag-match-src-dest

Possible values: N/A

Default value: The PBR policy is not applied to the fragment.

ip policy route-map

Enables Policy-Based Routing (PBR) on the interface.

EXAMPLE:

To enable PBR locally, enter commands such as the following:

```
BigIron(config)# interface ve 1
BigIron(config-vif-1)# ip policy route-map source-route
```

The commands in this example change the CLI to the Interface level for virtual interface 1, then apply the "source-route" route map to the interface. You can apply a PBR route map to Ethernet ports or virtual interfaces.

Syntax: [no] ip policy route-map <map-name>

Possible values: the name of a configured route map

Default value: N/A

ip redirect

Disables or re-enables ICMP redirects on the interface.

EXAMPLE:

To disable ICMP redirects on a specific interface, enter the same command at the configuration level for the interface:

```
BigIron(config)# int e 3/11
BigIron(config-if-e100-3/11)# no ip redirect
```

Syntax: [no] ip redirect

Possible values: N/A

Default value: Enabled

ip rip

Sets the RIP type on all interfaces that will route RIP. The following RIP types are supported:

- RIP version 1 only
- RIP version 2 only
- RIP v1-compatible-v2

RIP must be active and the redistribution table set (using the **permit** and **deny** commands) for this command to be operational. RIP is enabled at the Global CONFIG Level.

EXAMPLE:

To modify the RIP type for interface 1 (module 4), to version 1 only, enter the following:

```
BigIron(config)# int e 4/1
BigIron(config-if-4/1)# ip rip v1-only
BigIron(config-if-4/1)# end
BigIron# write memory
```

Syntax: ip rip v1-only | v1-compatible-v2 | v2-only

Possible values: v1-only, v1-compatible-v2, v2-only

Default value: v2-only

ip rip filter-group

Allows a group of RIP filters to be applied to an IP interface. The filter can be applied to either incoming or outgoing traffic.

EXAMPLE:

To apply filters to an individual interface basis (for example, interface 2/2), enter the following:

```
BigIron(config)# int e 2/2
BigIron(config-if-2/2)# ip rip filter-group in 1 2 3 10
```

Syntax: ip rip filter-group in | out <index>

Possible values: in or out, defined filter indices

Default value: disabled

ip rip learn-default

This feature allows a Layer 3 Switch to learn and advertise default RIP routes. This command can be applied on a global or interface basis. This example shows the feature enabled at the interface level.

EXAMPLE:

```
BigIron(config)# int e 2/2
BigIron(config-if-2/2)# ip rip learn-default
```

Syntax: ip rip learn-default

Possible values: N/A

Default value: N/A

ip rip poison-reverse

Enables poison-reverse on the RIP routing protocol to prevent routing loops and slow convergence within the network.

For this command to be operational, RIP must be enabled and active on the router, and the RIP type configured.

EXAMPLE:

```
BigIron(config)# int e 4/1
BigIron(config-if-4/1)# ip rip poison-reverse
```

Syntax: ip rip poison-reverse

Possible values: N/A

Default value: enabled

ip route-cache flow

Enables Flow Switching on an interface.

NetFlow or sFlow exports flow data only for the interfaces on which you enable Flow Switching. You can enable Flow Switching on the following types of interfaces:

- Ethernet interfaces
- Non-NPA POS interfaces
- Virtual routing interfaces (flows are collected and exported for all the ports in the VLAN on which the virtual interface is configured)

EXAMPLE:

```
BigIron(config)# interface ethernet 1/1
BigIron(config-if-1/1)# ip route-cache flow
```



```
BigIron(config-if-1/1)# exit
```

Syntax: [no] ip route-cache flow

Possible values: N/A

Default value: disabled

ip router isis

Enables IS-IS on an interface.

NOTE: You must enable the protocol globally first, then on individual interfaces. To enable IS-IS globally, see "router isis" on page 6-108.

EXAMPLE:

```
NetIron(config)# interface ethernet 1/1
NetIron(config-if-1/1)# ip router isis
NetIron(config-if-1/1)# interface ethernet 1/2
NetIron(config-if-1/2)# ip router isis
```

These commands enable IS-IS on ports 1/1 and 1/2. The NET configured above (at the IS-IS configuration level) applies to both interfaces.

NOTE: If you have not configured a NET, the software displays the message "ISIS: Please configure NET!" and changes the CLI to the IS-IS configuration level.

Syntax: [no] ip router isis

Possible values: See above

Default value: Disabled

ip tcp burst-normal

Configures the interface to protect itself against TCP SYN attacks by dropping TCP SYN packets when excessive numbers are encountered.

EXAMPLE:

To set threshold values for TCP SYN packets received on interface 3/11:

```
BigIron(config)# int e 3/11
BigIron(config-if-e100-3/11)# ip tcp burst-normal 10 burst-max 100 lockup 300
```

Syntax: ip tcp burst-normal <value> burst-max <value> lockup <seconds>

The **burst-normal** value can be from 1 – 100000.

The **burst-max** value can be from 1 – 100000.

The **lockup** value can be from 1 – 10000.

The number of incoming TCP SYN packets per second are measured and compared to the threshold values as follows:

- If the number of TCP SYN packets exceeds the **burst-normal** value, the excess TCP SYN packets are dropped.
- If the number of TCP SYN packets exceeds the **burst-max** value, *all* TCP SYN packets are 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.

In the example above, if the number of TCP SYN packets received per second exceeds 10, the excess packets are dropped. If the number of TCP SYN packets received per second exceeds 100, the device drops all TCP SYN packets for the next 300 seconds (five minutes).

Possible values: see above

Default value: no threshold

ip tunnel

This is a launch command that moves activity to the IP tunnel CONFIG level. It allows you to define an IP tunnel for a specific interface. The requested IP address is the end point of the tunnel (the destination interface). IP tunnels must be defined for multicast traffic that will be passing through routers that are not IP-multicast capable.

For this command to be operational, an IP multicast protocol must be enabled on the router. IP tunneling must also be enabled and defined on the destination router interface.

NOTE: IP tunnels are supported for DVMRP only in software release 07.6.01 and later.

EXAMPLE:

```
BigIron(config)# inter e 3/1

BigIron(config-if-3/1)# ip address 192.33.65.4/24

BigIron(config-if-3/1)# ip tunnel 209.157.22.26 pim

BigIron(config-if-pim-tunnel)# end

BigIron# write memory
```

Syntax: ip tunnel <ip-addr> pim

Possible values: valid ip address

Default value: N/A

ip use-acl-on-arp

Enables you to use ACLs to filter ARP packets.

EXAMPLE:

To implement the ACL ARP filtering feature, enter command such as the following:

```
BigIron(config-vlan-4)# interface ve 4
BigIron(config-ve-4)# ip follow ve 2
BigIron(config-ve-4)# ip use-acl-on-arp 103
BigIron(config-ve-4)# exit
```

Syntax: [no] ip use-acl-on-arp [<access-list-number>]

When this command is enabled, the ARP module checks the source IP address of the ARP request packets received on the interface. It then applies the specified ACL policies to the packet. Only the packet with the IP address that the ACL permits will be allowed to be to be written in the ARP table; those that are not permitted will be dropped.

Possible values:

The <access-list-number> parameter identifies the ID of the standard ACL that will be used to filter the packet. Only the source and destination IP addresses will be used to filter ARP packet. You can do one of the following for <access-list-number>:

- Enter an ACL ID to explicitly specify the ACL to be used for filtering. In the example above, the line BigIron `(config-ve-2)# ip use-acl-on-arp 103` specifies ACL 103 to be used as the filter.
- Allow the ACL ID to be inherited from the IP ACLs that have been defined for the device. In the example above, the line BigIron `(config-ve-3)# ip use-acl-on-arp` does not define an ACL, but allows the ACL to be inherited from the IP ACL 102. Also in the example, the line BigIron `(config-ve-4)# ip use-acl-on-arp` allows the ACL to be inherited from IP ACL 101 because of the ip follow relationship between virtual routing interface 2 and virtual routing interface 4. Virtual routing interface 2 is configured with IP IP ACL 101; thus virtual routing interface 4 inherits IP ACL 101.

ARP requests will not be filtered by ACLs if one of the following conditions occur:

- If the ACL is to be inherited from an IP ACL, but there is no IP ACL defined.
- An ACL ID is specified for the **use-acl-on-arp** command, but no IP address or “any any” filtering criteria have been defined under the ACL ID.

Default value: N/A

ip vrrp

Lets you configure a Virtual Router Redundancy Protocol (VRRP) virtual router on an interface.

EXAMPLE:

```
BigIron(config)# int e 4/1
BigIron(config-if-4/1)# ip vrrp 1
```

Syntax: ip vrrp vrid <vrid>

Possible values: VRID is a virtual router ID.

Default value: N/A

ip vrrp auth-type

Configures the authentication type for a virtual router interface.

EXAMPLE:

```
BigIron(config)# int e 4/1
BigIron(config-if-4/1)# ip vrrp auth-type simple-text-auth pword
```

Syntax: ip vrrp auth-type no-auth | simple-text-auth <auth-data>

Possible values: <auth-data> is a simple text password.

Default value: N/A

ip vrrp-extended

Lets you configure a VRRPE virtual router on an interface.

EXAMPLE:

```
BigIron(config)# int e 4/1
BigIron(config-if-4/1)# ip vrrp-extended 1
```

Syntax: ip vrrp vrid-extended <vrid>

Possible values: VRID is a virtual router ID.

Default value: N/A

ip vrrp-extended auth-type

Configures the authentication type for a virtual router interface.

EXAMPLE:

```
BigIron(config)# int e 4/1
BigIron(config-if-4/1)# ip vrrp-extended auth-type simple-text-auth pword
```

Syntax: [no] ip vrrp-extended auth-type no-auth | simple-text-auth <auth-data>

Possible values: <auth-data> is a simple text password.

Default value: N/A

ip vsrp auth-type

Configures a simple password for authentication on a VSRP interface.

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.

EXAMPLE:

```
BigIron(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.

Possible values: See above

Default value: No authentication

ipg10

Allows you to modify the inter-packet gap (delay) between packets on a 10Mbps Ethernet segment. By default, the delay between packets will be 12 bytes or 9.6 microseconds.

Use this command only to adjust the inter-packet gap to match older adapters that do not meet the default IPG requirements for Ethernet.

In determining the value to enter in the CLI command, note that one byte equals.8 microseconds for packets on a 10Mbps segment, so the following equation can be used:

IPG10 = 9.6 microseconds + (value *.8)

where value is the number of bytes by which you want to increase the inter-packet gap.

EXAMPLE:

To increase the delay between packets by 3.2 microseconds, enter the port to be modified and then enter the value of 4 (4*.8 =3.2 microseconds):

```
BigIron(config)# int e 4/4
BigIron(config-if-4/4)# ipg10 4
```

Syntax: ipg10 <value>

Possible values: 0 – 100 bytes

Default value: 12 bytes or ipg10 0

NOTE: Entering the value of 0 with the **ipg10**, **ipg100**, and **ipg1000** commands restores the inter-packet gap (IPG) to the default of 12 bytes.

ipg100

Allows you to modify the inter-packet gap (delay) between packets on a 100Mbps Ethernet segment on a port-by-port basis. By default, the delay between packets will be 12 bytes or 0.96 microseconds.

Use this command only to adjust the inter-packet gap to match that of older adapters that do not meet the default IPG requirements for Fast Ethernet.

In determining the value to enter in the CLI command, note that one byte equals.08 microseconds for packets on a 100Mbps segment, so the following equation can be used:

IPG100 = 0.96 microseconds + (value *.08)

where value is the number of bytes by which you want to increase the inter-packet gap.

EXAMPLE:

To increase the delay between packets by 3.2 microseconds, enter the port to be modified and then enter the value of 40 (40*.08 =3.2 microseconds):

```
BigIron(config)# int e 3/4
BigIron(config-if-3/4)# ipg100 40
```

Syntax: ipg100 <value>

Possible values: 0 – 100

Default value: 12 bytes or ipg100 0

NOTE: Entering the value of 0 with the **ipg10**, **ipg100**, and **ipg1000** commands restores the inter-packet gap (IPG) to the default of 12 bytes.

ipg1000

Allows you to modify the inter-packet gap (delay) between packets on a 1000Mbps Gigabit Ethernet segment on a port-by-port basis. By default, the delay between packets will be 12 bytes or.096 microseconds.

Use this command only to adjust the inter-packet gap to match that of older adapters that do not meet the default IPG requirements for Gigabit Ethernet.

In determining the value to enter in the CLI command, note that one byte equals.008 microseconds for packets on a 1000Mbps segment, so the following equation can be used:

IPG1000 =.096 microseconds + (value *.008)

where value is the number of bytes by which you want to increase the inter-packet gap.

EXAMPLE:

To increase the delay between packets by.32 microseconds, first enter the port to be modified and then enter the value of 40 (40*.008 =.32 microseconds):

```
BigIron(config)# int e 3/4
BigIron(config-if-3/4)# ipg1000 40
```

Syntax: ipg1000 <value>

Possible values: 1 – 100

Default value: 12 bytes or ipg1000 0

NOTE: Entering the value of 0 with the **ipg10**, **ipg100**, and **ipg1000** commands restores the inter-packet gap (IPG) to the default of 12 bytes.

ipx forward-filter-group

Allows a group of defined forward filters to be applied to an IPX interface. The filter can be applied to either **in**coming or **out**going traffic.

Prior to using this command, you must first enable IPX on the router using the **router ipx** command.

EXAMPLE:

```
BigIron(config)# int e 4/1
BigIron(config-if-4/1)# ipx forward-filter-group in 2 3 5
```

Syntax: ipx forward-filter-group in | out <index>

Possible values: in or out, defined filter indexes

Default value: N/A

ipx gns-reply-disable

Disables GNS replies on individual Layer 3 Switch ports.

EXAMPLE:

To disable IPX GNS replies for all IPX interfaces on port 1/1:

```
BigIron(config)# int eth 1/1
BigIron(config-if-1/1)# ipx gns-reply-disable
```

Syntax: [no] ipx gns-reply-disable

Possible values: When IPX is enabled in the Layer 3 Switch, the device responds to all GNS requests by default.

Default value: N/A

ipx netbios-allow

Enables NetBIOS broadcasts (type 20) to be routed over IPX. IPX must be enabled on the router and a network number and frame type defined for each IPX interface.

EXAMPLE:

To enable NetBIOS on an interface (for example, module 5 on port 2), enter the following:

```
BigIron(config)# int e 5/2
BigIron(config-if-5/2)# ipx netbios-allow
```

Syntax: ipx netbios-allow

Possible values: N/A

Default value: disabled

ipx network

Assigns network numbers and frame types for each IPX interface. This is the second step in using IPX on the router. Prior to using this command, enable IPX on the router using the **router ipx** command.

EXAMPLE:

```
BigIron(config)# int e 3/2
BigIron(config-if-3/2)# ipx network 11110055 ethernet_snap
```

NOTE: Other frame types are supported:

ethernet_802.2

ethernet_802.3

ethernet_ii

Syntax: ipx network <network-number> <frame-type> [netbios-allow | netbios-disallow]

Possible values: see above

Default value: NetBIOS allowed

ipx output-gns-filter

Applies IPX access lists for filtering GNS replies to an interface.

EXAMPLE:

To apply previously defined IPX ACLs 10, 20, and 32 to port 2/2 to control responses to GNS requests on that port:


```
BigIron(config)# int e 2/2
BigIron(config-if-2/2)# ipx output-gns-filter 10 20 32
```

Syntax: [no] ipx output-gns-filter <num> [<num>…]

Possible values: Defined filter indices

Default value: N/A

ipx rip-filter-group

Allows a group of RIP filters to be applied to an IPX interface. The filter can be applied to either incoming or outgoing traffic.

EXAMPLE:

To apply filters to an individual interface basis (for example, interface 2/2), enter the following:

```
BigIron(config)# int e 2/2
BigIron(config-if-2/2)# ipx rip-filter-group in 1 2 3 10
```

Syntax: ipx rip-filter-group in | out <index>

Possible values: in or out, defined filter indices

Default value: disabled

ipx rip-max-packetsize

Changes the maximum size of IPX RIP update packets sent by the router.

EXAMPLE:

To change the maximum packet size of IPX RIP advertisements sent on interface 1/1 from the default 432 bytes to 832 bytes, enter the following command. This command increases the number of IPX RIP routes an advertisement packet holds from 50 to 100.

```
BigIron(config) int e 1/1
BigIron(config-if-1/1) ipx rip-max-packetsize 832
```

Syntax: ipx rip-max-packetsize <bytes>

The number of bytes can be from 40 bytes (enough for one route) – 1488 bytes (enough for 182 routes). The default is 432 bytes.

Possible values: 40 – 1488 bytes

Default value: 432

ipx rip-multiplier

Changes the age time for learned IPX routes. The software calculates the age time by multiplying the advertisement interval times the age timer. For example, the default age time for IPX routes is 180 seconds, which is 60 (the default advertisement interval) multiplied by 3 (the default age timer).

EXAMPLE:

To change the age timer for IPX routes from 3 to 4 on interface 1/1, enter the following commands.

```
BigIron(config) int e 1/1
BigIron(config-if-1/1) ipx rip-multiplier 4
```

Syntax: ipx rip-multiplier <num>

The <num> parameter specifies the age time and can be from 1 – 65535. The default is 3.

Possible values: 1 – 65535

Default value: 3

ipx sap-filter-group

Allows a group of defined IPX/SAP filters to be applied to IPX interfaces. The filters can be applied to either incoming or outgoing traffic.

EXAMPLE:

To apply filters to an individual interface's inbound IPX filter group, enter commands such as the following:

```
BigIron(config)# int e 3/2
BigIron(config-if-3/2)# ipx sap-filter-group in 2 3 5
```

Syntax: ipx sap-filter-group in | out <index>

Possible values: in or out, defined filter indexes

Default value: N/A

ipx sap-interval

Changes how often the Layer 3 Switch sends IPX SAP updates to neighboring IPX routers.

EXAMPLE:

To change the advertisement interval for IPX SAP advertisements sent on interface 1/1 from 60 seconds to 120 seconds, enter the following commands:

```
BigIron(config) int e 1/1
BigIron(config-if-1/1) ipx sap-interval 120
```

Syntax: ipx sap-interval <interval>

The <interval> can be from 10 – 65535 seconds. The default is 60.

Possible values: 10 – 65535

Default value: 60

ipx sap-max-packetsize

Changes the maximum size of IPX SAP update packets sent by the router.

EXAMPLE:

To change the maximum number of bytes in IPX SAP advertisements sent on interface 5/1 from 480 to 672 (enough for 10 servers plus the 32 bytes of packet header), enter the following commands:

```
BigIron(config) int e 5/1
BigIron(config-if-5/1) ipx sap-max-packetsize 672
```

Syntax: ipx sap-max-packetsize <bytes>

The number of bytes can be from 96 bytes (enough for one server) – 1440 bytes (enough for 22 servers). The default is 480 bytes.

Possible values: 96 – 1440 bytes

Default value: 480

ipx sap-multiplier

Changes the age time for learned IPX SAP entries. The software calculates the age time by multiplying the advertisement interval times the age timer. For example, the default age time for IPX SAP entries is 180 seconds, which is 60 (the default advertisement interval) multiplied by 3 (the default age timer).

EXAMPLE:

To change the age timer for IPX servers from 3 to 2 on interface 5/1, enter the following commands.

```
BigIron(config) int e 5/1
BigIron(config-if-5/1) ipx sap-multiplier 2
```

Syntax: ipx sap-multiplier <num>

The <num> parameter specifies the age time and can be from 1 – 65535. The default is 3.

Possible values: 1 – 65535

Default value: 3

ipx update-time

Changes how often the Layer 3 Switch sends IPX RIP updates to neighboring IPX routers.

EXAMPLE:

To change the advertisement interval for IPX RIP advertisements sent on interface 1/1 from 60 seconds to 30 seconds, enter the following commands:

```
BigIron(config)# int e 1/1
BigIron(config-if-1/1)# ipx update-time 30
```

Syntax: ipx update-time <interval>

The <interval> can be from 10 – 65535 seconds. The default is 60.

Possible values: 10 – 65535

Default value: 60

isis circuit-type

Changes the IS-IS type (Level-1 or Level-2) supported on an interface.

NOTE: To change the type globally, see “is-type” on page 12-5.

EXAMPLE:

```
NetIron(config-isis-router)# interface ethernet 1/1
NetIron(config-if-1/1)# isis circuit-type level-1
```

These commands change the CLI to the interface configuration level, then change the IS-IS type supported for the IS-IS circuit on that interface.

Syntax: [no] isis circuit-type level-1 | level-1-2 | level-2

Possible values: See above

Default value: Both levels supported

isis metric

Changes the IS-IS metric on an interface.

When the Layer 3 Switch originates an IS-IS route or calculates a route, the Layer 3 Switch adds a metric (cost) to the route. Each IS-IS interface has a separate metric value.

The Layer 3 Switch applies the interface-level metric to routes originated on the interface and also when calculating routes. The Layer 3 Switch does not apply the metric to link-state information that the Layer 3 Switch receives from one IS and floods to other ISs.

NOTE: If the metric value you want to use is higher than 63 but you have not changed the metric style to wide, change the metric style first, then set the metric. To change the metric style, see “metric-style” on page 12-7. The IS-IS neighbors that will receive the advertisements also must be enabled to receive wide metrics.

EXAMPLE:

```
NetIron(config-isis-router)# interface ethernet 1/1
NetIron(config-if-1/1)# isis metric
```

Syntax: [no] isis metric <num>

The <num> parameter specifies the metric. You can change the metric on an individual interface to a value in one of the following ranges:

- 1 – 63 for the narrow metric style (the default metric style)
- 1 – 16777215 for the wide metric style

The default interface metric is 10.

Possible values: See above

Default value: 10

isis passive

Disables IS-IS adjacency formation on an interface.

When you enable IS-IS on any type of interface except a loopback interface, the interface also is enabled to send advertisements and form an adjacency with an IS at the other end of the link by default. Adjacency formation and advertisements are disabled by default on loopback interfaces.

You can enable or disable adjacency formation and advertisements on an interface.

NOTE: The Foundry device advertises an IS-IS interface to its area regardless of whether adjacency formation is enabled.

EXAMPLE:

```
NetIron(config-isis-router)# interface ethernet 1/1
NetIron(config-if-1/1)# isis passive
```

This command disables IS-IS adjacency formation on port 1/1. The device still advertises this IS-IS interface into the area, but does not allow the port to form an adjacency with the IS at the other end of the link.

Syntax: [no] isis passive

Possible values: See above

Default value: Adjacency formation is enabled

isis password

Configure an IS-IS interface password.

EXAMPLE:

To configure an IS-IS password on an interface, enter commands such as the following:

```
NetIron(config)# interface ethernet 1/1
NetIron(config-if-1/1)# isis password int1
```

This command configures the Foundry device to use the password “int1” to authenticate Hello PDUs sent or received on port 1/1.

Syntax: [no] isis password <string>

The <string> parameter specifies the password. You can enter an alphanumeric string up to 80 characters long. The password can contain blank spaces. If you use a blank space in the password, you must use quotation marks (“ “) around the entire password; for example, **isis password “interface 1”**.

Possible values: See above

Default value: None configured

isis priority

Sets the IS-IS priority on an interface.

The priority of an IS-IS interface determines the priority of the interface for being elected as a Designated IS. Level-1 has a Designated IS and Level-2 has a Designated IS. The Level-1 and Level-2 Designated ISs are independent, although the same device can become both the Level-1 Designated IS and the Level-2 Designated IS.

By default, the Level-1 and Level-2 priority is 64. You can configure an interface's priority to a value from 0 – 255. You can configure the same priority for both Level-1 and Level-2 or you can configure a different priority for each level. In case of a tie (if two or more devices have the highest priority within a given level), the device with the highest MAC address becomes the Designated IS for that level.

NOTE: You can set the IS-IS priority on an individual interface basis only. You cannot set the priority globally.

EXAMPLE:

To set the IS-IS priority on an interface, enter commands such as the following:

```
NetIron(config-isis-router)# interface ethernet 1/1
NetIron(config-if-1/1)# isis priority 128
```

This command sets the IS-IS priority on port 1/1 to 128. Since the command does not specify Level-1 or Level-2, the new priority setting applies to both IS-IS levels.

Syntax: [no] isis priority <num> [level-1 | level-2-only]

The <num> parameter specifies the priority and can be from 0 – 255. A higher numeric value means a higher priority. The default is 64.

The **level-1 | level-2-only** parameter applies the priority to Level-1 or Level-2 only. By default, the priority is applied to both levels.

Possible values: See above

Default value: 64

keepalive

Disables or re-enables a POS interface to send keepalive messages to the POS interface at the other end of the link. Keepalive messages are enabled by default.

EXAMPLE:

To disable keepalive messages on POS interface 2/1:

```
BigIron(config)# interface pos 2/1
BigIron(config-posif-2/1)# no keepalive
```

Syntax: [no] keepalive

Possible values: N/A

Default value: Keepalive messages are enabled by default.

l2-session-switch

Disables or re-enables session-based Layer 2 switching on a port.

NOTE: This command applies only to Chassis devices managed by a VM1 management module.

Some features on a Chassis device managed by a VM1 management module create session entries for forwarding Layer 2 and Layer 3 traffic. A session entry consists of a traffic flow's source and destination IP addresses and IP protocol, and the source and destination TCP or UDP application port numbers if applicable. The features that create session entries include ACLs, rate limiting, PBR, NAT, and NetFlow.

The first time traffic for a given flow enters the device, the device creates a session entry for the flow if one of the features that creates session entries is configured on the flow's inbound or outbound port. The device then uses the session entry to forward subsequent traffic in the flow.

It is possible for another device in the network to forward traffic back to the Foundry device. If the traffic that is forwarded back by the other device contains the same information as a session entry already on the Foundry device, it is possible for a loop to occur. For example, if the Foundry device uses a session entry to forward traffic to a load-balancing device, and the load-balancing device reroutes the traffic back through the Foundry device to another load-balancing device, by changing the Layer 2 information but not the Layer 3 information, the Foundry device will nonetheless forward the rerouted traffic back out the same port to the same load-balancing device.

This occurs because the Foundry device again uses the session entry that matches the Layer 3 information to forward the traffic. In this case, even though the traffic's destination (the other load-balancing device) is reachable at Layer 2, the Foundry device performs Layer 2 switching of the traffic based on the session entry, which is based on Layer 3 information.

If this situation occurs in your network, you can disable session-based Layer 2 switching on individual ports.

EXAMPLE:

```
BigIron(config-if-1/1)# no l2-session-switch
```

Syntax: [no] l2-session-switch

NOTE: This command is not valid on virtual routing interfaces. The command is valid only on individual physical ports.

Possible values: N/A

Default value: Enabled

link-aggregate active | passive | off

Enables 802.3ad link aggregation.

EXAMPLE:

To enable link aggregation on a set of ports, enter commands such as the following at the interface configuration level of the CLI:

```
BigIron(config)# interface ethernet 1/1
BigIron(config-if-e1000-1/1)# link-aggregate active
BigIron(config)# interface ethernet 1/2
BigIron(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.

The following commands enable passive link aggregation on ports 1/5 – 1/8

```
BigIron(config)# interface ethernet 1/5 to 1/8
BigIron(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:

```
BigIron(config-if-e1000-1/8)# link-aggregate off
```

Syntax: [no] link-aggregate active | passive | off

The **active** parameter enables 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.

The **passive** parameter enables 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.

The **off** parameter disables link aggregation on the interface.

Possible values: See above

Default value: Disabled (off)

link-aggregate configure

Configures 802.3ad link aggregation parameters.

EXAMPLE:

You can configure one or more parameters on the same command line, and you can enter the parameters in any order. 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:

```
BigIron(config)# interface ethernet 1/1 to 1/4
BigIron(config-mif-1/1-1/4)# link-aggregate configure key 10000
BigIron(config-mif-1/1-1/4)# interface ethernet 3/5 to 3/8
BigIron(config-mif-3/5-3/8)# link-aggregate configure key 10000
```

This command changes 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.

Possible values: See above

Default value: See above

load-interval

Adjusts the load interval for the interface. The **show interface** command displays the interface's input and output load in terms of bits per second, packets per second, and utilization percentage, averaged over the load interval.

EXAMPLE:

To change the load interval for interface 3/11 to 60 seconds:

```
BigIron(config)# int e 3/11
BigIron(config-if-e100-3/11)# load-interval 60
```


To change the load interval on multiple interfaces at the same time:

```
BigIron(config)# interface ethernet 2/1 to 2/11
BigIron(config-mif-2/1-2/11)# load-interval 150
```

Syntax: load-interval <interval>

Possible values: 30 – 300 seconds, in multiples of 30 seconds

Default value: 300 seconds

loop

Configures an ATM or POS interface for internal loopback. Foundry ATM and POS interfaces can use the following loopback configurations for self tests:

- ATM interfaces:
  - Disabled – The port is not in loopback mode. Loopback must be disabled to use the port for normal traffic. This is the default.
  - Internal – Packets that the router transmits on the interface are looped back to the interface's ATM framer. The internal loopback configuration is useful for checking the ATM circuitry.
  - Line – The interface's transmit and receive fibers are logically linked so that packets received on the receive fiber are sent back out on the transmit fiber. Use this mode on the ATM interfaces on both ends of a link to test the interfaces along with the link.
- POS interfaces:
  - Disabled – The port is not in loopback mode. Loopback must be disabled to use the port for normal traffic. This is the default.
  - Internal – Packets that the router transmits on the interface are looped back to the interface's POS framer. The internal loopback configuration is useful for checking the POS circuitry.
  - Line – The interface's transmit and receive fibers are logically linked so that packets received on the receive fiber are sent back out on the transmit fiber. Use this mode on the POS interfaces on both ends of a link to test the interfaces along with the link.

NOTE: Loopback must be disabled if you want to use the port for normal traffic. Do not specify a loopback source (which automatically enables loopback) unless you are testing the interface.

EXAMPLE:

To configure ATM interface 4/1 for internal loopback, enter the following commands:

```
BigIron(config)# interface atm 4/1
BigIron(config-atmif-4/1)# loop internal
```

To disable loopback again so you can use the port for normal traffic, enter a command such as the following:

```
BigIron(config-atmif-4/1)# no loop internal
```

EXAMPLE:

To configure POS interface 2/1 for internal loopback:

```
BigIron(config)# interface pos 2/1
BigIron(config-posif-2/1)# loop internal
```

Syntax: [no] loop internal | line

The **internal** and **line** parameters specify the path for the loopback. The **internal** parameter loops packets transmitted on the interface back to the framer. The **line** parameter loops packets that are received on the receive fiber of the port back out on the transmit fiber.

Possible values: See above

Default value: Disabled

mac deny-src-mac-filter-group

NOTE: This command applies to releases 07.6.03 and later.

Applies MAC filters to virtual routing interfaces; however, MAC filters used on a virtual routing interface can only deny packets. Permit is not available. Packets are denied based on their source MAC address. The Layer 3 Switch will drop any Layer 2 or Layer 3 packet that originated from the specified source MAC address.

NOTE: No etype arguments will be checked.

EXAMPLE:

To apply a MAC filter on a virtual routing interface using the CLI, first create a filter group that denies specific source MAC addresses using the **mac filter-group** command. (Refer to the*Foundry Switch and Router Installation and Basic Configuration Guide* for details.) Then use the **mac deny-src-mac-filter-grp...** command to apply them to virtual routing interfaces. Enter commands such as the following:

```
BigIron(config)# interface ve 2
BigIron(config-vif-2)# mac filter 1 deny 00a0.cc77.a18d ffff.ffff.ffff any
BigIron(config-vif-2)# mac filter 2 deny 0010.2222.3333 ffff.ffff.ffff any
BigIron(config-vif-2)# mac deny-src-mac-filter-grp 1 2
```

Syntax: [no] mac deny-src-mac-filter-group <number>

<number> is the number of the ID of the filter that you've defined. You can enter up to eight filter IDs.

Possible values: See above

Default value: Disabled

mac filter-group

Applies a group of MAC filters to an interface. You can configure one filter group on each interface. The MAC filters apply to incoming traffic only.

NOTE: You must define the filters at the global CONFIG level using the **mac filter** command (see "mac filter" on page 6-84) before you can apply the 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.

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

EXAMPLE:

To assign MAC filter 1 to interface port 1 on slot 1, enter the following:

```
BigIron(config-if-1/1)# mac filter-group 1
```

Syntax: mac-filter-group <filter-list>

Possible values: 1 – 1024

Default value: N/A

mac filter-group log-enable

Enables logging of packets that are denied by Layer 2 MAC filters. When you enable this feature, the device generates Syslog entries and SNMP traps for denied packets.

See Example 4 in "show logging" on page 43-93 for an example of log entries generated by this feature.

EXAMPLE:

```
BigIron(config)# int e 1/1

BigIron(config-if-1/1)# mac filter-group log-enable
```

Syntax: mac filter-group log-enable

Possible values: N/A

Default value: Disabled

monitor

Allows you to select a port to be diagnosed by a designated mirror port. You can configure incoming, outgoing or both incoming and outgoing traffic to be monitored. To observe the monitored traffic, attach a protocol analyzer to the mirror port. See "mirror-port" on page 6-88.

EXAMPLE:

To monitor both incoming and outgoing traffic on interface 5/1:

```
BigIron(config)# interface e 5/1
BigIron(config-if-5/1)# monitor both
```

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

The **ethernet** <portnum> parameter specifies the mirror port(s).

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

Possible values: N/A

Default value: Disabled

mtu

Configures the MTU (Maximum Transmission Unit) for a port on a JetCore device. The MTU specifies the maximum number of bytes a frame transmitted on the port can contain.

NOTE: The MTU on both sides of an ATM PVC or POS link must match.

NOTE: The Ethernet MTU on an IronCore device cannot be larger than 1500. On JetCore devices, the Ethernet MTU can be up to 14336. The maximum MTU for a 10 Gigabit Ethernet port is one of the above, depending on whether the port is in a chassis managed by an IronCore management module or a JetCore management module.

NOTE: You must save the configuration change and then reload the software to place an MTU change into effect.

NOTE: To globally change the MTU (valid only on a JetCore device), see "default-mtu" on page 6-23.

EXAMPLE:

To enable jumbo support on individual ports, enter commands such as the following:

```
BigIron(config)# interface ethernet 1/1 to 1/2
BigIron(config-mif-1/1-1/2)# mtu 14336
BigIron(config-mif-1/1-1/2)# write memory
```



```
BigIron(config-mif-1/1-1/2)# end
BigIron# reload
```

Syntax: [no] mtu <num>

These commands change the MTU on ports 1/1 and 1/2 to 14436. When you set the MTU on an individual port or group of ports, this setting overrides the global MTU setting.

EXAMPLE:

To change the MTU on ATM sub-interface 4/1.1, enter the following commands:

```
BigIron(config)# interface atm 4/1.1
BigIron(config-atmif-4/1.1)# mtu 9180
```

EXAMPLE:

To change the MTU for POS interface 2/1 to 1200 bytes:

```
BigIron(config)# interface pos 2/1
BigIron(config-posif-2/1)# mtu 1200
```

Syntax: [no] mtu <num>

The <num> parameter specifies the MTU.

- ATM – The MTU can be a value from 1500 – 9180. The default is 4470.
- Ethernet (JetCore) – The MTU can be a value from 64 – 14436. The default is 1518.
- Ethernet (IronCore) – The MTU can be a value from 572 – 1500 for Ethernet II packets or 572 – 1492 for SNAP Ethernet packets. The default for Ethernet II packets is 1500. The default for SNAP packets is 1492.
- POS – The MTU can be a value from 60 – 4470. The default is 4470.

Possible values: See above

Default value: See above

multicast limit

Specifies the maximum number of multicast packets the port can forward each second.

EXAMPLE:

```
BigIron(config-if-3/6)# multicast limit 55000
```

Syntax: multicast limit <num>

Possible values: 0 – 4294967295; if you specify 0, limiting is disabled.

Default value: N/A

mvlan-mode

Specifies a default port-based VLAN for MAC addresses that the Layer 2 Switch receives from sources other than the MAC VLAN list you configure. For example, if a host plugs into the network and that host's MAC address is not in the MAC VLAN list, the switch assigns that VLAN to the default VLAN you specify. See the "Configuring Virtual LANs (VLANs)" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

NOTE: The MAC VLAN feature and this command apply only to FastIron Backbone Layer 2 Switches.

You can specify the default VLAN on an individual interface basis. You can specify the same VLAN or different VLANs on each interface.

EXAMPLE:

To specify the default VLAN for an interface, enter commands such as the following:

```
FastIron(config)# int e 1
FastIron(config-if-1)# mvlan-mode 2
```


Syntax: mvlan-mode <vlan-id>

Possible values: A valid port-based VLAN ID

Default value: N/A

no

Disables other commands. To disable a command, place the word **no** before the command.

phy-mode

If a port on a Foundry device is to be attached to a Bay Networks™ 28000 switch, enter this command at the Interface Level as shown below.

This command helps the Foundry device to adjust to interoperability requirements of the 28000.

EXAMPLE:

```
BigIron(config)# int e 3/1
BigIron(config-if-3/1)# phy-mode 28k
```

Syntax: phy-mode 28k

Possible values: 28k

Default value: Option is turned off.

port security

Enables the MAC port security feature on the interface and launches you into the interface port security configuration level.

EXAMPLE:

```
BigIron(config)# int e 7/11
BigIron(config-if-e100-7/11)# port security
BigIron(config-port-security-e100-7/11)#
```

Syntax: [no] port security

Possible values: N/A

Default value: Disabled

port-name

Assigns a name to a port. Assigning a name to a physical interface (port) provides additional identification for a segment on the network.

EXAMPLE:

```
BigIron(config)# interface e 5/1
BigIron(config-if-1)# port-name marketing
```

Syntax: port-name <string>

Possible values: N/A

Default value: N/A

pos flag

Changes POS flag values. The POS flags set POS "overhead" values, which are values in the packet headers that identify the packet payload content type, the packet's compatibility with some other SDH devices, and the frame type.

- c2 – Identifies the payload type. This flag can have one of the following values:
  - cf (0xCF) – PPP or HDLC. This is the default.
  - 13 (0x13) – ATM

- j0 – This is the section trace byte, formerly the C1 byte. You can set this flag to 1 (0x01) for interoperability with certain SDH devices. The default value is cc (0xCC).
- j1 – This is the Path Trace field, which transmits a 64-byte, fixed-length string that the receiving Path Terminating Equipment can use to verify its connection to the device that sent the SPE. You can configure the string a POS interface transmits in the Path Trace field.
- h1 – Identifies the frame type. This flag is part of the payload pointer byte (bits 5 and 6 of the H1 number 1 payload pointer byte). The flag can have one of the following values:
  - 0 (0x00) – SONET. This is the default.
  - 2 (0x02) – SDH

EXAMPLE:

To change the j0 flag from the default value to 1, enter the following commands:

```
BigIron(config)# interface pos 2/1
BigIron(config-posif-2/1)# pos flag j0 1
```

Syntax: [no] pos flag c2 | h1 | j0 <value(Hex)>

Syntax: [no] pos flag j1 <string>

Possible values: The **c2**, **h1**, **j0**, and **j1** parameters specify the flag you are setting.

The <value(Hex)> parameter specifies the value you are assigning to the flag. The flag values are hexadecimal numbers.

The <string> parameter specifies the string to be transmitted in the Path Trace field. The string can be up to 62 bytes long. If you do not explicitly configure a Path Trace string, the Foundry device transmits a Path Trace string consisting of a concatenation of the hostname of the device, interface number, and IP address of the interface.

Default value: See above.

pos framing

Changes the frame type on a POS interface. Foundry POS interfaces support the following frame types:

- SDH (Synchronous Digital Hierarchy) – An international standard for optical digital transmission at rates from 155 Mbps (used for STM-1) to 2.5 Gbps (used for STM-16) and higher.
- SONET (Synchronous Optical Network) – An American National Standards Institute (ANSI) standard (T1.1051988) for optical digital transmission at rates from 51 Mbps (used for OC-1) to 2.5 Gbps (used for OC-48) and higher.

EXAMPLE:

To change the frame type of POS interface 2/1 to SDH:

```
BigIron(config)# interface pos 2/1

BigIron(config-posif-2/1)# pos framing sdh
```

Syntax: pos framing sdh | sonet

Possible values: The **sdh** and **sonet** parameters specify the framing type.

Default value: sonet

pos scramble-atm

Enables a POS interface to scramble the data in the Synchronous Payload Envelope (CPE), which is the data portion of ATM packets. Scrambling the data provides additional security.

NOTE: Both ends of the link must use the same scrambling algorithm.

EXAMPLE:

To enable scrambling on POS interface 2/1:


```
BigIron(config)# interface pos 2/1
BigIron(config-posif-2/1)# pos scramble-atm
```

Syntax: pos scramble-atm

Possible values: N/A

Default value: disabled

pos threshold

Sets the thresholds for when an APS switchover occurs.

EXAMPLE:

To set thresholds for bit error rate, signal degradation, and signal failure on a POS interface:

```
BigIron(config)# interface pos 2/1
BigIron(config-posif-2/1)# pos threshold b1-tca 9
BigIron(config-posif-2/1)# pos threshold sd-ber 5
BigIron(config-posif-2/1)# pos threshold sf-ber 5
BigIron(config-posif-2/1)# exit
```

Syntax: [no] pos threshold <alarm> <threshold>

Possible values:

The <alarm> parameter can be one of the following:

- **b1-tca** – B1 bit error rate alarm
- **b2-tca** – B2 bit error rate alarm
- **b3-tca** – B3 bit error rate alarm
- **sd-ber** – Signal degradation alarm
- **sf-ber** – Signal failure alarm

The <threshold> parameter is the bit error rate from 3 – 9.

Default values: For all alarms except **sf-ber**, the default is 6 (10e-6). For the **sf-ber** alarm, the default is 3 (10e-3).

priority

Sets the QoS priority for a port. The priority values depend on the device type:

- Chassis devices, the FastIron 4802, or the TurboIron/8 – You can select one of the following:
  - 0 or 1– Assigns an internal priority queue of 0. This is the default and is normal priority.
  - 2 or 3 – Assigns an internal priority queue of 1.
  - 4 or 5 – Assigns an internal priority queue of 2.
  - 6 or 7 – Assigns an internal priority queue of 3
- Stackable devices – You can select one of the following:
  - Normal – This is the default.
  - High – All packets for this port are forwarded before packets on ports with normal priority are forwarded.

NOTE: The ServerIron provides normal, high, cache, and fw. The "cache" and "fw" QoS queues apply only to Transparent Cache Switching (TCS) and Firewall Load Balancing (FWLB) and enable these features.

See the "IronClad Quality of Service (QoS)" chapter in the *Foundry Enterprise Configuration and Management Guide* for information about how the queues work.

EXAMPLE:

```
BigIron(config)# interface e 5/1
BigIron(config-if-5/1)# priority 7
```

Here is the syntax for Chassis devices, the FastIron 4802, and the TurboIron/8.

Syntax: priority <0-7>

Here is the syntax for Stackable devices.

Syntax: priority high | normal

Possible values: see above

Default value: 0 or normal

pvst-mode

Statically enables support for Cisco Systems' Per VLAN Spanning Tree (PVST).

PVST/PVST+ support is automatically enabled on a port if the port receives a BPDU in PVST/PVST+ format. However, you can statically enable PVST/PVST+ support on a port if desired. In this case, the support is enabled immediately and support for Foundry tagged BPDUs is disabled at the same time.

NOTE: When PVST/PVST+ support is enabled on a port, support for Foundry BPDUs is disabled.

EXAMPLE:

To enable PVST/PVST+ support on a port, enter commands such as the following:

```
BigIron(config)# interface ethernet 1/1
BigIron(config-if-1/1)# pvst-mode
```

Syntax: [no] pvst-mode

NOTE: If you disable PVST/PVST+ support, the software still automatically enables PVST/PVST+ support if the port receives an STP BPDU with PVST/PVST+ format.

Possible values: N/A

Default value: Enabled automatically when a PVST/PVST+ BPDU is received on the port

qos

Enables ToS-based QoS on an interface.

NOTE: This command is supported only on the NetIron stackable Layer 3 Switch, and is available only in software release 07.1.16 or higher. The command is not supported in software release 07.2.x.

EXAMPLE:

```
NetIron(config-if-1)# qos
```

Syntax: [no] qos

Possible values: N/A

Default value: Disabled

qos mark

Enables marking for IP ToS-based QoS.

NOTE: This command is supported only on the NetIron stackable Layer 3 Switch, and is available only in software release 07.1.16 or higher. The command is not supported in software release 07.2.x.

Marking changes the value of an outbound packet's 802.1p priority field or ToS field to match the results of the QoS mappings performed by the Foundry device. When you enable marking on a port, the marking applies to packets that enter the device through that port.

EXAMPLE:

To enable CoS marking for port 1, enter the following command:

```
NetIron(config-if-1)# qos mark cos
```

Syntax: [no] qos mark cos | dscp

The **cos | dscp** parameter specifies the type of marking.

- **cos** – The device changes the outbound packet's 802.1p priority value to match the results of the device's QoS mapping from the ToS value (IP precedence or DSCP) into the 802.1p value.
- **dscp** – The device changes the outbound packet's ToS value to match the results of the device's QoS mapping from IP precedence to DSCP.

NOTE: If you specify **cos**, the device does not examine the contents of the ToS field and therefore does not perform ToS-based QoS.

When you enable marking on a port, the marking applies to packets received by the device through that port.

DSCP marking is applicable only when the trust level is IP precedence; 802.1p priority marking is applicable only when the trust level is IP precedence or DSCP. Neither type of marking is applicable if the trust level is CoS, since the CoS trust level does not examine the contents of the ToS field at all.

NOTE: DSCP marking does not apply when the trust level is CoS because this trust level does not apply to the ToS field. DSCP marking does not apply when the trust level is DSCP, because in this case the ToS field already contains a DSCP value.

Possible values: See above

Default value: N/A

qos trust

Sets the trust level for IP ToS-based QoS.

NOTE: This command is supported only on the NetIron stackable Layer 3 Switch, and is available only in software release 07.1.16 or higher. The command is not supported in software release 07.2.x.

The trust level specifies where you want the device to get the QoS value for an IP version 4 packet received on the interface. To use ToS-based QoS for packets received on the interface, you must enable the IP precedence or DSCP trust level.

EXAMPLE:

To set the trust level for port 1 to DSCP, enter the following commands:

```
NetIron(config)# interface ethernet 1
NetIron(config-if-1)# qos trust dscp
```

Syntax: [no] qos trust cos | ip-prec | dscp

The **cos | ip-prec | dscp** parameter specifies the trust level.

- **cos** – The device uses the 802.1p priority value in the packet's IEEE frame header.
- **ip-prec** – The device uses the three most-significant bits in the packet's ToS field and interprets them as an IP precedence value.
- **dscp** – The device uses the six most-significant bits in the packet's ToS field and interprets them as a DSCP value.

Possible values: See above

Default value: CoS

qos-tos mark

Enables marking, used by the JetCore IP ToS-based QoS feature. Marking changes the value of an outbound packet's 802.1p priority field, ToS field, or both to match the results of the QoS mappings performed by the Foundry device. When you enable marking on an interface, the marking applies to packets that enter the device through that interface.

NOTE: This command is supported in release 07.6.01 and later and applies only to JetCore devices.

NOTE: ToS-based QoS must be enabled. See "port-priority" on page 6-92.

NOTE: When port priority is enabled, Foundry devices will use the higher of the 802.1p and DSCP priority for its internal system priority. To override this internal system priority, apply the **qos-tos mark cos** command on the interface where advanced ToS-based QoS was enabled. The true internal system priority will then be based on how the **qos-tos trust** command is configured on that interface. (See "Specifying the Trust Level" in the *Foundry Enterprise Configuration and Management Guide*.)

If the port for incoming traffic is a tagged port, then the outgoing 802.1p priority will be the "merged" priority even though the type of marking configured on the interface is actually lower than the "merged" priority for that port. If the port for the incoming traffic is an untagged port and the port for outgoing traffic is an 802.1p port, then the marking configured on these ports will be used. This is useful in the case where the port priority (via priority command under interface CLI) is higher and the user wants to mark all IP packets with a lower priority. Hence, the non-IP packets will continue to use the higher 802.1p priority.

EXAMPLE:

```
BigIron(config-if-1/1)# qos-tos mark cos
```

This command enables marking of the 802.1p field in the Ethernet frame.

Syntax: [no] qos-tos mark cos | dscp

The **cos | dscp** parameter specifies the type of marking.

- **cos** – The device changes the outbound packet's 802.1p priority value to match the results of the device's QoS mapping from the specified trust level.
- **dscp** – The device changes the outbound packet's DSCP value to match the results of the device's QoS mapping from the specified trust level.

Possible values: See above

Default value: None

qos-tos trust

Changes the trust level used by the JetCore IP ToS-based QoS feature. The trust level specifies where you want the device to get the QoS value for a packet received on the interface.

NOTE: This command is supported in release 07.6.01 and later and applies only to JetCore devices.

NOTE: ToS-based QoS must be enabled. See "port-priority" on page 6-92.

EXAMPLE:

```
BigIron(config-if-1/1)# qos-tos trust ip-prec
```

Syntax: [no] qos-tos trust cos | ip-prec | dscp

The **cos | ip-prec | dscp** parameter specifies the trust level.

- **cos** – The device uses the 802.1p (CoS) priority value in the packet's Ethernet frame header. Use this trust option when you plan to mark the packet's DSCP value based on the incoming 802.1p value.
- **ip-prec** – The device uses the three most-significant bits in the packet's ToS field and interprets them as an IP precedence value. Use this trust option when the incoming packet is from a device that does not support DSCP and you need to mark the packet for QoS on DSCP devices.
- **dscp** – The device uses the six most-significant bits in the packet's ToS field and interprets them as a DSCP value. This is the default.

Possible values: See above

Default value: DSCP

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-if-1)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

rate-limit control-packet

Disables or re-enables rate limiting exemption for control packets on an interface.

NOTE: This command applies only to Adaptive Rate Limiting, not Fixed Rate Limiting.

```
BigIron(config-if-e1000-25)# rate-limit control-packet no
```

This command disables exemption of all the control packet types listed below on port 25.

Syntax: [no] rate-limit control-packet no | yes

To re-enable exemption for the interface, enter the following command:

```
BigIron(config-if-e1000-25)# rate-limit control-packet yes
```

Table 7.1: IP Control Traffic Exempt from Rate Limiting

Traffic Type		IP Address	IP Protocol or Application Port
IP multicast	IP nodes multicast	224.0.0.1	
	IP routers multicast	224.0.0.2	
	IP DVMRP router multicast	224.0.0.4	
	IP OSPF router multicast	224.0.0.5	
	IP OSPF designated router multicast	224.0.0.6	
	IP RIP V.2 router multicast	224.0.0.9	
	IP VRRP multicast	224.0.0.18	

Table 7.1: IP Control Traffic Exempt from Rate Limiting (Continued)

Traffic Type		IP Address	IP Protocol or Application Port
IP unicast	BGP control packet		TCP port 179 (0xB3)
	OSPF control packet		IP protocol type 89 (0x59)
	RIP packet		UDP port 520 (0x0208)

Possible values: yes (enable exemption) or no (disable exemption)

Default value: all control packet types are exempted from rate limiting

rate-limit in | out

Configures a rate limiting policy on a JetCore Chassis device or the FastIron 4802.

EXAMPLE:

To configure an inbound port-based policy, enter commands such as the following:

```
BigIron(config)# interface ethernet 1/1
BigIron(config-if-e100-1/1)# rate-limit in 600000
The average rate has been adjusted to 513024
```

These commands configure an inbound policy on 10/100 Ethernet port 1/1 with an Average Rate of 513024 bps. The following commands configure an inbound rate limiting policy on a Gigabit Ethernet port.

```
BigIron(config)# interface ethernet 2/1
BigIron(config-if-e1000-2/1)# rate-limit in 2000000
The average rate has been adjusted to 2051328
```

To configure an outbound port-based policy, enter commands such as the following:

```
BigIron(config)# interface ethernet 1/2
BigIron(config-if-e100-1/2)# rate-limit out 5000000
The average rate has been adjusted to 5000192
```

These commands configure an outbound policy on 10/100 Ethernet port 1/2 with an Average Rate of 5000192 bps. The following commands configure an outbound rate limiting policy on a Gigabit Ethernet port.

```
BigIron(config)# interface ethernet 2/2
BigIron(config-if-e1000-2/2)# rate-limit out 40000000
The average rate has been adjusted to 40000512
```

EXAMPLE:

Port-and-priority-based rate limiting is supported for inbound traffic only. To configure a port-and-priority-based policy, enter commands such as the following:

```
BigIron(config)# interface ethernet 1/1
BigIron(config-if-e100-1/1)# rate-limit in priority q0 q2 600000
The average rate has been adjusted to 513024
```

These commands configure an inbound policy on 10/100 Ethernet port 1/1, for hardware forwarding queues q0 and q2 with an Average Rate of 769280 bps. The policy applies only to traffic that is received on the port and is placed in the specified forwarding queues.

EXAMPLE:

You can use standard or extended IP ACLs for ACL-based rate limiting. ACL-based rate limiting is supported for inbound traffic only.

- Standard IP ACLs match traffic based on source IP address information.
- Extended ACLs match traffic based on source and destination IP address and IP protocol information. For TCP and UDP, they also match on source and destination TCP or UDP addresses.

To configure ACL-based policies on a port, enter commands such as the following:

```
BigIron(config)# access-list 50 permit host 1.1.1.2
BigIron(config)# access-list 60 permit host 2.2.2.3
BigIron(config)# interface ethernet 1/1
BigIron(config-if-e100-1/1)# rate-limit in access-group 50 600000
The average rate has been adjusted to 513024
BigIron(config-if-e100-1/1)# rate-limit in access-group 60 3000000
The average rate has been adjusted to 3077120
```

These commands configure two inbound rate limiting policies on 10/100 Ethernet port 1/1. The first policy rate limits traffic from IP host 1.1.1.2. The second policy rate limits traffic from IP host 2.2.2.3.

NOTE: Use the **permit** condition for traffic that you want to include in the policy. If you use the **deny** condition, the policy does not apply to the specified traffic. Depending on whether the strict ACL option is enabled, the device either forwards denied traffic without rate limiting it, or drops the traffic. See the "Using ACLs for Filtering in Addition to Rate Limiting" section in the "JetCore Adaptive Rate Limiting" chapter of the *Foundry Enterprise Configuration and Management Guide*.

NOTE: You must configure the ACLs before you can use them to configure the rate limiting policy.

Syntax: [no] rate-limit in | out
[[priority q0 | q1 | q2 | q3] | [access-group <acl-id>]]
<average-rate>

The **in | out** parameter specifies the traffic direction to which the policy applies.

NOTE: The **out** option is supported only for port-based rate limiting. The **out** option is not supported for port-and-priority-based rate limiting or ACL-based rate limiting.

The syntax allows you to configure a port-based policy, a port-and-priority-based policy, or an ACL-based policy.

- To create a port-based policy, do not use the **priority** or **access-group** parameters.
- To create a port-and-priority-based policy, use the **priority** parameter.
- To create an ACL-based policy, use the **access-group** parameter.

The **priority q0 | q1 | q2 | q3** parameter specifies the hardware forwarding queue to which the policy applies. Use this parameter only if you are configuring a port-and-priority-based policy. The device prioritizes the queues from **q0** (normal priority) to **q3** (highest priority).

The **access-group** <acl-id> parameter specifies an IP ACL. Use this parameter only if you are configuring an ACL-based policy.

The <average-rate> parameter specifies the maximum number of bits per second (bps) you want the device to allow on the port. You can specify a value in the following ranges:

- Inbound rate limiting on 10/100 Ethernet: 256512 – 100000000 bps.
- Inbound rate limiting on Gigabit Ethernet: 1025792 – 1000000000 bps.
- Outbound rate limiting on 10/100 Ethernet: 1041910 – 100000000 bps.
- Outbound rate limiting on Gigabit Ethernet: 20833792 – 1000000000 bps.

NOTE: The software adjusts the Average Rate you enter so that the calculation of credits does not result in a remainder of a partial Credit. The CLI displays the adjusted rate. You also can display a table of the adjusted rate values. See "show rate-limit adjusted-rate" on page 43-122.

Possible values: See above

Default value: None configured

rate-limit input | output

Configures a rate-limiting policy on an interface.

NOTE: This command applies only to Adaptive Rate Limiting, not Fixed Rate Limiting. This command does not apply to the FastIron 4802. For the FastIron 4802 **rate-limit** syntax, see “rate-limit in | out” on page 7-68.

NOTE: After you configure the rate limiting policy, you need to apply the policy to an interface for the policy to take effect. See “rate-limit input | output” on page 7-70.

EXAMPLE:

```
BigIron(config)# interface virtual ve2
BigIron(config-ve-2)# rate-limit input access-group ratelimit 100 4000000 320000
400000 conform-action transmit exceed-action drop
```

Syntax: [no] rate-limit input | output [access-group <num>] <average-rate> <normal-burst-size> <excess-burst-size> conform-action <action> exceed-action <action>

NOTE: Devices managed by a VM1 have an additional option, **vlan** <vlan-id>, for use when configuring port-, VLAN-, and direction-based rate limiting. See the “Configuring Port-, VLAN- and Direction-Based Rate Limiting (VM1 only)“ section in the “Configuring IronClad Rate Limiting (IronCore)“ chapter of the *Foundry Enterprise Configuration and Management Guide*.

The **input | output** parameter specifies whether the rule applies to inbound traffic or outbound traffic.

- Specify **input** for inbound traffic.
- Specify **output** for outbound traffic.

The **access-group** <num> parameter specifies an ACL. When you use this parameter, the rule applies only to traffic that matches the specified ACL. Otherwise, the rule applies to all Ethernet traffic that does not match a previous rule on the interface. You can specify the number of a standard ACL, and extended ACL, or a rate limit ACL. If you specify a rate limit ACL, use the parameter **ratelimit** (without a space) in front of the ACL number; for example, **ratelimit 100**.

NOTE: You cannot specify a named ACL.

The <average-rate> parameter specifies the portion, in bits per second (bps) of the interface’s total bandwidth you want to allocate to traffic that matches the rule. You can specify a value can from 262144 (256Kbps) up to the maximum line rate of the port. For example, for a 100Mbps port, the maximum value is 100,00,000 (100Mbps).

If the interface is a trunk group, a virtual interface, or a VLAN, you can specify a value up to the maximum combined line rate of all the ports in the interface. For example, if the interface is a trunk group that consists of two one-Gigabit Ethernet ports, then the maximum value for <average-rate> is 200,000,000 (two times the maximum for each of the individual Gigabit ports).

The <normal-burst-size> parameter specifies the maximum number of bytes that specific traffic can send on the interface within the Committed Time Interval and still be within that traffic's rate limit. The minimum value is 3277[1] or 1/10th of the Average Rate (whichever is higher), and the maximum value is the Average Rate. The smallest fraction of the Average Rate you can specify is 1/10th.

The <excess-burst-size> parameter specifies the maximum number of additional bytes (bytes over the <normal-burst-size>) that can be transmitted within the Committed Time Interval. The <excess-burst-size> can be a value from the <normal-burst-size> up to the maximum number of bytes the interface can forward within the Committed Time Interval.

1.This value comes from dividing the minimum Average Rate (262144 bits) by eight to get 32768 bytes, then dividing 32768 bytes by 10 to get 3276.8, since the smallest fraction of the Average Rate you can specify is 1/10th. The value 3276.8 is then rounded up to 3277.

The device can take different actions for traffic within the <normal-burst-size> and traffic that falls into the <excess-burst-size>. For example, you can forward all traffic in the <normal-burst-size> and reset the precedence to a lower priority for all <excess-burst-size> traffic, or even just drop that traffic.

NOTE: Do not set the <excess-burst-size> parameter to a value greater than the maximum number of bytes the interface can forward within the Committed Time Interval. Even if the software allows you to specify a higher value, the interface cannot forward more data than its line rate supports.

The **conform-action** <action> parameter specifies the action you want the device to take for traffic that matches the rule and is within the Normal Burst Size. You can specify one of the following actions:

- **transmit** – Send the packet.
- **set-prec-transmit** <new-prec> – Set the IP precedence, then send the packet. You can specify one of the following:
  - **0** – routine precedence
  - **1** – priority precedence
  - **2** – immediate precedence
  - **3** – flash precedence
  - **4** – flash override precedence
  - **5** – critical precedence
  - **6** – internetwork control precedence
  - **7** – network control precedence
- **set-prec-continue** <new-prec> – Set the IP precedence to one of the values listed above, then evaluate the traffic based on the next rate policy.
- **drop** – Drop the packet.
- **continue** – Evaluate the traffic based on the next rate policy.

The **exceed-action** <action> parameter specifies the action you want the device to perform for traffic that matches the rule but exceeds the <normal-burst-size> within a given Committed Time Interval. You can specify one of the actions listed above.

Possible values: See above

Default value: N/A

rate-limit input | output fixed

Configures Fixed Rate Limiting on an interface.

NOTE: This command applies only to Fixed Rate Limiting, not Adaptive Rate Limiting.

EXAMPLE:

```
BigIron(config-if-1/1)# rate-limit input fixed 500000
```

This command configures a Fixed Rate Limiting policy that allows port 1/1 to receive a maximum of 500000 bps (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 | output fixed <rate>

The **input | output** parameter specifies whether the rate limit applies to inbound or outbound traffic on the port.

The <rate> parameter specifies the maximum rate for the port. Specify the rate in bits per second. You can specify from 1 up to any number. There is no default.

NOTE: If you specify a number that is larger than the port's line rate, the traffic will never cause the policy to go into effect.

Possible values: See above

Default value: N/A

rate-limit strict-acl

Configures a JetCore Chassis device or the FastIron 4802 to drop packets that are denied by ACLs used in the rate limiting policies on the port.

When you use the ACL-based mode, the permit and deny conditions in an ACL you use in a rate limiting policy work as follows:

- Permit – The traffic is rate limited according to the other parameters in the rate limiting policy.
- Deny – The traffic is forwarded instead of dropped, by default.

You can configure the device to drop traffic that is denied by the ACL instead of forwarding the traffic, on an individual port basis.

If you use the port-based or port- and priority-based mode, ACLs forward or drop traffic based on the permit and deny conditions.

NOTE: Once you configure an ACL-based rate limiting policy on a port, you cannot configure a regular (traffic filtering) ACL on the same port. To filter traffic, you must enable the strict ACL option.

EXAMPLE:

To configure the device to drop traffic that is denied by a rate limiting ACL, enter the following command at the configuration level for the port:

```
SW-FI4802-PREM(config-if-1)# rate-limit strict-acl
```

Syntax: [no] rate-limit strict-acl

Possible values: Disabled. The device does not rate limit traffic denied by the ACLs, and forwards the traffic.

Default value: N/A

route-only

Disables Layer 2 switching on an interface.

NOTE: Make sure you really want to disable all Layer 2 switching operations on the interface before you use this option. Consult your reseller or Foundry Networks for information.

NOTE: You also can disable Layer 2 switching globally. See "route-only" on page 6-106.

EXAMPLE:

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 3/2:

```
BigIron(config)# interface ethernet 3/2
BigIron(config-if-3/2)# route-only
```

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

```
BigIron(config-if-3/2)# no route-only
```

Syntax: [no] route-only

Possible values: N/A

Default value: N/A

sflow forwarding

Enables sFlow forwarding on an interface.

NOTE: You also must enable sFlow globally. See “sflow enable” on page 6-114.

NOTE: When you enable sFlow forwarding on an 802.1X-enabled interface, the samples taken from the interface include the user name string at the inbound and/or outbound port, if that information is available. For information about 802.1X, see the *Foundry Security Guide*.

EXAMPLE:

```
BigIron(config)# sflow enable
BigIron(config)# interface ethernet 1/1 to 1/8
BigIron(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 forwarding

Possible values: N/A

Default value: Disabled

sflow sample

Changes the sFlow sampling rate on an interface.

For additional information about sampling rates, see the “Changing the Sampling Rate” section in the “Remote Network Monitoring “ chapter of the *Foundry Enterprise Configuration and Management Guide*.

EXAMPLE:

```
BigIron(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 to the next higher odd power of 2. This value becomes the actual default sampling rate and is one of the following. 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.

- 2
- 8
- 32
- 128
- 512
- 2048
- 8192
- 32768
- 131072
- 524288
- 2097152
- 8388608
- 33554432
- 134217728

- 536870912
- 2147483648

Possible values: See above

Default value: The module's sFlow sampling rate

show

Displays a variety of configuration and statistical information about the device. See "Show Commands" on page 43-1.

snmp-server enable traps link-change

Disables SNMP traps for link-state changes on a port.

EXAMPLE:

```
BigIron(config)# interface ethernet 1/1
BigIron(config-if-1/1)# no snmp-server enable traps link-change
```

Syntax: [no] snmp-server enable traps link-change

To disable link-state change traps for a range of ports, enter commands such as the following:

```
BigIron(config)# interface ethernet 1/1 to 1/8
BigIron(config-mif-1/1-1/8)# no snmp-server enable traps link-change
```

This command disables the link-state change traps on ports 1/1 – 1/8.

Possible values: N/A

Default value: Link up and link down traps are both enabled. This command disables or re-enables both traps.

sonet-scram

Disables or re-enables the SONET scramble mode.

By default, Foundry devices perform bit scrambling at the SONET level. Scrambling helps ensure that long strings of zeros or ones are converted into more random-appearing bit sequences. This is useful for ensuring clock recovery and security. Most ATM links have this feature enabled, so the default is appropriate for most ATM networks. The setting of the SONET scramble mode affects both send and receive traffic on the port.

EXAMPLE:

To disable the SONET scramble mode for ATM interface 4/1, enter the following commands:

```
BigIron(config)# interface atm 4/1
BigIron(config-atmif-4/1)# no sonet-scram
```

Syntax: [no] sonet-scram

To re-enable the feature, enter the following command:

```
BigIron(config-atmif-4/1)# sonet-scram
```

Possible values: N/A

Default value: Enabled

spanning-tree

Enables the Spanning Tree Protocol (STP) on a port.

EXAMPLE:

To enable STP on port 1/5, enter the following commands.

```
BigIron(config)# interface ethernet 1/5
BigIron(config-if-1/5)# spanning-tree
```

Syntax: [no] spanning-tree

Possible values: N/A

Default value: By default, a port has the same STP state as the VLAN it is in.

speed-duplex

Modifies port speed and duplex. It defines the speed and duplex mode for a 10BaseT and 100BaseTx ports.

Gigabit (1000BaseSx and 1000BaseLx) and 100BaseFx ports operate at a fixed speed and mode (full-duplex) and cannot be modified.

EXAMPLE:

```
BigIron(config)# interface e 1/8

BigIron(config-if-1/8)# speed-duplex 10-full
```

Syntax: speed-duplex <value>

Possible values: 10-full, 10-half, 100-full, 100-half, auto

Default value: 10/100 autosense

stlb

Enables server trunk load balancing optimization.

For information about the feature, see the “Enabling Optimized Server Trunk Load Balancing (VM1 only)“ section in the “Configuring Trunk Groups and Dynamic Link Aggregation” chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

NOTE: This command applies only to Chassis devices managed by the VM1.

EXAMPLE:

```
BigIron(config-if-e1000-1/4)# stlb
```

Syntax: [no] stlb

Possible values: N/A

Default value: Disabled

stp-boundary

Configures a boundary interface for the SuperSpan™ feature.

For information about this feature, see the “SuperSpan™“ section in the “Configuring Spanning Tree Protocol (STP) and IronSpan“ chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

EXAMPLE:

```
BigIron(config)# interface 1/1
BigIron(config-if-e1000-1/1)# stp-boundary 1
BigIron(config)# interface 1/2
BigIron(config-if-e1000-1/2)# stp-boundary 2
```

These commands configure two interfaces on the Foundry device as SuperSpan boundary interfaces. Interface 1/1 is a boundary interface with customer 1. Interface 1/2 is a boundary interface with customer 2. Each boundary interface is associated with a number, which is the SuperSpan ID. The SuperSpan ID identifies the instance of SuperSpan you are associating with the interface. Use the same SuperSpan ID for each boundary interface with the same customer. Use a different SuperSpan ID for each customer. For example, use SuperSpan ID 1 for all the boundary interfaces with customer 1 and use SuperSpan ID 2 for all boundary interfaces with customer 2.

Syntax: [no] stp-boundary <num>

The <num> parameter specifies the SuperSpan ID. You can specify a number from 1 – 65535.

Possible values: 1 – 65535

Default value: N/A

unknown-unicast limit

Specifies the maximum number of unknown-unicast packets the port can forward each second.

EXAMPLE:

```
BigIron(config-if-4/2)# unknown-unicast limit 40000
```

Syntax: unknown-unicast limit <num>

Possible values: 0 – 4294967295; if you specify 0, limiting is disabled.

Default value: N/A

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-if-1/1)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-if-1/1)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 8
Redundant Management Module CONFIG Commands

active-management

In Chassis devices containing redundant management modules, changes the default assignment of the active management module. By default, the redundant management module in the lower slot number becomes the active redundant management module. You must use this command to override the default and make the redundant management module in the higher slot number the default active module.

NOTE: This command applies only to devices containing redundant management modules.

NOTE: The change does not take effect until you reload the system. If you save the change to the active module's system-config file before reloading, the change persists across system reloads. Otherwise, the change affects only the next system reload.

EXAMPLE:

To override the default and specify the active redundant management module, enter the following commands:

```
BigIron(config)# redundancy
BigIron(config-redundancy)# active-management 5
```

This command overrides the default and makes the redundant management module in slot 5 the active module following the next reload. The change affects only the next reload and does not remain in effect for future reloads.

Syntax: active-management <slot-num>

- Slots on a 4-slot chassis are numbered 1 – 4, from top to bottom.
- Slots on an 8-slot chassis are numbered 1 – 8, from left to right.
- Slots on a 15-slot chassis are numbered 1 – 15, from left to right.

To make the change permanent across future reloads, enter the **write memory** command to save the change to the startup-config file, as shown in the following example:

```
BigIron(config)# redundancy
BigIron(config-redundancy)# active-management 5
BigIron(config-redundancy)# write memory
```

NOTE: If you do not save the change to the startup-config file, the change affects only the next reload.

end

Moves activity to the privileged EXEC level from any level of the CLI, with the exception of the user level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
BigIron(config-redundancy)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the privileged level.

EXAMPLE:

To move from the global level, back to the privileged level, enter the following:

```
BigIron(config-redundancy)# exit
BigIron#
```

Syntax: exit

Possible values: N/A

Default value: N/A

no

Disables other commands. To disable a command, place the word ***no*** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-redundancy)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

sync-standby

Automates synchronization of software between active and standby redundant management modules.

EXAMPLE:

To change the automatic synchronization setting, use one of the following commands:

Syntax: [no] sync-standby boot

Syntax: [no] sync-standby code

Syntax: [no] sync-standby startup-config

Syntax: [no] sync-standby running-config [<num>]

To disable automatic synchronization of the boot code, flash code, or startup-config file, enter “no” in front of the command.

The <num> parameter with the **sync-standby running-config** command specifies the synchronization interval. You can specify from 4 – 20 seconds. The default is 10 seconds. To disable automatic synchronization of the running-config, set the synchronization interval (the <num> parameter) to 0.

Possible values: See above

Default value: Automatic synchronization of the flash code, running-config, and system-config file is enabled by default. Automatic synchronization of the boot code is disabled by default. The default synchronization interval for the running-config is 10 seconds.

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-redundancy)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-redundancy)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 9
Trunk Commands

config-trunk-ind

Allows you to enable, disable, or monitor an individual trunk port.

Once you enter this command, the following interface-level commands apply only to individual trunk ports, not to the entire trunk group:

- **enable**
- **disable**
- **monitor ethe-port-monitored** <portnum> | **named-port-monitored** <portname> **ethernet** | **pos** <portnum> **in** | **out** | **both**

NOTE: You can rename individual trunk ports without first entering the **config-trunk-ind** command. In software release 07.5.01 and later, the **port-name** command is always valid on individual ports, including primary and secondary ports in a trunk group.

NOTE: The **config-primary-ind** command is not required in software release 07.5.01 and later and is no longer supported.

EXAMPLE:

```
BigIron(config-trunk-4/1-4/4)# config-trunk-ind
BigIron(config-trunk-4/1-4/4)# disable ethernet 4/2
```

Syntax: [no] config-trunk-ind

Syntax: [no] disable ethernet | pos <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.

Possible values: N/A

Default value: N/A

disable

Disables an individual port in a trunk group.

NOTE: To disable all ports in the trunk group, enter the **disable** command at the interface configuration level for the primary port. The primary port is the lowest-numbered port in the trunk group.

NOTE: To disable an individual port, use the **config-trunk-ind** command first. See "config-trunk-ind" on page 9-1.

EXAMPLE:

```
BigIron(config-trunk-4/1-4/4)# disable ethernet 4/1
```

This command disables port 4/1 in the trunk group consisting of ports 4/1 – 4/4.

Syntax: [no] disable ethernet <portnum> [to <portnum> | ethernet <portnum>]

Syntax: [no] disable pos <portnum> [to <portnum> | pos <portnum>]

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 **ethernet** <portnum> or **pos** <portnum> parameter specifies an individual port. You can enter this parameter multiple times to specify a list, as shown in the examples above. You cannot specify Ethernet and POS ports on the same command line.

EXAMPLE:

If you have configured a name for the trunk port, you can specify the port name, as shown in the following example:

```
BigIron(config-trunk-4/1-4/4)# disable eparker
```

Syntax: disable <portname>

Possible values: N/A

Default value: Enabled

enable

Enables an individual port in a trunk group.

NOTE: To enable an individual port, use the **config-trunk-ind** command first. See "config-trunk-ind" on page 9-1.

EXAMPLE:

```
BigIron(config-trunk-4/1-4/4)# enable ethernet 4/1
```

Syntax: [no] enable ethernet <portnum> [to <portnum> | ethernet <portnum>]

Syntax: [no] enable pos <portnum> [to <portnum> | pos <portnum>]

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 **ethernet** <portnum> or **pos** <portnum> parameter specifies an individual port. You can enter this parameter multiple times to specify a list, as shown in the examples above. You cannot specify Ethernet and POS ports on the same command line.

EXAMPLE:

If you have configured a name for the trunk port, you can specify the port name, as shown in the following example:

```
BigIron(config-trunk-4/1-4/4)# enable guinness
```

Syntax: enable <portname>

Possible values: N/A

Default value: Enabled

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged EXEC level, enter the following from any level of the CLI.

```
BigIron(config-trunk-4/1-4/4)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global CONFIG level.

EXAMPLE:

```
BigIron(config-trunk-4/1-4/4)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

monitor

Enables an individual port in a trunk group to be monitored. Use this command to monitor a secondary port.

NOTE: By default, when you monitor the primary port, aggregated traffic for all the ports in the trunk group is copied to the mirror port. To monitor an individual port, use the **config-trunk-ind** command first. See "config-trunk-ind" on page 9-1.

NOTE: In the current release, you can use only one mirror port for each monitored trunk port.

EXAMPLE:

To monitor traffic on one of the secondary ports in a trunk group, enter commands such as the following:

```
BigIron(config)# mirror ethernet 2/1
BigIron(config)# trunk switch ethernet 4/1 to 4/8
BigIron(config-trunk-4/1-4/8)# monitor ethe-port-monitored 4/5 ethernet 2/1 in
```

The **monitor ethe-port-monitored** command in this example enables monitoring of the inbound traffic on port 4/5.

Syntax: [no] monitor ethe-port-monitored <portnum> | named-port-monitored <portname> ethernet | pos <portnum> in | out | both

The **ethe-port-monitored** <portnum> | **named-port-monitored** <portname> parameter specifies the trunk port you want to monitor. Use **ethe-port-monitored** <portnum> to specify a port number. Use **named-port-monitored** <portname> to specify a trunk port name.

The **ethernet** | **pos** <portnum> parameter specifies the port to which the traffic analyzer is attached.

The **in** | **out** | **both** parameter specifies the traffic direction to be monitored.

Possible values: See above

Default value: None configured

no

Disables other commands. To disable a command, place the word ***no*** before the command.

port-name

Assigns a name to a port in a trunk group.

Once you assign a name to the port, you can use the name when configuring trunk parameters for the port. The name also is used in trunk information displays.

EXAMPLE:

```
BigIron(config-trunk-4/1-4/4)# port-name josecuervo ethernet 4/1
```

This command assigns the name “josecuervo” to port 4/1 in the trunk group consisting of ports 4/1 – 4/4.

Syntax: [no] port-name <text> ethernet | pos <portnum>

The <text> parameter specifies the port name. The name can be up to 50 characters long.

Possible values: See above

Default value: None configured

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-trunk-4/1-4/4)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-trunk-4/1-4/4)# write memory
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-trunk-4/1-4/4)# write terminal
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 10
RIP Commands

default-metric

Defines the global default-metric value that will be assigned to all external routes imported into RIP for redistribution.

RIP must be active on the Layer 3 Switch for this command to be operational.

EXAMPLE:

To assign a default metric of 4 to all routes imported as RIP, enter the following:

```
BigIron(config-rip-router)# def 4
```

Syntax: default-metric <value>

Possible values: 1 – 15

Default value: 1

deny redistribute

Defines the route types upon which you ***do not*** want to perform RIP redistribution.

RIP must be active on the Layer 3 Switch for this command to be operational. RIP is enabled by default.

EXAMPLE:

To deny redistribution on all types of routes to the 207.92.0.0 network, enter the following:

```
BigIron(config-rip-router)# deny redistribute 2 all address 207.92.0.0 255.255.0.0
```

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.

Default value: N/A

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
BigIron(config-rip-router)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
BigIron(config-rip-router)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

filter

Defines which IP network numbers the Layer 3 Switch will learn from the RIP protocol and store in its IP routing table. Once RIP filters are defined, you can assign these filters to individual interfaces with the **filter-group** command at the Interface Level of CLI.

To define a RIP filter, you must first enable RIP on the Layer 3 Switch using the **router rip** command to access the RIP Router Level of the CLI.

NOTE: A filter defines for inbound routes what routes it will permit to be stored in its IP routing table. For outbound routes, the filter defines what routes are allowed to be advertised through a given interface. You can also specify all routes by using the value, **any**, instead of specifying a specific route.

An IP address and mask define a route.

EXAMPLE:

To define filters with respect to network traffic from 192.53.41, 192.53.5.1, 192.53.6.1 and 192.53.7.1, enter the following:

```
BigIron(config-rip-router)# filter 1 permit 192.53.4.1 255.255.255.0
BigIron(config-rip-router)# filter 2 permit 192.53.5.1 255.255.255.0
BigIron(config-rip-router)# filter 3 permit 192.53.6.1 255.255.255.0
BigIron(config-rip-router)# filter 4 deny 192.53.7.1 255.255.255.0
```

EXAMPLE:

To enable logging on filter 1 and apply the filter to interface 1/2:

```
BigIron(config-rip-router)# filter 1 deny any any log
BigIron(config-rip-router)# int e 1/2
```



```
BigIron(config-if-e1000-1/2)# ip rip filter-group in 1
BigIron(config-if-e1000-1/2)# ip rip filter-group out 1
```

Syntax: filter <filter-num> permit | deny <source-ip-addr> | any <source-ip-mask> | any [log]

When the RIP filter causes packets to be denied, the following messages appear in the syslog:

```
00d00h00m00s:W:rip filter list 1 in V1 denied 0.0.0.0, 1 packets
00d00h00m00s:W:rip filter list 1 out V1 denied 0.0.0.0, 1 packets
```

The format of the syslog message is as follows:

<time>:W:rip filter list <list-num> <direction> V1 | V2 denied <ip-addr>, <num> 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.

Possible values: Filter ID: 1 – 64

Default value: N/A

filter-group

Assigns a group of defined RIP filters on either a global or interface basis. Assignments to interfaces are done at the interface level of the CLI.

EXAMPLE:

```
BigIron(config-rip-router)# filter-group out 1 3 6 9
```

Syntax: filter-group in | out <1-64> [<1-64]

Possible values: 1 – 64 (filter index value)

Default value: N/A

learn-default

This feature allows a Layer 3 Switch to learn and advertise default RIP routes. This command can be applied on a global or interface basis. This example shows the feature enabled at the global level.

EXAMPLE:

```
BigIron(config-rip-router)# learn-default
```

Syntax: learn-default

Possible values: N/A

Default value: N/A

neighbor

Specifies those routers from which a Layer 3 Switch will receive RIP routes.

In the example below, no RIP routes will be learned from any neighbor router. By default, RIP routes will be learned from all neighbors.

EXAMPLE:

To configure a Layer 3 Switch so that no RIP routes are learned from its neighbor routers, enter the following:


```
BigIron(config-rip-router)# neighbor 1 deny any
```

Syntax: neighbor <number> permit | deny <ip-addr> | any

Possible values: 1 – 64

Default value: N/A

no

Disables other commands. To disable a command, place the word **no** before the command.

offset-list

Configures 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.

EXAMPLE:

To configure a global RIP offset list, enter commands such as the following:

```
BigIron(config)# access-list 21 deny 160.1.0.0 0.0.255.255
BigIron(config)# access-list 21 permit any
BigIron(config)# router rip
BigIron(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 | pos <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.

```
BigIron(config-rip-router)# offset-list 21 in ethernet 2/1
```

Possible values: See above

Default value: None

permit redistribute

Allows you to define the route types upon which you want to perform RIP redistribution.

EXAMPLE:

To allow (permit) redistribution of all routes received from network 192.147.72.0, enter the following:

```
BigIron(config-rip-router)# permit redistribute 1 all address 192.147.72.0
255.255.255.0
```

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.

Default value: N/A

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-rip-router)# quit

BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

redistribution

Enables RIP route redistribution on a Layer 3 Switch. When enabled, RIP will import external routes (OSPF or Static Routes) into the RIP domain. Do this prior to setting up the redistribution table using the **permit** and **deny** commands.

EXAMPLE:

To enable RIP redistribution on the Layer 3 Switch, enter the following within the Router RIP Level.

```
BigIron(config-rip-router)# redistribution
```

Syntax: redistribution

Possible values: N/A

Default value: disabled

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

update-time

Sets the time interval that will exist between the transmission of regular RIP response packets. This parameter is set to 30 seconds by default. RIP must be enabled and active on the Layer 3 Switch for this command to be operational.

EXAMPLE:

To modify the default update time value to 120 seconds, enter the following:

```
BigIron(config-rip-router)# update 120
```


Syntax: update-time <value>

Possible values: 1 – 1,000 seconds

Default value: 30 seconds

use-vrrp-path

Prevents Backup VRRP routers or VSRP switches from advertising route information for the backed up interface, by enabling suppression of the advertisements. To suppress RIP advertisements for a backed up interface, enter the following command on the VRRP or VSRP Backup:

```
BigIron(config-rip-router)# use-vrrp-path
```

Syntax: use-vrrp-path

Possible values: N/A

Default value: N/A

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-rip-router)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-rip-router)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 11
OSPF Commands

area

Assigns an area ID value to which the router will be attached. Area membership is assigned using the Interface Level command, **ip ospf area** <area-id>.

- When an area is defined as **normal**, all external routes will be advertised into the area.
- When an area is defined as **stub**, external routes will not be advertised into the area.
- When an area is defined as **nssa**, OSPF does not flood external routes from other areas into the area, but does translate and flood route information from the area into other areas, such as the backbone.

NOTE: You can assign one area per router port. If the router has 64 ports, 64 areas are supported on that router.

By default, the OSPF feature is disabled. OSPF must be enabled and active on the router for this command to be operational.

EXAMPLE:

To define a normal area, enter the following commands:

```
BigIron(config)# router ospf
BigIron(config-ospf-router)# area 192.53.0.0
```

EXAMPLE:

To define an area as a stub area, enter the following commands:

```
BigIron(config)# router ospf
BigIron(config-ospf-router)# area 192.53.0.0 stub 1
```

EXAMPLE:

To define an area as an NSSA, enter the following commands:

```
BigIron(config)# router ospf
BigIron(config-ospf-router)# area 192.53.0.0 nssa 1
```

EXAMPLE:

To disable summary LSAs for a stub area, enter commands such as the following:

```
BigIron(config-ospf-router)# area 40 stub 1 no-summary
```

Syntax: area <num> | <ip-addr> [stub <cost> [no-summary]]

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 **stub** <cost> parameter specifies that this is a stubby area. The <cost> 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.

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.

Possible values: Area ID: Valid IP address; Stub cost: 1 – 16777215

Default value: If you do not specify **stub** or **nssa**, a normal area is defined.

area <num> | <ip-addr> virtual-link <ip-addr>

Provides an area-border router a logical connection to the backbone area (0.0.0.0) when a physical connection to the backbone area does not exist.

The **area** <num> | <ip-addr> represents the shared area of the two area border routers—the one with a physical connection to the backbone and the router that requires a logical connection to the backbone. The defined area serves as the connection point between the two routers.

The **virtual-link** <ip-addr> is the Router ID of the router 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: When establishing an area virtual link, it must be configured on both of the routers (both ends of the virtual link).

By default, the OSPF feature is disabled. OSPF must be enabled and active on the router for this command to be operational. In addition, the neighbor area border router that has a physical connection to the backbone and the common area, must have connections to both the neighbor area border router and the area border router requiring a logical connection.

EXAMPLE:

To define the virtual link for area 195.22.0.0, enter the following:

```
BigIron(config)# router ospf
BigIron(config-ospf-router)# area 195.22.0.0 virtual 201.44.53.44
```

Other parameters that can be modified with this command, as seen in the syntax, are summarized below:

authentication-key: A password used to validate action

dead-interval: The number of seconds that a neighbor router will wait for a hello packet from the current router, before declaring the router down.

hello-interval: The length of time between the transmission of hello packets.

md5-authentication: The MD5 key-activation wait time, key ID, and key string.

retransmit-interval: The time between retransmits of link state advertisements to router adjacencies for this interface.

transmit-delay: The time it takes to transmit Link State Update packets on this interface

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>]

Possible values and Default values:

Parameter	Possible Values	Default
authentication-key	up to 8 alphanumeric characters	none
dead-interval	1 – 65535 seconds	40 seconds
hello-interval:	1 – 65535	10 seconds
md5-authentication key-activation-wait-time	0 – 14400	300 seconds (5 minutes)
md5-authentication key ID	1 – 255	none
md5-authentication key string	up to 16 alphanumeric characters	none
retransmit-interval	0 – 3600 seconds.	5 seconds
transmit-delay	0 – 3600	1

The optional **0 | 1** parameter with the **authentication-key** and **md5-authentication key-id** parameters affects encryption.

For added security, software release 07.1.10 and later encrypts 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 pasword 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.

area range

Assigns representative values to a range of IP addresses within an area, so that only those reference range addresses are advertised to the network, instead of all the addresses within that range.

NOTE: Range assignment is optional.

EXAMPLE:

```
BigIron(config)# router ospf

BigIron(config-ospf-router)# area 192.53.0.0 range 193.45.0.0 255.255.0.0
```

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**.

Possible values: See above

Default value: N/A

auto-cost reference-bandwidth

Changes the OSPF reference bandwidth used to calculate the default costs of 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

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.

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.

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.

EXAMPLE:

To change the reference bandwidth, enter a command such as the following at the OSPF configuration level of the CLI:

```
BigIron(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:

```
BigIron(config-ospf-router)# no auto-cost reference-bandwidth
```

Possible values: 1 – 4294967

Default value: 100

database-overflow-interval

Configures how often a router will check to see if the OSPF external link state database overflow condition has been eliminated by removal of entries originated on the router.

If the configured value of the data-base-overflow-interval is zero, then the router will never leave the database overflow condition. The default value for the database overflow interval is zero.

EXAMPLE:

```
BigIron(config-ospf-router)# data-base-overflow-interval 60
```

Syntax: database-overflow-interval <value>

Possible values: 0 – 86,400 seconds

Default value: 0

default-information-originate

Enables or disables origination of default routes.

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 **default-information-originate** command 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 “area” on page 11-1.

EXAMPLE:

To enable default route origination, enter the following command:

```
BigIron(config-ospf-router)# default-information-originate
```

To disable the feature, enter the following command:

```
BigIron(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.

Possible values: N/A

Default value: Disabled

default-metric

Sets the global default-metric value that will be adopted by all external routes imported into OSPF.

EXAMPLE:

To set a default metric of 1000 to be applied to all external routes imported into OSPF, enter the following command.

```
BigIron(config-ospf-router)# def 1000
```

Syntax: default-metric <value>

Possible values: 1 – 16,777,215

Default value: 10

deny redistribute

Defines the route(s) upon which you do not want to perform OSPF redistribution.

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).

EXAMPLE:

To deny redistribution on incoming routes received from the 192.95.0.0 network, enter the following:

```
BigIron(config-ospf-router)# deny redis 2 all 192.95.0.0 255.255.0.0
```

Syntax: deny redistribute <filter-num> all | bgp | connected | rip | static
[address <ip-addr> <ip-mask> [match-metric <value> [set-metric <value>]]]

Possible values: see below:

all	apply redistribution to all route types
bgp	apply redistribution to BGP4 routes only
connected	apply redistribution to directly-connected routes only
rip	apply redistribution to RIP routes only
static	apply redistribution to the static route only
ip address	network and sub-net addresses
match-metric	applies redistribution only to those incoming routes that match a specific metric value; Possible values: 1 – 15
set-metric	OSPF metric value that will be applied to all routes imported into OSPF

Default value: N/A

NOTE: If a **set-metric** value is not set using the set-metric parameter, then the value configured for the global parameter default-metric will be applied.

distance

Configures an administrative distance for a specific OSPF route type. 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 configure administrative distances for OSPF route types, use the following CLI method.

EXAMPLE:

To change the default administrative distances for inter-area routes, intra-area routes, and external routes, enter the following command:

```
BigIron(config-ospf-router)# distance external 100
BigIron(config-ospf-router)# distance inter-area 90
BigIron(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:

```
BigIron(config-ospf-router)# no distance external 100
```

Possible values: see above

Default value: 110

distribute-list

Configures a distribution list to explicitly deny specific routes from being eligible for installation in the IP route table. By default, all OSPF routes in the OSPF route table are 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.

EXAMPLE:

The following examples 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.

To use a standard ACL to configure an OSPF distribution list for denying specific routes, enter commands such as the following:

```
BigIron(config)# ip access-list standard no_ip
BigIron(config-std-nacl)# deny 4.0.0.0 0.255.255.255
BigIron(config-std-nacl)# permit any any
BigIron(config-std-nacl)# exit
BigIron(config)# router ospf
BigIron(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.

To use an extended ACL to configure an OSPF distribution list for denying specific routes, enter commands such as the following:

```
BigIron(config)# ip access-list extended no_ip
BigIron(config-ext-nacl)# deny ip 4.0.0.0 0.255.255.255 255.255.0.0 0.0.255.255
BigIron(config-ext-nacl)# permit ip any any
BigIron(config-ext-nacl)# exit
BigIron(config)# router ospf
BigIron(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] distribute-list <acl-name> | <acl-id> in

The <acl-name> | <acl-id> parameter specifies the ACL name or ID.

Possible values: See above

Default value: N/A

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.


```
BigIron(config-ospf-router)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
BigIron(config-ospf-router)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

external-lsdb-limit

Provides compliance with RFC 1765 in the handling of OSPF external link-state database (LSDB) overflow.

EXAMPLE:

To decrease this value to 1500 seconds, enter the following:

```
BigIron(config-ospf-router)# external-lsdb-limit 1500
```

Syntax: external-lsdb-limit <value>

Possible values: 0 – 3260643 (FastIron 4802); 0 – 6988913 (other Layer 3 Switches)

NOTE: If you specify 0, the software returns the parameter to its default value.

Default value: 3260643 (FastIron 4802); 06988913 (other Layer 3 Switches)

log

Specifies which kinds of OSPF-related Syslog messages are logged. In releases prior to 07.6.03, by default all OSPF Syslog messages are logged. In configurations with a large amount of OSPF activity, this can result in the Foundry device's Syslog buffer and the Syslog server filling up with OSPF messages.

Starting with release 07.6.03, 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.

EXAMPLE:

To specify that all OSPF-related Syslog messages be logged, enter the following commands.

```
BigIron(config)# router ospf
BigIron(config-ospf-router)# log all
```

Syntax: [no] log all | adjacency | bad_packet [checksum] | database | memory | retransmit

Possible values:

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.

Default value: N/A

maximum-number-of-lsa

Adjusts the table sizes for Link-State Advertisements (LSAs) on Layer 3 Switches with 32MB or more memory.

NOTE: This command is not supported in software release 07.1.00 and later. In these releases, OSPF dynamically allocates memory.

NOTE: You must save the configuration (**write memory**), then reload the software to place this command into effect.

NOTE: This feature applies only to Layer 3 Switches with 32MB memory or greater.

The following table lists the types of LSAs for which you can configure the table size, the default number of entries the tables can hold, and the range of maximum values you can specify. You cannot configure the LSA tables globally; you must configure them for individual LSA types.

LSA Type	Default Maximum Number of Entries	Range of Values
External	2000	500 – 8000
Network	2000	200 – 2000
Router	2200	200 – 2200
Summary	2000	500 – 8000 (NetIron) 500 – 18000 (NetIron Internet Backbone router, BigIron, and TurboIron/8)

EXAMPLE:

To change the maximum number of summary LSA entries from 2000 to 18000, enter the following commands:

```
BigIron(config-ospf-router)# maximum-number-of-lsa summary 18000
BigIron(config-ospf-router)# write memory
BigIron(config-ospf-router)# end
BigIron# reload
```

Syntax: maximum-number-of-lsa external | network | router | summary <value>

You must save the change to the startup-config file and reload or reboot. The change does not take effect until your reload or reboot.

NOTE: When you allocate high values for the LSA types, you reduce the amount of memory available to other protocols. If you need to run more than one memory-intensive protocol simultaneously, make sure you consider the memory needs of each. For example, if you are planning to run OSPF and BGP4 on the same Layer 3 Switch,

you might not want to allocate the maximum memory capacity for all the LSA types. Doing so could take memory away from BGP4 and thus limit configurable BGP4 table sizes such as the maximum number of routes the BGP route table can hold.

Possible values: see above

Default value: see above

max-routes

Changes the maximum number of OSPF routes the Layer 3 Switch can hold.

NOTE: This command is not supported in software release 07.1.00 and later. In these releases, OSPF dynamically allocates memory.

NOTE: You must save the configuration (**write memory**), then reload the software to place this command into effect.

EXAMPLE:

To change the maximum number of OSPF routes to 32000, enter the following command:

```
BigIron(config-ospf-router)# max-routes 32000
BigIron(config-ospf-router)# end
BigIron# reload
```

Syntax: `max-routes <num>`

Possible values: 4000 – 32000

Default value: 16000

metric-type

Specifies the type of OSPF metric to be used for routes imported into OSPF. Type 2 specifies a big metric (3 bytes). Type 1 specifies a small metric (2 bytes).

EXAMPLE:

```
BigIron(config-ospf-router)# metric-type type1
```

Syntax: metric-type type1 | type2

Possible values: type1, type2

Default value: type2

no

Disables other commands. To disable a command, place the word **no** before the command.

permit redistribute

Defines the route types upon which you want to perform OSPF redistribution.

OSPF must be enabled and active for this command to be operational. OSPF is disabled by default.

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).

EXAMPLE:

```
BigIron(config-ospf-router)# permit redis 1 rip address 192.147.72.0 255.255.255.0
```


Syntax: permit redistribute <filter-num> all | bgp | connected | rip | static [address <ip-addr> <ip-mask> [match-metric <value> [set-metric <value>]]]

Possible values:

all	apply redistribution to all route types
bgp	apply redistribution to BGP4 routes only
connected	apply redistribution to directly connected routes only
rip	apply redistribution to RIP routes only
static	apply redistribution to the static route only
ip address	network and sub-net addresses
match-metric	match a specific metric value; Possible values: are 1 – 16777215
set-metric	OSPF metric value that will be applied to all routes imported into OSPF

Default value: N/A

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-ospf-router)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

redistribution

Enables the OSPF route redistribution function on the Layer 3 Switch. When enabled, OSPF will import external routes into the OSPF domain. The Layer 3 Switch will behave as an Autonomous System Boundary Router (ASBR). You can select the types of routes you want to redistribute for each protocol.

EXAMPLE:

To enable redistribution of RIP routes into OSPF:

```
BigIron(config-ospf-router)# redistribution rip
```

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.

Possible values: See above

Default value: disabled

rfc1583-compatibility

Foundry Layer 3 Switches are configured by default to be compliant with RFC 1583 OSPF V2 specification. Routers can also be configured to operate with the OSPF standard RFC 2178 by entering the **no rfc1583-compatibility** command.

EXAMPLE:

```
BigIron(config-ospf-router)# no rfc1583-compatibility
```

Syntax: [no] rfc1583-compatibility

Possible values: N/A

Default value: enabled

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

summary-address

Configures 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. 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. See the "Configuring OSPF" chapter of the *Foundry Enterprise Configuration and Management Guide*.

EXAMPLE:

To configure a summary address for OSPF routes, enter commands such as the following:

```
BigIron(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:

```
BigIron(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

Possible values: see above

Default value: no summarization

timers lsa-group-pacing

Changes the LSA pacing interval. 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.

EXAMPLE:

To change the LSA pacing interval to two minutes (120 seconds), enter the following command:

```
BigIron(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:

```
BigIron(config-ospf-router)# no timers lsa-group-pacing
```

Possible values: 10 – 1800 seconds (30 minutes)

Default value: 240 seconds (four minutes)

timers spf

Changes the Shortest Path First (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, use the following CLI method.

EXAMPLE:

To change the SPF delay and hold time, enter commands such as the following:

```
BigIron(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:

```
BigIron(config-ospf-router)# no timers spf 10 20
```

Possible values: see above

Default value: delay 5 seconds; hold time 10 seconds

trap

Generation of OSPF traps is enabled, by default, on the router when OSPF is enabled. To disable all traps, use the global level CONFIG command **no snmp-server trap ospf**.

To stop a specific OSPF trap from being collected, use the CLI command **no trap** <trap>.

EXAMPLE:

To stop changes in the state of neighbors being generated by a router, enter the following command:

```
BigIron(config-ospf-router)# no trap neighbor-state-change-trap
```

To reinstate the command, enter the following command:

```
BigIron(config-ospf-router)# trap neighbor-state-change-trap
```

Syntax: [no] trap <trap>

Possible values: see below: traps are from RFC 1850

Default value: All traps are active when OSPF is enabled.

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: ospfIfConfigError]
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]

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-bgp-router)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-bgp-router)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 12
IS-IS Commands

area-password

Configures an IS-IS area password.

EXAMPLE:

To configure an IS-IS area password, enter a command such as the following:

```
NetIron(config-isis-router)# area-password area-51
```

This command configures the Foundry device to use the password “area-51” to authenticate Level-1 LSPDUs.

Syntax: [no] area-password <string>

The <string> parameter specifies the password. You can enter an alphanumeric string up to 80 characters long. The password can contain blank spaces. If you use a blank space in the password, you must use quotation marks (“ “) around the entire password; for example, **area-password “area 51”**.

Possible values: See above

Default value: None configured

csnp-interval

Changes the Complete Sequence Numbers PDU (CSNP) and Partial Sequence Numbers PDU (PSNP) interval.

The CSNP interval specifies how often the Designated IS sends a CSNP to the broadcast interface. Likewise, the PSNP interval specifies how often other ISs (non-Designated ISs) send a PSNP to the broadcast interface. (The PSNP interval also applies to ISs on a point-to-point network.)

The interval applies to both Level-1 and Level-2 CSNPs and PSNPs.

NOTE: Although the command name is **csnp-interval**, the interval also applies to PSNPs.

EXAMPLE:

To change the interval, enter a command such as the following:

```
NetIron(config-isis-router)# csnp-interval 15
```

Syntax: [no] csnp-interval <secs>

The <secs> parameter specifies the interval and can be from 0 – 65535 seconds. The default is 10 seconds.

Possible values: 0 – 65535 seconds

Default value: 10 seconds

default-information-originate

Enables advertisement of a default route.

By default, the Foundry device does not generate or advertise a default route to its neighboring ISs. This is true even if the device's IP route table contains a default route. You can enable the Foundry device to advertise a default route to all neighboring ISs using one of the following methods. By default, the feature originates the default route at Level-2 only. However, you can apply a route map to originate the default route to Level-1 only or at both Level-1 and Level-2.

NOTE: This feature requires the presence of a default route in the IP route table.

EXAMPLE:

To enable the Layer 3 Switch to advertise a default route, enter the following command:

```
NetIron(config-isis-router)# default-information-originate
```

This command enables the device to advertise a default route into the IS-IS area to which the device is attached.

Syntax: [no] default-information-originate [route-map <name>]

The **route-map** <name> parameter allows you to specify the level to which to advertise the default route. You can specify one of the following:

- Advertise to Level-1 ISs only.
- Advertise to Level-2 ISs only.
- Advertise to Level-1 and Level-2 ISs.

NOTE: The route map must be configured before you can use the route map as a parameter with the **default-information-originate** command.

To use a route map to specify the level on which you want the Layer 3 Switch to advertise a default route, change the CLI to the global CONFIG level, then enter commands such as the following:

```
NetIron(config)# route-map DarkAngel permit 1
NetIron(config-routemap DarkAngel)# set level level-1
NetIron(config-routemap DarkAngel)# router isis
NetIron(config-isis-router)# default-information-originate route-map DarkAngel
```

These commands configure a route map to set the default advertisement level to Level-1 only.

Possible values: See above

Default value: Disabled

default-metric

Changes the default metric.

When IS-IS redistributes a route from another route source (such as OSPF, BGP4, or an IP static route) into IS-IS, IS-IS uses the route's metric value as its IS-IS metric. However, if the route does not have a valid metric, IS-IS instead applies a default metric to the route.

NOTE: By default, the Layer 3 Switch supports metric values from 1 – 63. To use a higher metric value, you must change the metric style to "wide". See "metric-style" on page 12-7.

NOTE: The Foundry implementation of IS-IS does not support the optional metric types Delay, Expense, or Error.

EXAMPLE:

To change the default metric, enter a command such as the following:


```
NetIron(config-isis-router)# default-metric 20
```

Syntax: [no] default-metric <num>

The <num> parameter specifies the default metric. You can specify a value from 1 – 65535. The default is 10.

Possible values: 1 – 65535

Default value: 10

distance

Changes the administrative distance if IS-IS routes.

When the Layer 3 Switch has paths from multiple routing protocols to the same destination, the Layer 3 Switch compares the administrative distances of the paths and selects the path with the lowest administrative distance to place in the IP route table.

For example, if the Layer 3 Switch has a path from RIP, from OSPF, and a path from IS-IS to the same destination, and all the paths are using their protocols' default administrative distances, the Layer 3 Switch selects the OSPF path, because that path has a lower administrative distance than the RIP and IS-IS paths.

Lower administrative distances are preferred over higher distances. For example, if the Layer 3 Switch receives routes for the same network from IS-IS (distance 115) and from RIP (distance 120), the Layer 3 Switch will prefer the IS-IS route by default.

EXAMPLE:

To change the administrative distance for IS-IS routes, enter a command such as the following:

```
NetIron(config-isis-router)# distance 100
```

Syntax: [no] distance <num>

This command changes the administrative distance for all IS-IS routes to 100.

The <num> parameter specifies the administrative distance. You can specify a value from 1 – 255. The default for IS-IS is 115.

Possible values: 1 – 255

Default value: 115

domain-password

Configures an IS-IS domain password.

EXAMPLE:

To configure an IS-IS domain password, enter a command such as the following:

```
NetIron(config-isis-router)# domain-password domain-1
```

This command configures the Foundry device to use the password "domain-1" to authenticate Level-2 LSPDUs.

Syntax: [no] domain-password <string>

The <string> parameter specifies the password. You can enter an alphanumeric string up to 80 characters long. The password can contain blank spaces. If you use a blank space in the password, you must use quotation marks (" ") around the entire password; for example, **domain-password "domain 1"**.

Possible values: See above

Default value: None configured

hello-interval

Changes the IS-IS hello interval.

The hello interval controls how often the Layer 3 Switch sends hello messages to its IS-IS neighbors. The default interval is 10 seconds for Level-1 and Level-2. You can change the hello interval for one or both levels to a value from 1 – 65535 seconds.

EXAMPLE:

To change the hello interval, enter a command such as the following:

```
NetIron(config-isis-router)# hello-interval 20
```

This command changes the hello interval to 20 seconds. By default, the change applies to both Level-1 and Level-2.

Syntax: [no] hello-interval <num> [level-1 | level-2-only]

The <num> parameter specifies the interval, and can be from 1 – 65535 seconds. The default is 10 seconds.

The **level-1 | level-2-only** parameter applies the change to only the level you specify. If you do not use this parameter, the change applies to both levels.

Possible values: See above

Default value: 10 seconds

hello-multiplier

Changes the IS-IS hello multiplier.

The hello multiplier is the number by which the Layer 3 Switch multiplies the hello interval to obtain the hold time for Level-1 and Level-2 IS-to-IS hello PDUs. The default multiplier is 3. You can set the multiplier to a value from 1 – 2147483647.

EXAMPLE:

To change the hello multiplier, enter a command such as the following:

```
NetIron(config-isis-router)# hello-multiplier 50
```

This command changes the hello interval to 50. By default, the change applies to both Level-1 and Level-2.

Syntax: [no] hello-multiplier <num> [level-1 | level-2-only]

The <num> parameter specifies the multiplier, and can be from 1 – 2147483647. The default is 3.

The **level-1 | level-2-only** parameter applies the change to only the level you specify. If you do not use this parameter, the change applies to both levels.

Possible values: See above

Default value: 3

hello padding

Disables IS-IS padding.

By default, the Layer 3 Switch adds extra data to the end of a hello packet to make the packet the same size as the maximum length of PDU the Layer 3 Switch supports.

The padding applies to the following types of hello packets:

- ES hello (ESH PDU)
- IS hello (ISH PDU)
- IS to IS hello (IIH PDU)

The padding consists of arbitrarily valued octets. A padded hello PDU indicates the largest PDU that the Layer 3 Switch can receive. Other ISs that receive a padded hello PDU from the Layer 3 Switch can therefore ensure that the IS-IS PDUs they send the Layer 3 Switch. Similarly, if the Layer 3 Switch receives a padded hello PDU from a neighbor IS, the Layer 3 Switch knows the maximum size PDU that the Layer 3 Switch can send to the neighbor.

When padding is enabled, the maximum length of a Hello PDU sent by the Layer 3 Switch is 1514 bytes.

If you need to disable padding, you can do so globally or on individual interfaces. Generally, you do not need to disable padding unless a link is experiencing slow performance, for example due to point-to-point interoperability issues. If you enable or disable padding on an interface, the interface setting overrides the global setting.

By default, disabling or re-enabling padding affects hello PDUs sent on point-to-point circuits and to an IS-IS broadcast address. You can specify an option to enable or disable the padding for point-to-point or broadcast PDUs.

NOTE: To disable or re-enable padding globally, see “hello padding” on page 7-16.

EXAMPLE:

```
NetIron(config-isis-router)# interface ethernet 1/1
NetIron(config-if-1/1)# hello padding
```

Syntax: [no] hello padding [point-to-point]

The **point-to-point** parameter disables or re-enables the padding only for point-to-point connections.

Possible values: See above

Default value: Enabled

hostname

Disables ro re-enables mapping of IS-IS system IDs to the hostnames of the devices that have those IDs.

Foundry’s implementation of IS-IS supports RFC 2763, which describes a mechanism for mapping IS-IS system IDs to the hostnames of the devices with those IDs. For example, if you set the hostname on the Layer 3 Switch to “IS-IS Router 1”, the mapping feature uses this name instead of the Layer 3 Switch’s IS-IS system ID in the output of the following commands:

- **show isis database**
- **show isis interface**
- **show isis neighbor**

The Layer 3 Switch’s hostname is displayed in each CLI command prompt, for example:

```
IS-IS Router 1(config-isis-router)#
```

EXAMPLE:

If you want to disable name mapping, enter the following command:

```
IS-IS Router 1(config-isis-router)# no hostname
```

Syntax: [no] hostname

To display the name mappings, enter the **show isis hostname** command. See “show isis hostname” on page 43-89.

Possible values: N/A

Default value: Enabled

is-type

Changes the type of IS-IS packets (Level-1 or Level-2) supported on the device.

NOTE: To change the type on an individual interface only, see “isis circuit-type” on page 7-52.

EXAMPLE:

To globally change the type of IS-IS packets supported on the device from Level-1 and Level-2 to Level-1 only, enter the following command:

```
NetIron(config-isis-router)# is-type level-1
```

Syntax: [no] is-type level-1 | level-1-2 | level-2-only

The **level-1 | level-1-2 | level-2-only** parameter specifies the IS-IS type. If you want to re-enable support for both IS-IS types, re-enter the command you entered to change the IS-IS type, and use “no” in front of the command.

For example, to reverse the command shown above an re-enable support for both IS-IS types, enter the following command:

```
NetIron(config-isis-router)# no is-type level-1
```

Possible values: See above

Default value: Both levels are supported

log-adjacency-changes

Enables logging of the adjacency changes. The Layer 3 Switch can generate a Syslog entry and an SNMP trap to indicate a change in the status of an adjacency with another IS.

EXAMPLE:

To enable logging of adjacency changes, enter the following command:

```
NetIron(config-isis-router)# log-adjacency-changes
```

Syntax: [no] log-adjacency-changes

To disable logging of adjacency changes, enter the following command:

```
NetIron(config-isis-router)# no log-adjacency-changes
```

Possible values: N/A

Default value: Disabled

lsp-gen-interval

Changes the LSP general interval. The LSP general interval is the minimum number of seconds the Layer 3 Switch waits between sending updated LSPs to its IS-IS neighbors.

EXAMPLE:

To change the LSP general interval to 45 seconds, enter a command such as the following:

```
NetIron(config-isis-router)# lsp-gen-interval 45
```

Syntax: [no] lsp-gen-interval <secs>

The <secs> parameter specifies the minimum refresh interval and can be from 1 – 120 seconds. The default is 10 seconds.

Possible values: 1 – 120 seconds

Default value: 10 seconds

lsp-refresh-interval

Changes the LSP refresh interval. The LSP refresh interval is the maximum number of seconds the Layer 3 Switch waits between sending updated LSPs to its IS-IS neighbors.

EXAMPLE:

To change the LSP refresh interval to 20000 seconds, enter a command such as the following:

```
NetIron(config-isis-router)# lsp-refresh-interval 20000
```

Syntax: [no] lsp-refresh-interval <secs>

The <secs> parameter specifies the maximum refresh interval and can be from 1 – 65535 seconds. The default is 900 seconds (15 minutes).

Possible values: 1 – 65535 seconds

Default value: 900 seconds (15 minutes)

maximum-paths

Changes the maximum number of route paths the device can have for an IS-IS route.

When the Layer 3 Switch has multiple IS-IS equal-cost paths to the same destination, the Layer 3 Switch can load share among the paths. For example, if the device has three IS-IS paths to the same destination and each path has the same default metric, the Layer 3 Switch alternates among the three paths when forwarding traffic.

By default, IS-IS load sharing is enabled for four paths. You can change the number of paths to an amount from 1 – 8. If you change the number of paths to 1, the Layer 3 Switch does not load share route paths learned from IS-IS.

NOTE: IS-IS load sharing requires IP load sharing to be enabled for at least the number of paths you want to use for IS-IS load sharing. IP load sharing is enabled for four paths by default. The IP load sharing settings affect all routing protocols that support load sharing, including IS-IS, OSPF and BGP4.

EXAMPLE:

To change the maximum number of IS-IS load sharing paths, enter a command such as the following:

```
NetIron(config-isis-router)# maximum-paths 6
```

Syntax: [no] maximum-paths <num>

The <num> parameter specifies the maximum number of load sharing paths and can be from 1 – 8. The default is 4.

Possible values: 1 – 8

Default value: 4

max-lsp-lifetime

Changes the maximum LSP lifetime. The maximum LSP lifetime is the maximum number of seconds an unrefreshed LSP can remain in the Layer 3 Switch's LSP database.

EXAMPLE:

To change the maximum LSP lifetime to 2400 seconds, enter a command such as the following:

```
NetIron(config-isis-router)# max-lsp-lifetime 2400
```

Syntax: [no] max-lsp-lifetime <secs>

The <secs> parameter specifies the maximum LSP lifetime and can be from 1 – 65535 seconds. The default is 1200 seconds (20 minutes).

Possible values: 1200 seconds (20 minutes)

Default value: narrow

metric-style

Changes the metric style for one or both IS-IS levels (Level-1 and Level-2).

The metric style specifies the Types, Lengths, and Values (TLVs) an IS-IS LSP can have. The TLVs specify the types of data, the maximum length of the data, and the valid values for the data. One of the types of data the TLVs control is a route's default-metric. By default, the Layer 3 Switch uses the standard IS-IS TLVs, which allows metric values from 1 – 63. The default metric style is called "narrow". You can increase the range of metric values supported by the Layer 3 Switch by changing the metric style to wide. The wide metric style allows metric values from 1 – 16777215.

EXAMPLE:

To change the metric style to wide, enter the following command:

```
NetIron(config-isis-router)# metric-style wide
```

This command changes the metric style for both Level-1 and Level-2.

Syntax: [no] metric-style wide [level-1 | level-2-only]

The **level-1 | level-1-2 | level-2-only** parameter specifies the level(s) to which the change applies.

Possible values: See above

Default value: narrow

net <area-id>.<system-id>.<sel>

Configures an IS-IS Network Entity Title (NET).

EXAMPLE:

```
NetIron(config-isis-router)# net 49.2211.aaaa.bbbb.cccc.00
```

The command in this example configures a NET that has the area ID 49.2211, the system ID aaaa.bbbb.cccc (the Layer 3 Switch's base MAC address), and SEL value 00.

Syntax: [no] net <area-id>.<system-id>.<sel>

The <area-id> parameter specifies the area and has the format xx or xx.xxxx. For example, 49 and 49.2211 are valid area IDs.

The <system-id> parameter specifies the Layer 3 Switch's unique IS-IS router ID and has the format xxxx.xxxx.xxxx. You can specify any value for the system ID. A common practice is to use the device's base MAC address as the system ID. The base MAC address is also the MAC address of port 1/1. To determine the base MAC address, enter the following command at any level of the CLI: **show interfaces brief**. The base MAC address is listed in the first row of information, in the MAC column.

You must use the same system ID in all the NETs on the Layer 3 Switch.

NOTE: The parameter descriptions above are the recommended values for the NET. However, the CLI accepts any value that fits within the following lengths and formats:

xx.xxxx.xxxx.xxxx.00 – minimum length of NET

xx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.xxxx.00 – maximum length of NET

The <sel> parameter specifies the NSAP Selector (SEL). This value must always be 00 (two zeros). The value 00 indicates that this address is an NET.

Possible values: See above

Default value: None configured

redistribute bgp

Enables redistribution of BGP4 routes into IS-IS.

EXAMPLE:

```
NetIron(config-isis-router)# redistribute bgp
```

This command configures the Layer 3 Switch to redistribute all its BGP4 routes into Level-2 IS-IS.

Syntax: [no] redistribute bgp [level-1 | level-1-2 | level-2 | metric <num> |
metric-type external | internal |
route-map <name>]

Possible values: The parameters are the same as the parameters for the **redistribute static** command.

Default value: Not configured

redistribute connected

Enables redistribution of directly connected IP routes into IS-IS.

EXAMPLE:

```
NetIron(config-isis-router)# redistribute connected
```

This command configures the Layer 3 Switch to redistribute all directly connected routes in the IP route table into Level-2 IS-IS.

Syntax: [no] redistribute connected [level-1 | level-1-2 | level-2 | metric <num> |
metric-type external | internal |
route-map <name>]

Possible values: The parameters are the same as the parameters for the **redistribute static** command.

Default value: Not configured

redistribute isis

Enables redistribution of routes from one IS-IS level to the other.

NOTE: By default, the Layer 3 Switch redistributes routes from Level-1 into Level-2. The Layer 3 Switch automatically redistributes Level-1 routes into Level-2 routes, even if you do not enable redistribution.

EXAMPLE:

```
NetIron(config-isis-router)# redistribute isis level-2 into level-1
```

This command redistributes the Level-2 routes into Level-1. The Layer 3 Switch automatically redistributes Level-1 routes into Level-2.

Syntax: [no] redistribute isis level-1 into level-2 | level-2 into level-1

The **level-1 into level-2 | level-2 into level-1** parameter specifies the direction of the redistribution:

- **level-1 into level-2** – Redistributes Level-1 routes into Level-2. This is the default.
- **level-2 into level-1** – Redistributes Level-2 routes into Level-1.

Possible values: The parameters are the same as the parameters for the **redistribute static** command.

Default value: Level-1 routes are redistributed into Level-2

redistribute ospf

Enables redistribution of OSPF routes into IS-IS.

EXAMPLE:

```
NetIron(config-isis-router)# redistribute ospf
```

This command configures the Layer 3 Switch to redistribute all OSPF routes into Level-2 IS-IS.

Syntax: [no] redistribute ospf [level-1 | level-1-2 | level-2 |
match external1 | external2 | internal
metric <num> |
metric-type external | internal |
route-map <name>]

Most of the parameters are the same as the parameters for the **redistribute static** command. However, the **redistribute ospf** command also has the **match external1 | external2 | internal** parameter. This parameter specifies the OSPF route type you want to redistribute into IS-IS. By default, the **redistribute ospf** command redistributes only internal routes.

- **external1** – An OSPF type 1 external route.
- **external2** – An OSPF type 2 external route.
- **internal** – An internal route calculated by OSPF.

Possible values: See above

Default value: Internal routes are redistributed

redistribute rip

Enables redistribution of RIP routes into IS-IS.

EXAMPLE:

```
NetIron(config-isis-router)# redistribute rip
```


This command configures the Layer 3 Switch to redistribute all RIP routes into Level-2 IS-IS.

Syntax: [no] redistribute rip [level-1 | level-1-2 | level-2 | metric <num> |
metric-type external | internal |
route-map <name>]

The parameters are the same as the parameters for the **redistribute static** command.

Possible values: The parameters are the same as the parameters for the **redistribute static** command.

Default value: Not configured

redistribute static

Enables redistribution of static routes from the IP route table into IS-IS.

EXAMPLE:

```
NetIron(config-isis-router)# redistribute static
```

This command configures the Layer 3 Switch to redistribute all IP static routes into Level-2 IS-IS routes.

Syntax: [no] redistribute static [level-1 | level-1-2 | level-2 | metric <num> |
metric-type external | internal |
route-map <name>]

The **level-1**, **level-1-2**, and **level-2** parameters restrict redistribution to the specified IS-IS level.

The **metric** <num> parameter restricts the redistribution to only those routes that have the metric you specify.

The **metric-type external | internal** parameter restricts redistribution to one of the following:

- **external** – The metric value is not comparable to an IS-IS internal metric and is always higher than the IS-IS internal metric.
- **internal** – The metric value is comparable to metric values used by IS-IS. This is the default.

The **route-map** <name> parameter restricts distribution to those routes that match the specified route map. The route map must already be configured before you use the route map name with the **redistribute** command. For example, to configure a route map that redistributes only the static IP routes to the destination network 2.4.69.x, enter commands such as the following:

```
NetIron(config)# access-list 101 permit ip any 2.4.69.0 255.255.255.0
NetIron(config)# route-map Alba permit 1
NetIron(config-routemap DarkAngel)# match ip address 101
NetIron(config-routemap DarkAngel)# router isis
NetIron(config-isis-router)# redistribute static route-map Alba
```

Possible values: See above

Default value: Not configured

set-overload-bit

Sets the overload bit on.

If an IS's resources are overloaded, preventing the IS from properly performing IS-IS routing, the IS can inform other ISs of this condition by setting the overload bit in LSPDUs sent to other ISs from 0 (off) to 1 (on).

When an IS is overloaded, other ISs will not use the overloaded IS to forward traffic. An IS can be in the overload state for Level-1, Level-2, or both.

- If an IS is in the overload state for Level-1, other Level-1 ISs stop using the overloaded IS to forward Level-1 traffic. However, the IS can still forward Level-2 traffic, if applicable.
- If an IS is in the overload state for Level-2, other Level-2 ISs stop using the overloaded IS to forward Level-2 traffic. However, the IS can still forward Level-1 traffic, if applicable.
- If an IS is in the overload state for both levels, the IS cannot forward traffic at either level.

By default, the Layer 3 Switch automatically sets the overload bit to 1 (on) in its LSPDUs to other ISs if an overload condition occurs.

You can set the overload bit on to administratively shut down IS-IS without disabling the protocol. Setting the overload bit on is useful when you want to make configuration changes without removing the Layer 3 Switch from the network.

In addition, you can configure the Layer 3 Switch to set the overload bit on for a specific number of seconds during startup, to allow IS-IS to become fully active before the device begins IS-IS routing. By default, there is no delay (0 seconds).

EXAMPLE:

To immediately set the overload bit on, enter the following command:

```
NetIron(config-isis-router)# set-overload-bit
```

This command administratively shuts down IS-IS by configuring the Layer 3 Switch to immediately set the overload bit to 1 (on) in all LSPs sent to other ISs.

To configure the Layer 3 Switch to temporarily set the overload bit on after a software reload, enter a command such as the following:

```
NetIron(config-isis-router)# set-overload-bit on-startup 5
```

This command configures the Layer 3 Switch to set the overload bit on in all IS-IS LSPs sent to other ISs during the first five seconds following a successful software reload. After the five seconds expire, the Layer 3 Switch stops setting the overload bit on, and instead starts setting the overload bit off.

Syntax: [no] set-overload-bit [on-startup <secs>]

The **on-startup** <secs> parameter specifies the number of seconds following a reload to set the overload bit on. You can specify 0 or a number from 5 – 86400 (24 hours). The default is 0, which means the Layer 3 Switch starts performing IS-IS routing immediately following a successful software reload.

Possible values: See above

Default value: Off

spf-interval

Changes the SPF timer.

Every IS maintains a Shortest Path First (SPF) tree, which is a representation of the states of each of the IS's links to ESs and other ISs. If the IS is both a Level-1 and Level-2 IS, it maintains separate SPF trees for each level.

To ensure that the SPF tree remains current, the IS updates the tree at regular intervals following a change in network topology or the link state database.

EXAMPLE:

To change the SPF interval, enter a command such as the following:

```
NetIron(config-isis-router)# spf-interval 30
```

Syntax: [no] spf-interval <secs>

The <secs> parameter specifies the interval and can be from 1 – 120 seconds. The default is 5 seconds.

Possible values: 1 – 120 seconds

Default value: 5 seconds

summary-address

Configures a summary address to aggregate IS-IS route information. Summary addresses can enhance performance by reducing the size of the Link State database, reducing the amount of data the Layer 3 Switch needs to send to its neighbors, and reducing the CPU cycles used for IS-IS.

When you configure a summary address, the address applies only to Level-2 routes by default. You can specify Level-1 only, Level-2 only, or Level-1 and Level-2 when you configure the address.

EXAMPLE:

To configure a summary address, enter a command such as the following:

```
NetIron(config-isis-router)# summary-address 192.168.0.0 255.255.0.0
```

This command configures a summary address for all Level-2 IS-IS route destinations between 192.168.1.0 – 192.168.255.255.

Syntax: [no] summary-address <ip-addr> <ip-mask> [level-1 | level-1-2 | level-2-only]

The <ip-addr> <ip-mask> parameters specify the aggregate address. The mask indicates the significant bits in the address. Ones are significant, and zeros allow any value. In the command example above, the mask 255.255.0.0 matches on all addresses that begin with 192.168 and contain any values for the final two octets.

The **level-1 | level-1-2 | level-2-only** parameter specifies the route types to which the aggregate route applies. The default is **level-2-only**.

Possible values: See above

Default value: None configured

Chapter 13
BGP4 Commands

address-filter

Configures an address filter for filtering routes in BGP4 updates based on IP address.

EXAMPLE:

To define an IP address filter to deny routes to 209.157.0.0, enter the following command:

```
BigIron(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.

Possible values: see above

Default value: N/A

aggregate-address

Configures the Layer 3 Switch to aggregate routes in a range of networks into a single CIDR number.

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.

EXAMPLE:

To aggregate routes for 209.157.22.0, 209.157.23.0, and 209.157.24.0, enter the following command:

```
BigIron(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 the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide* for information on defining a route map.

Possible values: see above

Default value: N/A

always-compare-med

Configures the Layer 3 Switch to compare the MEDs for all paths for a route, regardless of the AS through which the paths pass.

EXAMPLE:

To configure the router to always compare MEDs, enter the following command:

```
BigIron(config-bgp-router)# always-compare-med
```

Syntax: [no] always-compare-med

Possible values: N/A

Default value: Disabled

as-path-filter

Configures an AS-path filter for filtering routes in BGP4 updates based on AS-path.

EXAMPLE:

To define AS-path filter 4 to permit AS 2500, enter the following command:

```
BigIron(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 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.

NOTE: You can use regular expressions as part of the AS-path. See the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

Possible values: N/A

Default value: Disabled

as-path-ignore

Disables AS-Path comparison, which is Step 5 in the algorithm BGP4 uses to select the next path for a route.

EXAMPLE:

```
BigIron(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 skips from Step 4 to Step 6.

For information about the algorithm, see the "How BGP4 Selects a Path for a Route" section in the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

Syntax: [no] as-path-ignore

Possible values: N/A

Default value: Enabled

auto-summary

Enables or disables auto summary. The auto summary feature summarizes the routes it redistributes from IGP to BGP4. The router summarizes sub-nets into their natural class A, B, or C networks. For example, if an AS contains sub-nets 1.1.0.0, 1.2.0.0, and 1.3.0.0 with the network mask 255.255.0.0, the auto summary feature summarizes the sub-nets in its advertisements to BGP4 neighbors as 1.0.0.0/8. The auto summary feature is disabled by default.

NOTE: The auto summary feature summarizes only the routes that are redistributed from IGP into BGP4.

NOTE: The auto summary feature does not summarize networks that use CIDR numbers instead of class A, B, or C numbers. To summarize CIDR networks, use the aggregation feature. See the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

EXAMPLE:

To enable auto summary, enter the following command:

```
BigIron(config-bgp-router)# auto-summary
```

Syntax: [no] auto-summary

Possible values: N/A

Default value: Disabled

bgp-redistribute-internal

Enables redistribution of IBGP routes from BGP4 into RIP or OSPF.

EXAMPLE:

To enable the Layer 3 Switch to redistribute BGP4 routes into OSPF and RIP, enter the following command:

```
BigIron(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:

```
BigIron(config-bgp-router)# no bgp-redistribute-internal
```

Possible values: N/A

Default value: Disabled

client-to-client-reflection

Disables or re-enables route reflection. For more information about route reflection, see the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

EXAMPLE:

If you need to disable route reflection on a router, enter the following command. Disabling route reflection allows you to turn off the feature without removing Cluster ID and route reflector client information from the system configuration file.

```
BigIron(config-bgp-router)# no client-to-client-reflection
```

Enter the following command to re-enable the feature:

```
BigIron(config-bgp-router)# client-to-client-reflection
```

Syntax: [no] client-to-client-reflection

Possible values: N/A

Default value: Enabled

cluster-id

Changes the BGP4 cluster ID. Use this command only on a BGP4 Layer 3 Switch that you are using as a route reflector. For more information about route reflection, see the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

EXAMPLE:

Enter the following commands to configure a Foundry Layer 3 Switch as a route reflector for two neighbors in cluster 1:


```
BigIron(config-bgp-router)# cluster-id 1
BigIron(config-bgp-router)# neighbor 10.0.1.0 route-reflector-client
BigIron(config-bgp-router)# neighbor 10.0.2.0 route-reflector-client
```

Syntax: [no] cluster-id <num> | <ip-addr>

Possible values: a number from 1 – 4294967295 or an IP address. You can configure one cluster ID on the router. All route-reflector clients for the router are members of the cluster.

Default value: the router ID

community-filter

Configures a community address filter for filtering routes in BGP4 updates based on community.

EXAMPLE:

To define filter 3 to permit routes that have the NO_ADVERTISE community, enter the following command:

```
BigIron(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 <community-number> parameter indicates a specific community number to filter. Use this parameter to filter for a private (administrator-defined) community. If you want to filter for the well-known communities "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** parameter specifies a community value. If a BGP4 route received by the Layer 3 Switch from a peer has the community type LOCAL_AS, the Layer 3 Switch advertises the route only within the sub-AS. If the route has the community type NO_EXPORT, the Layer 3 Switch advertises the route only within the confederation.

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.

Possible values: N/A

Default value: Disabled

compare-routerid

Enables Router ID comparison, which 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 default in software release 07.5.02. In previous releases, router ID comparison is enabled by default and cannot be disabled.

EXAMPLE:

To enable router ID comparison, enter the following command at the BGP configuration level of the CLI:

```
BigIron(config-bgp-router)# compare-routerid
```

Syntax: [no] compare-routerid

For more information, see the "How BGP4 Selects a Path for a Route" section in the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

Possible values: N/A

Default value: Disabled

confederation

Configures a Layer 3 Switch to be a member of a BGP confederation.

EXAMPLE:

To configure a Layer 3 Switch to be a member of confederation 10, consisting of two sub-ASs (64512 and 64513):

```
BigIronA(config-bgp-router)# confederation identifier 10
BigIronA(config-bgp-router)# confederation peers 64512 64513
```

Syntax: confederation identifier <num>

Syntax: confederation peers <num> [<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.

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.

Possible values: 1 – 65535

Default value: N/A

dampening

Configures route flap dampening.

This section shows how to globally configure dampening. You also can use route maps to configure dampening for specify neighbors and routes.

EXAMPLE:

The following example shows how to change the dampening parameters.

```
BigIron(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.

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.

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.

Possible values: See above

Default value: Disabled

default-information-originate

Enables the Layer 3 Switch to advertise a default BGP4 route.

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.

EXAMPLE:

To enable the router to advertise a default BGP4 route, enter the following command:

```
BigIron(config-bgp-router)# default-information-originate
```

Syntax: [no] default-information-originate

Possible values: N/A

Default value: Enabled

default-local-preference

Changes the local preference. The local preference is an attribute that indicates a degree of preference for a route relative to other routes in the local AS. 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.

EXAMPLE:

To change the default local preference to 200, enter the following command:

```
BigIron(config-bgp-router)# default-local-preference 200
```

Syntax: default-local-preference <num>

Possible values: 0 – 4294967295

Default value: 100

default-metric

Sets the default BGP4 MED (metric), 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.

EXAMPLE:

To change the default metric to 40, enter the following command:

```
BigIron(config-bgp-router)# default-metric 40
```

Syntax: default-metric <num>

Possible values: 0 – 4294967295

Default value: 0

distance

Changes the administrative distance for IBGP, EBGP, or Local BGP routes. 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.

See the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide* for a list of the default administrative distances for all types of routes.

EXAMPLE:

To change the default administrative distances for EBGP, IBGP, and Local BGP, enter a command such as the following:

```
BigIron(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 default is 20.

The <internal-distance> sets the IBGP distance and can be a value from 1 – 255. The default is 200.

The <local-distance> sets the Local BGP distance and can be a value from 1 – 255. The default is 200.

Possible values: see above

Default value: see above

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
BigIron(config-bgp-router)# end

BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

enforce-first-as

Requires 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. By default, the Foundry device does not require this.

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.

EXAMPLE:

```
BigIron(config-bgp-router)# enforce-first-as
```

Syntax: [no] enforce-first-as

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
BigIron(config-bgp-router)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

fast-external-fallover

Configures the Layer 3 Switch to immediately close the BGP4 session and TCP connection to locally attached neighbors that die. This feature is disabled by default.

EXAMPLE:

To enable fast external fallover, enter the following command:

```
BigIron(config-bgp-router)# fast-external-fallover
```

Syntax: [no] fast-external-fallover

Possible values: N/A

Default value: Disabled

local-as

The local AS number identifies the AS the Foundry BGP4 router is in. The AS number can be from 1 – 65535. AS numbers 64512 – 65535 are the well-known private BGP4 AS numbers. There is no default local AS number.

EXAMPLE:

To set the local AS, enter a command such as the following:

```
BigIron(config-bgp-router)# local-as 64512
```

Syntax: local-as <num>

Possible values: 1 – 65535

Default value: None

max-attribute-entries

Changes the maximum number of BGP4 route attribute entries the Layer 3 Switch can have.

NOTE: This command applies only to the TurboIron/8 and Stackable NetIron Layer 3 Switches. The Chassis Layer 3 Switches use dynamic memory allocation for BGP4.

EXAMPLE:

To change the maximum number of route-attribute entries to 2500, enter the following command:

```
BigIron(config-bgp-router)# max-attribute-entries 2500
BigIron(config-bgp-router)# end
```



```
BigIron# reload
```

Syntax: max-attribute-entries <num>

Possible values:

See the "Memory Considerations" section of the "Configuring BGP" chapter in the *Foundry Enterprise Configuration and Management Guide* for a listing of the maximum number of BGP4 route attribute entries the Layer 3 Switch can have.

maximum-paths

Changes the maximum number of shared 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 or use the # of Paths field next to Load Sharing on the IP configuration panel of the Web management interface.

EXAMPLE:

To change the maximum number of shared paths:

```
BigIron(config-bgp-router)# maximum-paths 4
```

Syntax: [no] maximum-paths <num>

Possible values: 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.

Default value: The default is 1.

max-neighbors

Changes the maximum number of BGP4 neighbors the Layer 3 Switch can have.

NOTE: This command applies only to the TurboIron/8 and Stackable NetIron Layer 3 Switches. The Chassis Layer 3 Switches use dynamic memory allocation for BGP4.

EXAMPLE:

To change the maximum number of BGP4 neighbors to 3, enter the following command:

```
BigIron(config-bgp-router)# max-neighbors 3
BigIron(config-bgp-router)# end
BigIron# reload
```

Syntax: max-neighbors <num>

Possible values: See the "Memory Considerations" section of the "Configuring BGP" chapter in the *Foundry Enterprise Configuration and Management Guide* for a listing of the maximum number of BGP4 route attribute entries the Layer 3 Switch can have.

max-routes

Changes the maximum number of BGP4 routes the Layer 3 Switch can have.

NOTE: You must save the configuration (**write memory**), then reload the software to place this command into effect.

NOTE: This command applies only to the TurboIron/8 and Stackable NetIron Layer 3 Switches. The Chassis Layer 3 Switches use dynamic memory allocation for BGP4.

EXAMPLE:

To change the maximum number of BGP4 routes to 30000, enter the following command:

```
BigIron(config-bgp-router)# max-routes 30000
BigIron(config-bgp-router)# end
BigIron# reload
```

Syntax: max-routes <num>

Possible values: See the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide* for a listing of the maximum number of BGP4 routes the Layer 3 Switch can have.

med-missing-as-worst

Configures the Layer 3 Switch to favor a route that has a MED over a route that is missing its MED.

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.

EXAMPLE:

```
BigIron(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. For information about how BGP4 selects a route path, see the "How BGP4 Selects a Path for a Route" section in the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

Possible values: N/A

Default value: Disabled

multipath

Disables load sharing of EBGP paths.

By default, when BGP4 load sharing is enabled, both IBGP and EBPG 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.

EXAMPLE:

```
BigIron(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:

```
BigIron(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:

```
BigIron(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.

Possible values: See above.

Default value: Load sharing of IBGP and EBGP paths is enabled.

neighbor

Adds a BGP4 neighbor (peer). In addition to identifying the neighbor's IP address and AS number, you can set other parameters that control the Layer 3 Switch's interaction with the neighbor.

EXAMPLE:

You can add a neighbor by specifying just the IP address and AS number. To set additional options, see the syntax descriptions below.

```
BigIron(config-bgp-router)# neighbor 1.1.1.10 remote-as 1
```

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 <portnum> | loopback <num> | pos <portnum> | ve <num>]
[weight <num>]

Syntax: 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.

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, when a Layer 3 Switch needs to send its entire routing table to a BGP4 neighbor, it sends the updates one immediately after another at a rate of one TCP window per second, without waiting for the advertisement interval.

The Layer 3 Switch still applies the advertisement interval to an update if the update contains a route for which the

it has just sent an update. For example, if the Layer 3 Switch sends an update for routes 1,2, and 3, then receives a change to an attribute of one of the routes before the advertisement interval has expired, the Layer 3 Switch waits to send an update for the change until the advertisement interval has expired.

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.

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.

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 Access Control Lists (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 “ip as-path” on page 6-42.

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 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.

nlri multicast | unicast | multicast unicast 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.

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 “ip prefix-list” on page 6-65.

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.

---

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. 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.

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.

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.

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.

unsuppress-map <map-name> removes route dampening from a neighbor's routes when those routes have been dampened due to aggregation. See the "Removing Route Dampening from a Neighbor's Routes Suppressed Due to Aggregation" section in the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

update-source <ip-addr> | **ethernet** <portnum> | **loopback** <num> | **pos** <portnum> | **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.

Possible values: see above

Default value: N/A

network

Specifies a list of networks for the Layer 3 Switch to advertise through BGP4 or MBGP.

EXAMPLE:

To configure the Layer 3 Switch to advertise network 209.157.22.0/24, enter the following command:

```
BigIron(config-bgp-router)# network 209.157.22.0 255.255.255.0
```

To configure the Layer 3 Switch to advertise network 207.95.22.0/24 as a multicast route, enter the following command:

```
BigIron(config-bgp-router)# network 207.95.22.0 255.255.255.0 nlri multicast
```

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 or MBGP 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.

NOTE: The **weight** and **backdoor** parameters do not apply to MBGP.

Possible values: see above

Default value: N/A

next-hop-enable-default

Enables the Layer 3 Switch to use the default IP route to resolve a BGP4 next-hop 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.

EXAMPLE:

```
BigIron(config-bgp-router)# next-hop-enable-default
```

Syntax: [no] next-hop-enable-default

Possible values: N/A

Default value: Disabled

next-hop-recursion

Enables the BGP4 next-hop recursive lookups. When you enable this feature, the Layer 3 Switch finds the IGP route to a BGP route's next-hop gateway. 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.

EXAMPLE:

To enable recursive next-hop lookups, enter the following command at the BGP configuration level of the CLI:

```
BigIron(config-bgp-router)# next-hop-recursion
```

Syntax: [no] next-hop-recursion

Possible values: N/A

Default value: Disabled

no

Disables other commands. To disable a command, place the word **no** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-bgp-router)# quit

BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

readvertise

Allows or prevents readvertising of a learned best BGP4 route unless that route also is installed in the IP route table. By default, the Layer 3 Switch readvertises all learned best BGP4 routes to BGP4 neighbors, unless the routes are discarded or blocked by route maps or other filters.

EXAMPLE:

To disable readvertisement of BGP4 routes to BGP4 neighbors except for routes that the software also installs in the route table, enter the following command:

```
BigIron(config-bgp-router)# no readvertise
```

Syntax: [no] readvertise

To re-enable readvertisement, enter the following command:

```
BigIron(config-bgp-router)# readvertise
```

Possible values: N/A

Default value: Enabled

redistribute connected

Configures parameters for redistributing routes to directly attached devices into BGP4. Redistribution into BGP4 is disabled by default.

EXAMPLE:

To configure the Layer 3 Switch to redistribute routes to directly attached devices, enter the following command:

```
BigIron(config-bgp-router)# redistribute connected
```

Syntax: redistribute connected [metric <num>] [route-map <map-name>] [weight <num>]

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 filter to the IP route table.

NOTE: The route map you specify must already be configured on the router. See the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide* for information about defining route maps.

The **weight** <num> parameter changes the weight. You can specify a value from 0 – 65535. The default is 0.

Possible values: see above

Default value: see above

redistribute ospf

Configures parameters for redistributing OSPF routes into BGP4. Redistribution into BGP4 is disabled by default.

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.

EXAMPLE:

To configure the Layer 3 Switch to redistribute OSPF external type 1 routes, enter the following command:

```
BigIron(config-bgp-router)# redistribute ospf match external1
```

Syntax: redistribute ospf [metric <num>] [route-map <map-name>] [weight <num>]
[match internal | external1 | external2]

The **ospf** parameter indicates that you are redistributing OSPF 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 OSPF route to the BGP4 route table.

NOTE: The route map you specify must already be configured on the router. See the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide* for information about defining route maps.

The **weight** <num> parameter changes the weight. You can specify a value from 0 – 65535. The default is 0.

The **match internal | external1 | external2** parameter applies only to OSPF. This parameter specifies the types of OSPF routes to be redistributed into BGP4.

Possible values: see above

Default value: see above

redistribute rip

Configures parameters for redistributing RIP routes into BGP4. Redistribution into BGP4 is disabled by default.

EXAMPLE:

To configure BGP4 to redistribute RIP routes and add a metric of 10 to the redistributed routes, enter the following command:

```
BigIron(config-bgp-router)# redistribute rip metric 10
```

Syntax: redistribute rip [metric <num>] [route-map <map-name>] [weight <num>]

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 the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide* for information about defining route maps.

The **weight** <num> parameter changes the weight. You can specify a value from 0 – 65535. The default is 0.

Possible values: see above

Default value: see above

redistribute static

Configures parameters for redistributing static routes into BGP4. Redistribution into BGP4 is disabled by default.

EXAMPLE:

To configure the Layer 3 Switch to redistribute static routes, enter the following command:

```
BigIron(config-bgp-router)# redistribute static
```

Syntax: redistribute static [metric <num>] [route-map <map-name>] [weight <num>]

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 the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide* for information about defining route maps.

The **weight** <num> parameter changes the weight. You can specify a value from 0 – 65535. The default is 0.

Possible values: see above

Default value: see above

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

synchronization

Enables or disables synchronization. When synchronization is enabled, the router waits until the IGPs in the local AS have fully exchanged route information before BGP4 advertises the routes to its remote BGP4 neighbors.

EXAMPLE:

To enable synchronization, enter the following command:

```
BigIron(config-bgp-router)# synchronization
```

Syntax: [no] synchronization

Possible values: N/A

Default value: Disabled

table-map

Configures an existing route map to change the route tag in routes when adding them to the IP route table.

EXAMPLE:

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.

```
BigIron(config)# route-map TAG_IP permit 1
BigIron(config-routemap TAG_IP)# match address-filters 11
BigIron(config-routemap TAG_IP)# set tag 100
BigIron(config-routemap TAG_IP)# router bgp
BigIron(config-bgp-router)# table-map TAG_IP
```

Syntax: table-map <route-map>

Possible values: a route-map name

Default value: N/A

timers

Sets the BGP4 Keep Alive Time and Hold Time on the Layer 3 Switch.

EXAMPLE:

To change the Keep Alive Time to 30 and Hold Time to 90, enter the following command:

```
BigIron(config-bgp-router)# timers keep-alive 30 hold-time 90
```

Syntax: timers keep-alive <num> hold-time <num>

Possible values:

Keep Alive Time 0 – 65535.

Hold Time 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.

Default value:

Keep Alive Time 60 seconds.

Hold Time 180 seconds.

update-time

Changes the BGP4 update timer. The update timer controls how often the Layer 3 Switch updates its BGP4 next-hop tables and affected BGP4 routes.

By default, the Layer 3 Switch updates its BGP4 next-hop tables and affected BGP4 routes five seconds after IGP route changes.

EXAMPLE:

```
BigIron(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.

Possible values: 1 – 30 seconds after IGP route changes

Default value: 5 seconds after IGP route changes

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-bgp-router)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-bgp-router)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 14
IP Tunnel Commands

encap-control

Enables or disables encapsulation of IP multicast control messages such as probe, route report, and so on using IP-IN-IP encapsulation on an IP Tunnel.

EXAMPLE:

```
BigIron(config-if-5/4)# ip tunnel 192.3.45.6
BigIron(config-if-pim-tunnel)# encap on
```

Syntax: encap-control on | off

Possible values: on, off

Default value: off

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
BigIron(config-if-4/5-tunnel)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the interface level.

EXAMPLE:

```
BigIron(config-if-4/5-tunnel)# exit
BigIron(config-if-4/5)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

metric

Sets the metric for an IP tunnel for use with the multicast protocol.

NOTE: Assign a higher metric to an IP tunnel than that of the interface, to ensure that the IP tunnel path takes precedence.

EXAMPLE:

To define an IP tunnel metric (cost) of 15, enter the following:

```
BigIron(config)# interface 5/2
BigIron(config-if-5/2)# ip tunnel 192.45.3.2 pim
BigIron(config-if-pim-tunnel)# metric 15
```

Syntax: metric <1-255>

Possible values: 1 – 255

Default value: N/A

no

Disables other commands. To disable a command, place the word **no** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-if-4/5-tunnel)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

ttl-threshold

Specifies how long a packet is considered viable on an IP Tunnel link.

EXAMPLE:

```
BigIron(config-if-3/4)# ip tunnel
BigIron(config-if-3/4-tunnel)# ttl 60
```

Syntax: ttl-threshold <value>

Possible values: 1 – 254

Default value: 1

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-if-3/4-tunnel)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-if-3/4-tunnel)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 15
MSDP Commands

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
BigIron(config-msdp-router)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
BigIron(config-msdp-router)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

msdp-peer

Configures an MSDP peer.

EXAMPLE:

```
BigIron(config-msdp-router)# msdp-peer 205.216.162.1
```

EXAMPLE:

```
BigIron(config)# interface loopback 1
BigIron(config-lbif-1)# ip address 9.9.9.9/32
BigIron(config-lbif-1)# interface ethernet 3/1
BigIron(config-if-3/1)# msdp-peer 2.2.2.99 connect-source loopback 1
```

Syntax: [no] msdp-peer <ip-addr> [connect-source loopback <num>]

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

The **connect-source loopback** <num> parameter specifies the loopback interface you want to use as the source for sessions with the neighbor. If you do not use this parameter, the Layer 3 Switch uses the sub-net interface configured on the port.

Possible values: See above

Default value: N/A

no

Disables other commands. To disable a command, place the word **no** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-msdp-router)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

sa-filter in

Filters source-group pairs received in Source-Active messages from an MSDP neighbor.

EXAMPLE:

The commands in this example configure the following filters:

- **sa-filter in 2.2.2.99** – This command drops all source-group pairs received from neighbor 2.2.2.99.

  NOTE: The default action is to deny all source-group pairs from the specified neighbor. If you want to permit some pairs, use route maps.

- **sa-filter in 2.2.2.97 route-map msdp_map** – This command drops source-group pairs received from neighbor 2.2.2.97 if the pairs have source address 10.x.x.x and any group address.
- **sa-filter in 2.2.2.96 route-map msdp2_map rp-route-map msdp2_rp_map** – This command accepts all source-group pairs except those associated with RP 2.2.42.3.

The following commands configure an IP address on port 3/1. This is the port on which the MSDP neighbors will be configured.

```
BigIron(config)# interface ethernet 3/1
BigIron(config-if-3/1)# ip address 2.2.2.98/24
BigIron(config-if-3/1)# exit
```

The following commands configure a loopback interface. The Layer 3 Switch will use this interface as the source address for communicating with the MSDP neighbors.

```
BigIron(config)# interface loopback 1
BigIron(config-lbif-1)# ip address 9.9.9.8/32
BigIron(config-lbif-1)# exit
```

The following commands configure extended ACLs. The ACLs will be used in route maps, which will be used by the Source-Active filters.

```
BigIron(config)# access-list 123 permit 10.0.0.0 0.255.255.255 any
BigIron(config)# access-list 124 permit 2.2.42.3 0.0.0.0 any
BigIron(config)# access-list 125 permit any any
```


The following commands configure the route maps.

```
BigIron(config)# route-map msdp_map deny 1
BigIron(config-routemap msdp_map)# match ip address 123
BigIron(config-routemap msdp_map)# exit
BigIron(config)# route-map msdp2_map permit 1
BigIron(config-routemap msdp2_map)# match ip address 125
BigIron(config-routemap msdp2_map)# exit
BigIron(config)# route-map msdp2_rp_map deny 1
BigIron(config-routemap msdp2_rp_map)# match ip route-source 124
BigIron(config-routemap msdp2_rp_map)# exit
```

The following commands enable MSDP and configure the MSDP neighbors on port 3/1.

```
BigIron(config)# router msdp
BigIron(config-msdp-router)# exit
BigIron(config)# interface ethernet 3/1
BigIron(config-if-3/1)# msdp-peer 2.2.2.99 connect-source loopback 1
BigIron(config-if-3/1)# msdp-peer 2.2.2.97 connect-source loopback 1
BigIron(config-if-3/1)# msdp-peer 2.2.2.96 connect-source loopback 1
BigIron(config-if-3/1)# exit
```

The following commands configure the Source-Active filters.

```
BigIron(config)# router msdp
BigIron(config-msdp-router)# sa-filter in 2.2.2.99
BigIron(config-msdp-router)# sa-filter in 2.2.2.97 route-map msdp_map
BigIron(config-msdp-router)# sa-filter in 2.2.2.96 route-map msdp2_map rp-route-map
msdp2_rp_map
```

Syntax: [no] sa-filter in <ip-addr> [route-map <map-tag>] [rp-route-map <rp-map-tag>]

The <ip-addr> parameter specifies the IP address of the MSDP neighbor. The filter applies to Active-Source messages received from this neighbor.

The **route-map** <map-tag> parameter specifies a route map. The Layer 3 Switch applies the filter to source-group pairs that match the route map. Use the **match ip address** <acl-id> command in the route map to specify an extended ACL that contains the source and group addresses.

The **rp-route-map** <rp-map-tag> parameter specifies a route map to use for filtering based on Rendezvous Point (RP) address. Use this parameter if you want to filter Source-Active messages based on their origin. If you use the **route-map** parameter instead, messages are filtered based on source-group pairs but not based on origin. Use the **match ip route-source** <acl-id> command in the route map to specify the RP address.

NOTE: The default filter action is deny. If you want to permit some source-group pairs, use a route map. A permit action in the route map allows the Layer 3 Switch to receive the matching source-group pairs. A deny action in the route map drops the matching source-group pairs.

Possible values: See above

Default value: None configured

sa-filter originate

Filters source-group pairs in Source-Active messages in advertisements to an MSDP neighbor.

EXAMPLE:

The following example configures the Layer 3 Switch to advertise all source-group pairs except the ones that have source address 10.x.x.x.

The following commands configure an IP address on port 3/1. This is the port on which the MSDP neighbors will be configured.

```
BigIron(config)# interface ethernet 3/1
BigIron(config-if-3/1)# ip address 2.2.2.98/24
```



```
BigIron(config-if-3/1)# exit
```

The following commands configure a loopback interface. The Layer 3 Switch will use this interface as the source address for communicating with the MSDP neighbors.

```
BigIron(config)# interface loopback 1
BigIron(config-lbif-1)# ip address 9.9.9.8/32
BigIron(config-lbif-1)# exit
```

The following command configures an extended ACL to specify the source and group addresses you want to filter.

```
BigIron(config)# access-list 123 permit 10.0.0.0 0.255.255.255 any
```

The following commands configure a route map. The map matches on source address 10.x.x.x and any group address. Since the action is deny, the Source-Active filter that uses this route map will remove the source-group pairs that match this route map from the Source-Active messages to the neighbor.

```
BigIron(config)# route-map msdp_map deny 1
BigIron(config-routemap msdp_map)# match ip address 123
BigIron(config-routemap msdp_map)# exit
```

The following commands enable MSDP and configure MSDP neighbors on port 3/1.

```
BigIron(config)# router msdp
BigIron(config-msdp-router)# exit
BigIron(config)# interface ethernet 3/1
BigIron(config-if-3/1)# msdp-peer 2.2.2.99 connect-source loopback 1
BigIron(config-if-3/1)# msdp-peer 2.2.2.97 connect-source loopback 1
BigIron(config-if-3/1)# exit
```

The following commands configure the Source-Active filter.

```
BigIron(config)# router msdp
BigIron(config-msdp-router)# sa-filter originate route-map msdp_map
```

This filter removes source-group pairs that match route map msdp_map from Source-Active messages before sending them to MSDP neighbors.

Syntax: [no] sa-filter originate [route-map <map-tag>]

The **route-map** <map-tag> parameter specifies a route map. The Layer 3 Switch applies the filter to source-group pairs that match the route map. Use the **match ip address** <acl-id> command in the route map to specify an extended ACL that contains the source and group addresses.

NOTE: The default filter action is deny. If you want to permit some source-group pairs, use a route map. A permit action in the route map allows the Layer 3 Switch to receive the matching source-group pairs. A deny action in the route map drops the matching source-group pairs.

Possible values: See above

Default value: None configured

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-msdp-router)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-msdp-router)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 16
DVMRP Commands

default-gateway

Defines the default gateway for DVMRP IP multicast routing. If designated as the default gateway, the router must be on a directly connected network for this command to be operational.

EXAMPLE:

```
BigIron(config)# router dvmrp
BigIron(config-dvmrp-router)# default-gateway 192.35.4.1
```

Syntax: default-gateway <ip-addr>

Possible values: valid IP address

Default value: no system default

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
BigIron(config-dvmrp-router)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
BigIron(config-dvmrp-router)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

graft-retransmit-timer

Defines the initial period of time that a DVMRP router sending a graft message, will wait for a graft acknowledgment from an upstream router, before re-transmitting that message.

Subsequent re-transmissions will be sent at an interval of twice that of the preceding interval.

DVMRP must be enabled on the router for this command to be operational.

EXAMPLE:

```
BigIron(config)# router dvmrp
BigIron(config-dvmrp-router)# graft 120
```

Syntax: graft-retransmit-timer <value>

Possible values: 5 – 3600 seconds

Default value: 10 seconds

nbr-timeout

Sets neighbor timeout value, which is the period of time that a router will wait before it defines an attached DVMRP neighbor router as down.

DVMRP must be enabled on the router for this command to be operational.

EXAMPLE:

```
BigIron(config)# router dvmrp
BigIron(config-dvmrp-router)# nbr-timeout 100
```

Syntax: nbr-timeout <value>

Possible values: 40 – 8000 seconds

Default value: 40 seconds

no

Disables other commands. To disable a command, place the word **no** before the command.

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.

DVMRP must be enabled on the router for this command to be operational.

EXAMPLE:

```
BigIron(config)# router dvmrp
BigIron(config-dvmrp-router)# probe 10
```

Syntax: probe-interval <value>

Possible values: 5 – 30 seconds

Default value: 10 seconds

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.

DVMRP must be enabled on the router for this command to be operational.

EXAMPLE:

```
BigIron(config)# router dvmrp
BigIron(config-dvmrp-router)# prune 25
```


Syntax: prune-age <value>

Possible values: 20 – 3600 seconds

Default value: 180 seconds

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-dvmrp-router)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

report-interval

Defines how often routers will propagate their complete routing tables to other neighbor DVMRP routers.

DVMRP must be enabled on the router for this command to be operational.

EXAMPLE:

```
BigIron(config)# router dvmrp
BigIron(config-dvmrp-router)# report 100
```

Syntax: report-interval <value>

Possible values: 10 – 2000 seconds

Default value: 60 seconds

route-discard-timeout

Defines the period of time before a route is deleted on a DVMRP router.

DVMRP must be enabled on the router for this command to be operational.

EXAMPLE:

```
BigIron(config)# router dvmrp
BigIron(config-dvmrp-router)# route-discard-timeout 50
```

Syntax: route-discard-timeout <value>

Possible values: 40 – 8000 seconds

Default value: 340 seconds

route-expire-timeout

Defines how long a route is considered valid without the next route update.

DVMRP must be enabled on the router for this command to be operational.

EXAMPLE:

```
BigIron(config)# router dvmrp
BigIron(config-dvmrp-router)# route-expire-time 50
```

Syntax: route-expire-time <value>

Possible values: 20 – 4000 seconds

Default value: 200 seconds

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

trigger-interval

Defines how often trigger updates, which reflect changes in the network topology, are sent. Changes in a network topology can include a router coming up or going down or changing its metric.

DVMRP must be enabled on the router for this command to be operational.

EXAMPLE:

```
BigIron(config)# router dvmrp
BigIron(config-dvmrp-router)# trigger-interval 25
```

Syntax: trigger-interval <value>

Possible values: 5 – 30 seconds

Default value: 5

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-dvmrp-router)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-dvmrp-router)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 17
PIM Commands

bsr-candidate

Configures the Layer 3 Switch as a candidate PIM Sparse Bootstrap Router (BSR).

EXAMPLE:

To configure the Layer 3 Switch as a candidate BSR, enter a command such as the following:

```
BigIron(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 <portnum> | loopback <num> | ve <num> <hash-mask-length> [<priority>]

The **ethernet** <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** <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.

Possible values: N/A

Default value: N/A

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
BigIron(config-pim-router)# end
```

```
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
BigIron(config-pim-router)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

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 it will respond with a Graft Ack message. If this Graft Ack message is lost, the router that sent the graft message, resends it. The interval between the transmission of the first and subsequent graft message is what is configurable with the PIM graft retransmit timer.

EXAMPLE:

To change the graft retransmit timer from the default of 180 to 90 seconds, enter the following:

```
BigIron(config-pim-router)# graft-retransmit-timer 90
```

Syntax: graft-retransmit-timer <value>

Possible values: 10 – 3600 seconds

Default value: 180 seconds

hello-timer

Defines the time interval at which periodic hellos are sent out on all interfaces of a PIM-capable router. Routers use hello messages to inform neighboring routers of their presence.

EXAMPLE:

To apply a PIM hello timer of 120 seconds to all ports on the router operating with PIM, enter the following:

```
BigIron(config-pim-router)# hello-timer 120
```

Syntax: hello-timer <value>

Possible values: 10 – 3600 seconds

Default value: 60 seconds

inactivity-timer

A forwarding entry is deleted if it is not used to send multicast packets. The PIM inactivity timer defines the time interval after which an inactive forwarding entry is deleted.

EXAMPLE:

To apply a PIM inactivity timer of 90 seconds to all ports on the router operating with PIM, enter the following:

```
BigIron(config-pim-router)# inactivity-timer 90
```

Syntax: inactivity-timer <value>

Possible values: 10 – 3600 seconds

Default value: 180 seconds

message-interval

Changes the PIM Sparse Join/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.

EXAMPLE:

To change the Join/Prune interval, enter commands such as the following:

```
BigIron(config)# router pim
BigIron(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.

Possible values: 1 – 65535 seconds

Default value: 60 seconds

nbr-timeout

If a neighboring PIM router stops sending out PIM Hello messages, the router will eventually discover that the neighbor is not present. Neighbor timeout is the interval after which a PIM-capable router will consider a neighbor to not be present.

EXAMPLE:

To apply a PIM neighbor timeout value of 360 seconds to all ports on the router operating with PIM, enter the following:

```
BigIron(config-pim-router)# nbr-timeout 360
```

Syntax: nbr-timeout <value>

Possible values: 60 – 8000 seconds.

Default value: 180 seconds

no

Disables other commands. To disable a command, place the word **no** before the command.

prune-timer

This parameter is used to define how long a Foundry Layer 3 Switch will maintain a prune state for a forwarding entry.

The first received multicast interface is forwarded to all other PIM interfaces on the Layer 3 Switch. If there is no presence of groups on that interface, the leaf node will send a prune message upstream and store a prune state. This prune state will travel up the tree and install a prune state.

A prune state is maintained until the prune timer expires or a graft message is received for the forwarding entry.

EXAMPLE:

To apply a PIM prune timer of 90 seconds to all ports on the Layer 3 Switch operating with PIM, enter the following:


```
BigIron(config-pim-router)# prune-timer 90
```

Syntax: prune-timer <value>

Possible values: 10 – 3600 seconds.

Default value: 180 seconds

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-pim-router)# quit

BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

rp-address

Statically configures the address of the PIM Sparse Rendezvous Point (RP). Furthermore, beginning with software release 07.6.03, 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. See the examples below.

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.

EXAMPLE:

To specify the IP address of the RP, enter commands such as the following:

```
BigIron(config)# router pim
BigIron(config-pim-router)# rp-address 207.95.7.1
```

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.

Syntax: [no] rp-address <ip-address> [<access-list-num>] [override]

Possible values:

The <ip-address> parameter specifies the IP address of the RP.

The <access-list-num> parameter is the number of the standard ACL that will filter the multicast group.

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.

Default value: see above

NOTE: Extended ACLs cannot be used to limit static RP groups.

EXAMPLE:

To configure an RP that covers multicast groups in 239.255.x.x, enter commands such as the following:

```
BigIron(config)# access-list 2 permit 239.255.162.0 0.0.0.255

BigIron(config)# router pim
BigIron(config-pim-router)# rp-address 43.43.43.1 2
```

EXAMPLE:

To configure an RP that covers multicast groups in the 239.255.x.x range, except the 239.255.162.2 group, enter commands such as the following:

```
BigIron(config)# access-list 5 deny host 239.255.162.2

BigIron(config)# access-list 5 permit 239.255.0.0 0.0.255.255

BigIron(config)# router pim
BigIron(config-pim-router)# rp-candidate ve 43
BigIron(config-pim-router)# rp-address 99.99.99.5 5
```

EXAMPLE:

To configure an RP for multicast groups using the override switch, enter commands such as the following:

```
BigIron(config)# access-list 44 permit 239.255.162.0 0.0.0.255

BigIron(config)# router pim
BigIron(config-pim-router)# rp-address 43.43.43.1
BigIron(config-pim-router)# rp-address 99.99.99.5 44 override
```

rp-candidate

Configures the Layer 3 Switch as a candidate PIM Sparse Rendezvous Point (RP).

EXAMPLE:

Enter a command such as the following to configure the Layer 3 Switch as a candidate RP:

```
BigIron(config-pim-router)# rp-candidate ethernet 2/2
```

Syntax: [no] rp-candidate ethernet <portnum> | loopback <num> | ve <num> | [group-list <access-list-num>]

The **ethernet** <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** <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.

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.

```
BigIron(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:


```
BigIron(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.

Possible values: see above

Default value: see above

EXAMPLE:

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:

```
BigIron(config)# interface ethernet 1/1
BigIron(config-if-1/1)# ip address 99.99.99.5 255.255.255.0
BigIron(config-if-1/1)# ip pim-sparse
BigIron(config-if-1/1)# exit

BigIron(config)# access-list 5 deny host 239.255.162.2
BigIron(config)# access-list 5 permit 239.0.0.0 0.0.255.255

BigIron(config)# router pim
BigIron(config-pim-router)# rp-candidate ethernet 1/1 group-list 5
```

The example above shows a configuration for an Ethernet interface.

EXAMPLE:

To configure ACLs that are applied to a virtual routing interface, enter commands such as the following:

```
BigIron(config)# interface ve 16
BigIron(config-vif-16)# ip address 16.16.16.1 255.255.255.0
BigIron(config-vif-16)# ip pim-sparse
BigIron(config-vif-16)# exit

BigIron(config)# access-list 5 deny host 239.255.162.2
BigIron(config)# access-list 5 permit 239.255.0.0 0.0.255.255

BigIron(config)# router pim
BigIron(config-pim-router)# rp-candidate ve 16 group-list 5
```

EXAMPLE:

To configure ACLs that are applied to a loopback interface, enter commands such as the following:

```
BigIron(config)# interface loopback 1
BigIron(config-lbif-1)# ip address 88.88.88.8 255.255.255.0
BigIron(config-lbif-1)# ip pim-sparse
BigIron(config-lbif-1)# exit

BigIron(config)# access-list 5 deny host 239.255.162.2
BigIron(config)# access-list 5 permit 239.255.0.0 0.0.255.255

BigIron(config)# router pim
BigIron(config-pim-router)# rp-candidate loopback 1 group-list 5
```

NOTE: Extended ACLs cannot be used for group-list.

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

spt-threshold

Changes the PIM Sparse Shortest Path Tree (SPT) threshold, which specifies the number of packets the Layer 3 Switch sends using the RP before switching to the SPT.

EXAMPLE:

To change the number of packets the Layer 3 Switch sends using the RP before switching to the SPT, enter commands such as the following:

```
BigIron(config)# router pim
BigIron(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. The default is 1 packet.

Possible values: see above

Default value: 1 packet

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-pim-router)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-pim-router)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 18
Broadcast and Multicast Filter Commands

Broadcast Filter Commands

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
BigIron(config-bcast-filter-id-4)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exclude-ports

Specifies the ports to which you are applying a Layer 2 broadcast filter. Broadcast filters drop broadcast packets from the outbound queue of the ports you specify.

EXAMPLE:

To configure a Layer 2 broadcast filter to filter all types of broadcasts, then apply the filter to ports 1/1, 1/2, and 1/3, enter the following commands:

```
BigIron(config)# broadcast filter 1 any
BigIron(config-bcast-filter-id-1)# exclude-ports ethernet 1/1 to 1/3
BigIron(config-bcast-filter-id-1)# write memory
```

EXAMPLE:

To configure two filters, one to filter IP UDP traffic on ports 1/1 – 1/4, and the other to filter all broadcast traffic on port 4/6, enter the following commands:

```
BigIron(config)# broadcast filter 1 ip udp
BigIron(config-bcast-filter-id-1)# exclude-ports ethernet 1/1 to 1/4
BigIron(config-bcast-filter-id-1)# exit
BigIron(config)# broadcast filter 2 any
BigIron(config-bcast-filter-id-2)# exclude-ports ethernet 4/6
```

```
BigIron(config-bcast-filter-id-2)# write memory
```

EXAMPLE:

To configure an IP UDP broadcast filter that applies only to port-based VLAN 10, then apply the filter to two ports within the VLAN, enter the following commands:

```
BigIron(config)# broadcast filter 4 ip udp vlan 10
BigIron(config-bcast-filter-id-4)# exclude-ports eth 1/1 eth 1/3
BigIron(config-bcast-filter-id-4)# write memory
```

Syntax: [no] exclude-ports ethernet <portnum> [to | ethernet <portnum>]

Possible values: see above

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the interface level.

EXAMPLE:

```
BigIron(config-bcast-filter-id-4)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

no

Disables other commands. To disable a command, place the word **no** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-bcast-filter-id-4)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-bcast-filter-id-4)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-bcast-filter-id-4)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Multicast Filter Commands

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
BigIron(config-mcast-filter-id-1)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exclude-ports

Specifies the ports to which you are applying a Layer 2 multicast filter. Multicast filters drop multicast packets from the outbound queue of the ports you specify.

EXAMPLE:

To configure a Layer 2 multicast filter to filter all multicast groups, then apply the filter to ports 2/4, 2/5, and 2/8, enter the following commands:

```
BigIron(config)# multicast filter 1 any
BigIron(config-mcast-filter-id-1)# exclude-ports ethernet 2/4 to 2/5 ethernet 2/8
```

EXAMPLE:

To configure a multicast filter to block all multicast traffic destined for multicast addresses 0100.5e00.5200 – 0100.5e00 on port 4/8, enter the following commands:

```
BigIron(config)# multicast filter 2 0100.5e00.5200 ffff.ffff.ff00
BigIron(config-mcast-filter-id-2)# exclude-ports ethernet 4/8
```

The software calculates the range by combining the mask with the multicast address. In this example, all but the last two bits in the mask are “significant bits” (ones). The last two bits are zeros and thus match on any value.

Syntax: [no] exclude-ports ethernet <portnum> [to | ethernet <portnum>]

Possible values: see above

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the interface level.

EXAMPLE:

```
BigIron(config-mcast-filter-id-1)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

no

Disables other commands. To disable a command, place the word **no** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-mcast-filter-id-1)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-mcast-filter-id-1)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-mcast-filter-id-1)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 19
Route Map Commands

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged EXEC level, enter the following from any level of the CLI.

```
BigIron(config-routemap GET_ONE)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the port-based VLAN level if configuring a protocol VLAN. If configuring a port-based VLAN, activity would be moved to the global level.

EXAMPLE:

```
BigIron(config-routemap GET_ONE)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

match

Configures a value that a route must match in order for the route map instance containing the match statement to be evaluated as "true".

When a route comparison yields a "true" value, the Layer 3 Switch uses the set statements configured for the route map instance to modify the route.

EXAMPLE:

```
BigIron(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.

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.

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** <acl> | **prefix-list** <string> parameter specifies an ACL or IP prefix list. Use this parameter to match based on the destination network. To configure an IP ACL for use with this command, use the **ip access-list** command.

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.

Possible values: see above

Default value: see above

no

Disables other commands. To disable a command, place the word ***no*** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-routemap GET_ONE)# quit
```



```
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

set

Modifies a route that matches at least one of the match statements in a route map.

EXAMPLE:

```
BigIron(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 | pos <portnum>] |
[ip [default] next hop <ip-addr>]
[ip next-hop peer-address] |
[level level-1 | level-1-2 | level-2] |
[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.

The **[default] interface null0 | pos** <portnum> 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. Alternatively, you can send the traffic to a POS interface. 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 **level level-1 | level-1-2 | level-2** parameter sets the IS-IS level.

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 **set comm-list** <acl> delete parameter deletes the specified communities from a route's communities attribute.

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.

The **weight** <num> parameter sets the weight for the route. You can specify a weight value from 0 – 4294967295.

Possible values: see above

Default value: see above

set mirror-interface

Configures a mirror port for hardware-based Policy-Based Routing (PBR) traffic.

NOTE: This feature applies to software release 07.6.03 or later, and hardware-based PBR, which is currently supported only on JetCore and FastIron 4802 (FWS 4802) premium devices, and on 10 Gigabit Ethernet modules.

EXAMPLE:

```
BigIron(config)# route-map ssl-pbr-map permit 1
BigIron(config-routemap ssl-pbr-map)# match ip address 100
BigIron(config-routemap ssl-pbr-map)# set mirror-interface 5
BigIron(config-routemap ssl-pbr-map)# set next-hop 10.10.10.1
BigIron(config-routemap ssl-pbr-map)# exit
```

The above commands configure an entry in the PBR route map named “ssl-pbr-map” . The **match** statement matches on IP information in ACL 100. The **set mirror-interface** statement specifies interface e 5 as the mirror port for matched ACL permit clauses. The **set next-hop** statement sets the IP address of the route’s next hop router to 10.10.10.1.

Syntax: set mirror-interface <slot number>/<port number>

You can specify up to 4 mirror ports for each PBR route map instance. To do so, enter the **set mirror interface** command for each mirror port.

Possible values:

The <slot number> parameter specifies the slot number on a Foundry chassis device. This parameter is not applicable to stackable devices.

The <port number> parameter specifies the mirror port number.

Default value: disabled

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-routemap GET_ONE)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-routemap GET_ONE)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 20
VRRP Commands

activate

Activates a virtual router configuration on a router interface.

EXAMPLE:

```
BigIron(config-if-1/6-vrid-1)# activate
```

Syntax: activate

Possible values: N/A

Default value: N/A

advertise backup

Enables a Backup router to send keepalive messages to the Master router.

EXAMPLE:

```
BigIron(config-if-1/6-vrid-1)# advertise backup
```

Syntax: [no] advertise backup

backup

Indicates that the virtual router interface you are configuring is for a Backup router.

Syntax: backup [priority <value>] [track-priority <value>]

backup-hello-interval

Changes the rate at which a Backup router sends keepalive messages to the Master router.

EXAMPLE:

```
BigIron(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.

dead-interval

Sets the VRRP dead interval.

Syntax: dead-interval <value>

Possible values: The Dead interval can be from 1 – 84 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.

Default value: The default is 3.5 seconds.

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged EXEC level, enter the following from any level of the CLI.

```
BigIron(config-if-1/6-vrid-1)# end

BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the port-based VLAN level if configuring a protocol VLAN. If configuring a port-based VLAN, activity would be moved to the global level.

EXAMPLE:

```
BigIron(config-if-1/6-vrid-1)# exit

BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

hello-interval

Sets the VRRP hello interval.

Syntax: hello-interval <value>

ip-address

Indicates the IP address for which the virtual router is providing redundancy.

Syntax: ip-address <ip-addr>

no

Disables other commands. To disable a command, place the word ***no*** before the command.

non-preempt-mode

By default, a Backup VRRP router that has a higher priority than another Backup router that has become Master can preempt that router to become the new Master router. If you want to prevent this behavior, disable preemption.

Syntax: non-preempt-mode

owner

Indicates that the virtual router interface you are configuring owns the real IP address for which the virtual router is providing redundancy, and allows you to change the VRRP priority for the address owner.

You can force a VRRP master router to abdicate (give away control) of a virtual router ID (VRID) to a VRRP backup router by temporarily changing the master router's VRRP priority to a value less than the backup router's.

The default VRRP Master router always has VRRP priority 255. You can change the priority to a value from 1 – 254.

NOTE: When you change the default VRRP Master router'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 default VRRP Master router again has priority 255.

EXAMPLE:

To change the Master VRRP router's priority:

```
BigIron(config)# ip int eth 3/1
BigIron(config-if-3/1)# ip vrrp vrid 1
BigIron(config-if-3/1-vrid-1)# owner priority 99
```

Syntax: [no] owner priority | track-priority <num>

Possible values: The <num> parameter specifies the priority and can be a number from 1 – 254.

Default value: N/A

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-if-1/6-vrid-1)# quit

BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See "Show Commands" on page 43-1.

track-port

Configures a track port.

Syntax: track-port ethernet <portnum>

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-if-1/6-vrid-1)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-if-1/6-vrid-1)# wr term
```


Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 21
VRRPE Commands

activate

Activates a virtual router configuration on a router interface.

EXAMPLE:

```
BigIron(config-if-1/6-vrid-1)# activate
```

Syntax: activate

Possible values: N/A

Default value: N/A

advertise backup

Enables a Backup router to send keepalive messages to the Master router.

EXAMPLE:

```
BigIron(config-if-1/6-vrid-1)# advertise backup
```

Syntax: [no] advertise backup

backup

Indicates that the virtual router interface you are configuring is for a Backup router.

Syntax: [no] backup [priority <value>] [track-priority <value>]

backup-hello-interval

Changes the rate at which a Backup router sends keepalive messages to the Master router.

EXAMPLE:

```
BigIron(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.

dead-interval

Sets the VRRPE dead interval.

Syntax: dead-interval <value>

Possible values: The Dead interval can be from 1 – 84 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.

Default value: The default is 3.5 seconds.

disable

Disables the VRID.

Syntax: disable

enable

Enables the VRID. This command does the same thing as the **activate** command.

Syntax: enable

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged EXEC level, enter the following from any level of the CLI.

```
BigIron(config-if-1/6-vrid-1)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the port-based VLAN level if configuring a protocol VLAN. If configuring a port-based VLAN, activity would be moved to the global level.

EXAMPLE:

```
BigIron(config-if-1/6-vrid-1)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

hello-interval

Sets the VRRPE hello interval.

Syntax: hello-interval <value>

ip address

Indicates the IP address for which the virtual router is providing redundancy. This command does the same thing as the **ip-address** command.

Syntax: ip address <ip-addr>

ip-address

Indicates the IP address for which the virtual router is providing redundancy. This command does the same thing as the **ip address** command.

Syntax: ip-address <ip-addr>

no

Disables other commands. To disable a command, place the word ***no*** before the command.

non-preempt-mode

By default, a Backup VRRPE router that has a higher priority than another Backup router that has become Master can preempt that router to become the new Master router. If you want to prevent this behavior, disable preemption.

Syntax: non-preempt-mode

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-if-1/6-vrid-1)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

track-port

Configures a track port.

Syntax: track-port ethernet <portnum>

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-if-1/6-vrid-1)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-if-1/6-vrid-1)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 22
MPLS Commands

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
NetIron(config-mpls)# end
NetIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
NetIron(config-mpls)# exit
NetIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

ldp

Enters the MPLS LDP configuration level. See “MPLS LDP Commands” on page 31-1 for a description of commands available at this configuration level.

EXAMPLE:

```
NetIron(config-mpls)# ldp
NetIron(config-mpls-ldp)#
```

Syntax: ldp

Possible values: N/A

Default value: N/A

ldp-enable

Enables LDP on an interface.

EXAMPLE:

```
NetIron(config-mpls)# mpls-interface e 1/2
NetIron(config-mpls)# ldp-enable
```

Syntax: ldp-enable

Possible values: N/A

Default value: N/A

NOTE: You should enable LDP on the same set of interfaces that IGP routing protocols such as OSPF and IS-IS are enabled.

lsp

Creates a signalled LSP and enters the signalled LSP configuration level. See “MPLS Signalled LSP Commands” on page 26-1 for a description of commands available at this configuration level.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)#
```

Syntax: [no] lsp <name>

Possible values: Signalled LSP name

Default value: N/A

mpls-interface

Enables MPLS either on all POS interfaces at once or on specified individual interfaces.

EXAMPLE:

```
NetIron(config)# router mpls
NetIron(config-mpls)# mpls-interface pos 3/11
NetIron(config-mpls-interface)
```

Syntax: [no] mpls-interface all-pos | <interface number>

Possible values: The **all-pos** parameter enables MPLS on all POS interfaces. The <interface number> parameter enables MPLS on a specified POS interface.

After you specify an <interface number>, you enter the MPLS interface configuration level, where you can assign the interface to administrative groups or change the amount of reservable bandwidth on the interface. See “MPLS Interface Commands” on page 23-1 for a description of commands available at this configuration level.

Default value: N/A

no

Disables other commands. To disable a command, place the word **no** before the command.

path

Creates a path and enters the MPLS path configuration level. A path is a list of router hops that specifies a route across an MPLS domain. See “MPLS Path Commands” on page 25-1 for a description of commands available at this configuration level.

EXAMPLE:

```
NetIron(config-mpls)# path sf_to_sj
NetIron(config-mpls-path)#
```

Syntax: [no] path <path name>

Possible values: MPLS path name

Default value: N/A

policy

Enters the global MPLS policy configuration level. See “MPLS Policy Commands” on page 24-1 for a description of commands available at this configuration level.

EXAMPLE:

```
NetIron(config-mpls)# policy
NetIron(config-mpls-policy)#
```

Syntax: policy

Possible values: N/A

Default value: N/A

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
NetIron(config-mpls)# quit
NetIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

rsvp

Enters the MPLS RSVP configuration level. See “MPLS RSVP Commands” on page 28-1 for a description of commands available at this configuration level.

EXAMPLE:

```
NetIron(config-mpls)# rsvp
NetIron(config-mpls-rsvp)#
```

Syntax: rsvp

Possible values: N/A

Default value: N/A

vll

Creates an MPLS VLL and enters the MPLS VLL configuration level. See “MPLS VLL Commands” on page 30-1 for a description of commands available at this configuration level.

EXAMPLE:

```
NetIron(config-mpls)# vll foundry-sj-to-sf 40000
NetIron(config-mpls-vll)#
```

Syntax: vll <vll-name> <vll-vc-id> [cos <cos value>]

Possible values: The <vll-vc-id> corresponds to the user-configurable ID defined in draft-martini-l2circuit-trans-mpls-07.txt.

The **cos** <cos value> parameter specifies an optional Class of Service (COS) setting for the VLL. If a COS value is set, the Foundry device selects a tunnel LSP that also has this COS value, if one is available. If no tunnel LSP with this COS value is available, the Foundry device selects a tunnel LSP with the highest configured COS value (although never higher than the COS setting for the VLL). The COS value can be between 0 – 7.

Default value: N/A

vpls

Creates an MPLS VPLS instance and enters the MPLS VPLS configuration level. See "MPLS VPLS Commands" on page 32-1 for a description of commands available at this configuration level.

EXAMPLE:

```
NetIron(config-mpls)# vpls v1 40000
NetIron(config-mpls-vpls-v1)#
```

Syntax: vpls <name> <vpls-vcid> [cos <cos-value>] [max-mac <max-mac-entries>]

Possible values: The <vpls-vcid> corresponds to the VPLS VC ID entered on the other PE routers in the VPLS instance

The **cos** <cos value> parameter specifies an optional Class of Service (COS) setting for the VPLS instance. If a COS value is set, the Foundry device selects a tunnel LSP that also has this COS value, if one is available. If no tunnel LSP with this COS value is available, the Foundry device selects a tunnel LSP with the highest configured COS value (although never higher than the COS setting for the VLL). The COS value can be between 0 – 7.

The **max-mac** <max-mac-entries> parameter specifies the maximum number of MAC entries expected to be learned for the VPLS instance. The default is 2048. This setting can be used for optimizing the amount of memory resources for each VPLS instance. The <max-mac-entries> setting is a soft limit – that is, if the number of database entries for a VPLS instance exceeds its <max-mac-entries> setting, but there is still space available in the VPLS MAC database, new MAC addresses will still be learned.

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See "Show Commands" on page 43-1.

static-lsp

Creates a static LSP and enters the static LSP configuration level. See "MPLS Static LSP Commands" on page 29-1 for a description of commands available at this configuration level.

EXAMPLE:

```
NetIron(config-mpls)# static-lsp tunnel1
NetIron(config-mpls-static-lsp)#
```

Syntax: [no] static-lsp <name>

Possible values: Static LSP name

Default value: N/A

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
NetIron(config-mpls)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
NetIron(config-mpls)# wr term
```


Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 23
MPLS Interface Commands

admin-group

Assigns an MPLS-enabled interface to one or more an administrative groups.

EXAMPLE:

To assign an interface to group “gold”:

```
NetIron(config-mpls)# mpls-interface pos 3/11
NetIron(config-mpls-interface)# admin-group gold
```

An MPLS-enabled interface can belong to any number of administrative groups. For example, to assign an interface to group “gold” and group 31:

```
NetIron(config-mpls)# mpls-interface pos 3/11
NetIron(config-mpls-interface)# admin-group gold 31
```

Syntax: admin-group <number> | <name> ...

Possible values: The <number> can be from 0 – 31. The <name> must be a previously configured administrative group name.

Default value: N/A

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
NetIron(config-mpls)# end
NetIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
NetIron(config-mpls)# exit
NetIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

no

Disables other commands. To disable a command, place the word **no** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
NetIron(config-mpls)# quit
NetIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

reservable-bw

Changes the amount of reservable bandwidth on an MPLS-enabled interface (that is, modifies the value in the Maximum Reservable Bandwidth TLV in OSPF-TE LSAs sent out for the interface). Changing the amount of reservable bandwidth on an interface causes the amount of unreserved bandwidth to be recalculated. In addition, it may cause an OSPF-TE LSA to be issued, as well as possibly pre-empt existing LSPs if bandwidth reservations can no longer accommodate them.

EXAMPLE:

```
NetIron(config-mpls)# mpls-interface pos 3/11
NetIron(config-mpls-interface)# reservable-bw 10000
```

Syntax: reservable-bw <number>

Possible values: The reservable bandwidth is expressed in Kbits/sec. If the amount of reservable bandwidth is greater than the maximum available bandwidth, then the link can be oversubscribed. If the reservable bandwidth is less than the maximum available bandwidth, then LSPs cannot reserve all physical bandwidth on the interface.

Default value: By default, the reservable bandwidth is the same as the maximum available bandwidth on the interface.

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
NetIron(config-mpls)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
NetIron(config-mpls)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 24
MPLS Policy Commands

admin-group

Configures administrative groups for MPLS interfaces. When a Foundry device calculates the path for an LSP, it can take into account the administrative group to which a interface belongs. You can refer to an administrative group either by name or number.

EXAMPLE:

```
NetIron(config-mpls)# policy
NetIron(config-mpls-policy)# admin-group gold 30
NetIron(config-mpls-policy)# admin-group silver 20
NetIron(config-mpls-policy)# admin-group bronze 10
```

Syntax: [no] admin-group <name> <number>

Possible values: The <number> can be from 0 – 31. Up to 32 administrative groups can be configured on the Foundry device.

Default value: N/A

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
NetIron(config-mpls)# end
NetIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
NetIron(config-mpls)# exit
NetIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

no

Disables other commands. To disable a command, place the word **no** before the command.

propagate-ttl

Enables or disables TTL propagation for packets travelling through an LSP. When TTL propagation is disabled, the entire MPLS domain appears as a single hop. In this case, the ingress LER places a value of 255 into the packet's MPLS TTL field, regardless of the TTL value in the packet's IP header. When the MPLS label is popped, the LSR does not change the TTL value in the IP header.

EXAMPLE:

```
NetIron(config-mpls)# policy
NetIron(config-mpls-policy)# no propagate-ttl
```

Syntax: [no] propagate-ttl

Possible values: N/A

Default value: By default, TTL propagation is enabled.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
NetIron(config-mpls)# quit
NetIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

retry-limit

Sets the number of times the ingress LER attempts to connect to the egress LER in a signalled LSP before declaring the path failed. If the ingress LER fails to connect to the egress LER in a signalled LSP, it will keep trying to make the connection indefinitely. The **retry-limit** command sets a limit for these connection attempts. After this limit is exceeded, the ingress LER stops trying to connect to the egress LER over the primary path.

If a secondary path is configured for the LSP, it is immediately activated once the primary path fails. After the secondary path is activated, the ingress LER continues to try to connect to the egress LER over the primary path, either up to the configured retry limit or indefinitely if no retry limit is set. If a connection over the primary path can be established, the secondary path is deactivated, and traffic for the LSP is again sent over the primary path.

EXAMPLE:

To set the number of connection attempts to 20:

```
NetIron(config-mpls)# policy
NetIron(config-mpls-policy)# retry-limit 20
```

Syntax: retry-limit <number>

Possible values: Number of retry attempts

Default value: By default, the ingress LER attempts to connect to the egress LER indefinitely.

retry-time

Sets the amount of time the ingress LER waits between connection attempts when it tries to connect to the egress LER. When a signalled LSP is enabled, the ingress LER attempts to connect to the egress LER over the primary path specified in the LSP's configuration. If the connection is not successful, by default the ingress LER waits 30 seconds before attempting the connection again. You can configure the amount of time the ingress LER waits between connection attempts.

EXAMPLE:

For example, to change the retry time to 45 seconds:

```
NetIron(config-mpls)# policy
NetIron(config-mpls-policy)# retry-time 45
```

Syntax: retry-time <seconds>

Possible values: 1 – 600 seconds.

Default value: 30 seconds.

show

Displays a variety of configuration and statistical information about the device. See "Show Commands" on page 43-1.

traffic-engineering ospf

Causes the Foundry device to send out OSPF-TE LSAs for its MPLS-enabled interfaces.

EXAMPLE:

```
NetIron(config-mpls)# policy
NetIron(config-mpls-policy)# traffic-engineering ospf
```

Syntax: [no] traffic-engineering ospf

Possible values: N/A.

Default value: By default, the Foundry device does not send out OSPF-TE LSAs for its MPLS-enabled interfaces.

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
NetIron(config-mpls)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
NetIron(config-mpls)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 25
MPLS Path Commands

delete

Deletes a node from an existing path.

EXAMPLE:

```
NetIron(config-mpls)# path sf_to_sj
NetIron(config-mpls-path)# delete loose 64.1.1.1
NetIron(config-mpls-path)# exit
```

Syntax: delete strict | loose <ip address>

Possible values: Strict or loose node in a path.

Default value: N/A

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
NetIron(config-mpls)# end
NetIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
NetIron(config-mpls)# exit
NetIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

insert

Inserts a node into an existing path.

EXAMPLE:

```
NetIron(config-mpls)# path sf_to_sj
NetIron(config-mpls-path)# insert strict 216.150.1.1 before 216.150.1.2
NetIron(config-mpls-path)# exit
```

Syntax: insert strict | loose <ip address> before <ip address>

Possible values: Strict or loose node in a path.

Default value: N/A

loose

Adds a loose node to a path. A loose node is one that is not directly connected to the preceding node in the path.

EXAMPLE:

```
NetIron(config-mpls)# path sf_to_sj
NetIron(config-mpls-path)# loose 64.1.1.1
NetIron(config-mpls-path)# exit
```

Syntax: [no] loose <ip address>

Possible values: Address of an LSR

Default value: N/A

no

Disables other commands. To disable a command, place the word **no** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
NetIron(config-mpls)# quit

NetIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

strict

Adds a strict node to a path. A strict node is one that is directly connected to the preceding node in the path.

EXAMPLE:

```
NetIron(config-mpls)# path sf_to_sj
NetIron(config-mpls-path)# strict 64.1.1.1
NetIron(config-mpls-path)# exit
```

Syntax: [no] strict <ip address>

Possible values: Address of an LSR.

Default value: N/A

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
NetIron(config-mpls)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
NetIron(config-mpls)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 26
MPLS Signalled LSP Commands

cos

Assigns a CoS (Class of Service) value to all packets traveling through a signalled LSP.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# cos 7
```

Syntax: cos <number>

Possible values: The CoS value can be an integer from 0 – 7.

Default value: N/A

cspf

Enables or disables the CSPF (Constrained Shortest Path First) process for selecting a path through an MPLS domain.

EXAMPLE:

To disable constraint-based path selection for LSP tunnel1:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# no cspf
```

Syntax: [no] cspf

Possible values: N/A

Default value: CSPF is enabled by default.

disable

Disables a signalled LSP. Disabling an LSP de-activates it, but does not remove the LSP from the device's configuration. (To remove a signalled LSP from the device's configuration, use the **no lsp** command.)

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# disable
```

Syntax: disable

Possible values: N/A

Default value: N/A

enable

Enables a signalled LSP. Enabling the LSP causes the path to be set up and resources reserved on the LSRs in the LSP's primary path. Enabling the LSP is the final step in configuring it.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# enable
```

Syntax: enable

Possible values: N/A

Default value: N/A

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
NetIron(config-mpls)# end
NetIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exclude-any

Excludes specified administrative groups from LSP calculations. Only interfaces that are not part of the specified groups can be considered for the LSP.

EXAMPLE:

To exclude interfaces in either administrative group "gold" or "silver" when the path for LSP tunnel1 is calculated:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# exclude-any gold silver
```

Syntax: [no] exclude-any <groups>

Possible values: Administrative group name(s)

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
NetIron(config-mpls)# exit
NetIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

from

Specifies a source IP address for a signalled LSP. The specified IP source address is carried in RSVP Path messages.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# from 1.2.3.4
```


Syntax: from <ip address>

Possible values: IP address

Default value: If the **from** command is not specified in an LSP's configuration, then when the LSP is enabled, the Foundry device dynamically determines the source address of the LSP (using the device's router ID or the address of the first loopback as the source address).

hop-limit

Changes the maximum number of hops that can be used in a traffic-engineered path.

EXAMPLE:

For example, to limit CSPF to choosing a path consisting of no more than 20 hops for LSP tunnel1:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# hop-limit 20
```

Syntax: [no] hop-limit <number>

Possible values: 1 – 255

Default value: 255 hops, including the ingress and egress LERs.

include-all

Includes the specified administrative groups in LSP calculations. An interface must be a member of all the groups specified in the **include-all** command in order to be considered for the LSP.

EXAMPLE:

To specify that an interface must be a member of both the "gold" or "silver" administrative groups in order to be included in the path calculations for LSP tunnel1:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# include-all gold silver
```

Syntax: [no] include-all <groups>

Possible values: Administrative group name(s)

Default value: N/A

include-any

Includes the specified administrative groups in LSP calculations. An interface must be a member of one of the groups specified in the **include-any** command in order to be considered for the LSP. Interfaces that are not part of any of the specified groups, as well as interfaces that are not part of any group, are eliminated from consideration.

EXAMPLE:

For example, to include interfaces in either administrative group "gold" or "silver" in the path calculations for LSP tunnel1:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# include-any gold silver
```

Syntax: [no] include-any <groups>

Possible values: Administrative group name(s)

Default value: N/A

install

Configures aliases for the egress LER in a signalled LSP. Traffic whose BGP next hop matches one of the aliases is forwarded along the LSP. When the traffic arrives at the egress LER, it can be forwarded to the actual BGP next hop.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# install 1.1.1.1/16
```

Syntax: install <ip address/mask>

Possible values: IP address and mask. Up to 12 aliases can be specified per LSP.

Default value: N/A

metric

Assigns a metric to the signalled LSP, which can be used by routing protocols to determine the relative preference of the LSP compared to other hop-by-hop routes. An LSP metric can also be used to determine preference among several LSPs towards a destination.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# metric 5
```

Syntax: metric <number>

Possible values: 1 – 65535. A lower metric is preferred over a higher one. If there are multiple LSPs to the same destination LSR, and they share the same metric, the traffic load is shared among them.

Default value: 1

no

Disables other commands. To disable a command, place the word **no** before the command.

primary-path

Specifies the name of the user-defined path to be used as the primary path for the LSP.

EXAMPLE:

To specify the sf_to_sj path as the primary path for LSP tunnel1:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# primary-path sf_to_sj
```

Syntax: primary-path <path name>

Possible values: Name of a user-defined path (set with the **path** command).

Default value: If you do not specify a primary path, the path used in the LSP is the shortest path to the egress LER, as determined from standard IP routing methods, or CSPF if it is enabled.

priority

Specifies the setup and hold priorities for the LSP. The setup priority is the priority for taking resources; the hold priority is the priority for holding resources. An LSP's setup priority is considered during admission control, and its hold priority is considered when bandwidth is allocated to the LSP.

EXAMPLE:

To configure LSP tunnel1 with a setup priority of 6 and hold priority of 1.

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# priority 6 1
```

Syntax: priority <setup-priority> <hold-priority>

Possible values: 0 (highest priority) through 7 (lowest priority). An LSP's setup priority must be lower than or equal to its hold priority.

Default value: By default, an LSP's setup priority is 7 and its hold priority is 0.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
NetIron(config-mpls)# quit
NetIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

record

Enables or disables LSP path recording with the RSVP RECORD_ROUTE object (RRO). An RRO consists of a series of subobjects that can contain the addresses of the LSRs in the path. This information can be viewed with the **show mpls lsp detail** command.

EXAMPLE:

To disable path recording in the RRO:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# no record
```

Syntax: [no] record

Possible values: N/A

Default value: The path information is recorded in the RRO by default.

secondary-path

Specifies the name of a user-defined path to be used as the secondary path for the LSP and enters the secondary path configuration level. See “MPLS Secondary Path Commands” on page 27-1 for a description of commands available at this configuration level.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# secondary-path alt_sf_to_sj
NetIron(config-mpls-lsp-sec-path)#
```

Syntax: secondary-path <path name>

Possible values: Name of a user-defined path (set with the **path** command).

Default value: N/A

shortcuts

Configures the signalled LSP to serve as an IGP shortcut between nodes in an AS. When an LSP is configured to be an IGP shortcut, OSPF includes the LSP in its SPF calculation.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# shortcuts ospf
```

Syntax: [no] shortcuts ospf

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

tie-breaking

Specifies how CSPF chooses a path from among several equal cost paths to a destination.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# tie-breaking least-fill
```

Syntax: [no] tie-breaking least-fill | most-fill | random

The **least-fill** parameter causes CSPF to choose the path with the highest available bandwidth (that is, the path with the least utilized links).

The **most-fill** parameter causes CSPF to choose the path with the lowest available bandwidth (that is, the path with the most utilized links).

The **random** parameter causes CSPF to choose the path randomly from among the equal-cost paths. This is the default.

Possible values: See above.

Default value: random

to

Specifies the destination (egress LER) for a signalled LSP.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# to 64.100.1.1
```

Syntax: to <ip address>

Possible values: IP address of the destination LER.

Default value: N/A

traffic-eng max-rate

Sets the maximum rate of packets that can go through the LSP (in Kbits/sec).

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# traffic-eng max-rate 20
```

Syntax: traffic-eng max-rate <rate>

Possible values: <rate> is specified in Kbits/sec.

Default value: N/A

traffic-eng mean-rate

Sets the average rate of packets that can go through the LSP (in Kbits/sec).

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# traffic-eng mean-rate 20
```

Syntax: traffic-eng mean-rate <rate>

Possible values: <rate> is specified in Kbits/sec.

Default value: N/A

traffic-eng max-burst

Sets the maximum size (in bytes) of the largest burst that can be sent over the LSP at the maximum rate.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# traffic-eng max-burst 10
```

Syntax: traffic-eng max-burst <bytes>

Possible values:

Default value: N/A

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
NetIron(config-mpls)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
NetIron(config-mpls)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 27
MPLS Secondary Path Commands

cos

Assigns a CoS (Class of Service) value to all packets traveling through a signalled LSP.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# secondary-path alt_sf_to_sj
NetIron(config-mpls-lsp-sec-path)# cos 7
```

Syntax: cos <number>

Possible values: The CoS value can be an integer from 0 – 7.

Default value: N/A

cspf

Enables or disables the CSPF (Constrained Shortest Path First) process for selecting a path through an MPLS domain.

EXAMPLE:

To disable constraint-based path selection for LSP tunnel1:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# secondary-path alt_sf_to_sj
NetIron(config-mpls-lsp-sec-path)# no cspf
```

Syntax: [no] cspf

Possible values: N/A

Default value: CSPF is enabled by default.

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
NetIron(config-mpls)# end
NetIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exclude-any

Excludes specified administrative groups from LSP calculations. Only interfaces that are not part of the specified groups can be considered for the LSP.

EXAMPLE:

To exclude interfaces in either administrative group “gold” or “silver” when the secondary path for LSP tunnel1 is calculated:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# secondary-path alt_sf_to_sj
NetIron(config-mpls-lsp-sec-path)# exclude-any gold silver
```

Syntax: [no] exclude-any <groups>

Possible values: Administrative group name(s)

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the LSP level.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# secondary-path alt_sf_to_sj
NetIron(config-mpls-lsp-sec-path)# exit
```

Syntax: exit

Possible values: N/A

Default value: N/A

hop-limit

Changes the maximum number of hops that can be used in a traffic-engineered path.

EXAMPLE:

For example, to limit CSPF to choosing a path consisting of no more than 20 hops for LSP tunnel1’s secondary path:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# secondary-path alt_sf_to_sj
NetIron(config-mpls-lsp-sec-path)# hop-limit 20
```

Syntax: [no] hop-limit <number>

Possible values: 1 – 255

Default value: 255 hops, including the ingress and egress LERs.

include-all

Includes the specified administrative groups in LSP calculations. An interface must be a member of all the groups specified in the **include-all** command in order to be considered for the LSP.

EXAMPLE:

To specify that an interface must be a member of both the “gold” or “silver” administrative groups in order to be included in the path calculations for LSP tunnel1’s secondary path:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# secondary-path alt_sf_to_sj
NetIron(config-mpls-lsp-sec-path)# include-all gold silver
```

Syntax: [no] include-all <groups>

Possible values: Administrative group name(s)

Default value: N/A

include-any

Includes the specified administrative groups in LSP calculations. An interface must be a member of one of the groups specified in the **include-any** command in order to be considered for the LSP. Interfaces that are not part of any of the specified groups, as well as interfaces that are not part of any group, are eliminated from consideration.

EXAMPLE:

For example, to include interfaces in either administrative group “gold” or “silver” in the path calculations for LSP tunnel1’s secondary path:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# secondary-path alt_sf_to_sj
NetIron(config-mpls-lsp-sec-path)# include-any gold silver
```

Syntax: [no] include-any <groups>

Possible values: Administrative group name(s)

Default value: N/A

no

Disables other commands. To disable a command, place the word **no** before the command.

priority

Specifies the setup and hold priorities for the LSP’s secondary path. The setup priority is the priority for taking resources; the hold priority is the priority for holding resources. An LSP’s setup priority is considered during admission control, and its hold priority is considered when bandwidth is allocated to the LSP.

EXAMPLE:

To configure LSP tunnel1’s secondary path with a setup priority of 6 and hold priority of 1.

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# secondary-path alt_sf_to_sj
NetIron(config-mpls-lsp-sec-path)# priority 6 1
```

Syntax: priority <setup-priority> <hold-priority>

Possible values: 0 (highest priority) through 7 (lowest priority). An LSP’s setup priority must be lower than or equal to its hold priority.

Default value: By default, an LSP’s setup priority is 7 and its hold priority is 0.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# secondary-path alt_sf_to_sj
NetIron(config-mpls-lsp-sec-path)# quit
NetIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

record

Enables or disables LSP path recording with the RSVP RECORD_ROUTE object (RRO). An RRO consists of a series of subobjects that can contain the addresses of the LSRs in the path. This information can be viewed with the **show mpls lsp detail** command.

EXAMPLE:

To disable path recording in the RRO:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# secondary-path alt_sf_to_sj
NetIron(config-mpls-lsp-sec-path)# no record
```

Syntax: [no] record

Possible values: N/A

Default value: The path information is recorded in the RRO by default.

show

Displays a variety of configuration and statistical information about the device. See "Show Commands" on page 43-1.

standby

Specifies that the secondary path is to serve as a hot standby path. Once the LSP is enabled, both the primary and hot-standby paths are activated, although packets are directed over only the primary path.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# secondary-path alt_sf_to_sj
NetIron(config-mpls-lsp-sec-path)# standby
```

Syntax: standby

Possible values: N/A

Default value: N/A

traffic-eng max-rate

Sets the maximum rate of packets that can go through the LSP (in Kbits/sec).

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# secondary-path alt_sf_to_sj
NetIron(config-mpls-lsp-sec-path)# traffic-eng max-rate 20
```

Syntax: traffic-eng max-rate <rate>

Possible values: <rate> is specified in Kbits/sec.

Default value: N/A

traffic-eng mean-rate

Sets the average rate of packets that can go through the LSP (in Kbits/sec).

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# secondary-path alt_sf_to_sj
NetIron(config-mpls-lsp-sec-path)# traffic-eng mean-rate 20
```

Syntax: traffic-eng mean-rate <rate>

Possible values: <rate> is specified in Kbits/sec.

Default value: N/A

traffic-eng max-burst

Sets the maximum size (in bytes) of the largest burst that can be sent over the LSP at the maximum rate.

EXAMPLE:

```
NetIron(config-mpls)# lsp tunnel1
NetIron(config-mpls-lsp)# secondary-path alt_sf_to_sj
NetIron(config-mpls-lsp-sec-path)# traffic-eng max-burst 10
```

Syntax: traffic-eng max-burst <bytes>

Possible values:

Default value: N/A

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
NetIron(config-mpls)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
NetIron(config-mpls)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 28
MPLS RSVP Commands

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
NetIron(config-mpls)# end
NetIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
NetIron(config-mpls)# exit
NetIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

no

Disables other commands. To disable a command, place the word **no** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
NetIron(config-mpls)# quit
NetIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

refresh-interval

Controls how often RSVP Path and Resv messages are sent. To maintain path states and resource reservations on the routers in an LSP, RSVP Path and Resv messages are sent at regular intervals. Path messages flow downstream in an LSP, from the ingress LER towards the egress LER. Resv messages flow upstream, in the reverse direction of Path messages.

EXAMPLE:

To set the refresh interval to 20 seconds

```
NetIron(config-mpls)# rsvp
NetIron(config-mpls-rsvp)# refresh-interval 20
```

Syntax: refresh-interval <seconds>

Possible values: 0 – 2147483 seconds

Default value: 30 seconds

refresh-multiple

Sets the number of refresh intervals that must elapse without a refresh message before a path state or resource reservation times out. If refresh messages are not received, RSVP path states and resource reservations are removed from the routers in an LSP.

EXAMPLE:

To set the refresh multiple to 5 intervals:

```
NetIron(config-mpls)# rsvp
NetIron(config-mpls-rsvp)# refresh-multiple 5
```

Syntax: refresh-multiple <intervals>

Possible values: 0 – 65535 intervals

Default value: 3 intervals

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
NetIron(config-mpls)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
NetIron(config-mpls)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 29
MPLS Static LSP Commands

cos

Assigns a CoS (Class of Service) value to all packets traveling through a static LSP.

EXAMPLE:

```
NetIron(config-mpls)# lsp static_tunnel
NetIron(config-mpls-static-lsp)# cos 7
```

Syntax: cos <number>

Possible values: The CoS value can be an integer from 0 – 7.

Default value: N/A

disable

Disables a static LSP. Disabling an LSP de-activates it, but does not remove the LSP from the device's configuration. (To remove a static LSP from the device's configuration, use the **no static-lsp** command.)

EXAMPLE:

```
NetIron(config-mpls)# static-lsp static_tunnel
NetIron(config-mpls-static-lsp)# disable
```

Syntax: disable

Possible values: N/A

Default value: N/A

enable

Enables a static LSP. You should enable the static LSP on the transit and egress LERs prior to enabling the static LSP on the ingress LER.

EXAMPLE:

```
NetIron(config-mpls)# static-lsp static_tunnel
NetIron(config-mpls-static-lsp)# enable
```

Syntax: enable

Possible values: N/A

Default value: N/A

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
NetIron(config-mpls)# end
NetIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
NetIron(config-mpls)# exit
NetIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

in-segment

Specifies an inbound interface and inbound label for a static LSP.

EXAMPLE:

```
NetIron(config-mpls)# static-lsp static_tunnel
NetIron(config-mpls-static-lsp)# in-segment pos 2/1 in-label 123
```

Syntax: in-segment pos <interface> in-label <label>

Possible values: N/A

Default value: N/A

install

Configures aliases for the egress LER in a static LSP. Traffic whose BGP next hop matches one of the aliases is forwarded along the LSP. When the traffic arrives at the egress LER, it can be forwarded to the actual BGP next hop.

EXAMPLE:

```
NetIron(config-mpls)# static-lsp static_tunnel
NetIron(config-mpls-static-lsp)# install 1.1.1.1/16
```

Syntax: install <ip address/mask>

Possible values: IP address and mask. Up to 12 aliases can be specified per LSP.

Default value: N/A

metric

Assigns a metric to the static LSP, which can be used by routing protocols to determine the relative preference of the LSP compared to other hop-by-hop routes. An LSP metric can also be used to determine preference among several LSPs towards a destination.

EXAMPLE:

```
NetIron(config-mpls)# static-lsp static_tunnel
NetIron(config-mpls-static-lsp)# metric 5
```

Syntax: metric <number>

Possible values: 1 – 65535. A lower metric is preferred over a higher one. If there are multiple LSPs to the same destination LSR, and they share the same metric, the traffic load is shared among them.

Default value: 1

no

Disables other commands. To disable a command, place the word **no** before the command.

out-segment

Specifies an outbound interface and outbound label for a static LSP.

EXAMPLE:

```
NetIron(config-mpls)# static-lsp static_tunnel
NetIron(config-mpls-static-lsp)# out-segment pos 3/1 out-label 456
```

Syntax: out-segment pos <interface> out-label <label>

Syntax: out-segment ethernet <interface> out-label <label> next-hop <address>

Possible values: The **next-hop** parameter only applies to static LSPs whose outbound interface is an Ethernet interface.

Default value: N/A

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
NetIron(config-mpls)# quit
NetIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

to

Specifies the destination (egress LER) for a static LSP. A static LSP has one and only one destination. The egress LER is the only required parameter in an LSP.

EXAMPLE:

```
NetIron(config-mpls)# static-lsp static_tunnel
NetIron(config-mpls-static-lsp)# to 64.100.1.1
```

Syntax: to <ip address>

Possible values: IP address of the destination LER.

Default value: N/A

traffic-eng max-rate

Sets the maximum rate of packets that can go through the static LSP (in Kbits/sec).

EXAMPLE:

```
NetIron(config-mpls)# static-lsp static_tunnel
NetIron(config-mpls-static-lsp)# traffic-eng max-rate 20
```

Syntax: traffic-eng max-rate <rate>

Possible values: <rate> is specified in Kbits/sec.

Default value: N/A

traffic-eng mean-rate

Sets the average rate of packets that can go through the static LSP (in Kbits/sec).

EXAMPLE:

```
NetIron(config-mpls)# static-lsp static_tunnel
NetIron(config-mpls-static-lsp)# traffic-eng mean-rate 20
```

Syntax: traffic-eng mean-rate <rate>

Possible values: <rate> is specified in Kbits/sec.

Default value: N/A

traffic-eng max-burst

Sets the maximum size (in bytes) of the largest burst that can be sent over the LSP at the maximum rate.

EXAMPLE:

```
NetIron(config-mpls)# static-lsp static_tunnel
NetIron(config-mpls-static-lsp)# traffic-eng max-burst 10
```

Syntax: traffic-eng max-burst <bytes>

Possible values:

Default value: N/A

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
NetIron(config-mpls)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
NetIron(config-mpls)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 30
MPLS VLL Commands

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
NetIron(config-mpls)# end
NetIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
NetIron(config-mpls)# exit
NetIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

no

Disables other commands. To disable a command, place the word **no** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
NetIron(config-mpls)# quit
NetIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

untagged

Specifies an untagged endpoint for a VLL:

EXAMPLE:

```
NetIron(config-mpls)# vll foundry-sf-to-sj 40000
NetIron(config-mpls-vll)# untagged e 2/1
```

Syntax: untagged [pos | ethernet] <portnum>

Possible values: Ethernet or POS port number

Default value: N/A

vlan

Specifies a <vlan-id, port> endpoint for a VLL:

EXAMPLE:

```
NetIron(config-mpls-vll)# vlan 200
NetIron(config-mpls-vll-vlan)# tagged e 3/11
```

Syntax: vlan <num>

Syntax: tagged [pos | ethernet] <slot/port>

NOTE: Within a port group, a VLAN ID cannot be used for more than one purpose.

- On a 10/100 module, a port group is defined as all the ports on the module.
- On a Gigabit Ethernet module, the eight ports are divided into two port groups, corresponding to ports 1 – 4 and ports 5 – 8.

If a VLAN ID is already used by a regular VLAN, it cannot be assigned to a VLL, and vice-versa.

If a VLAN ID is already assigned to a VLL on one port, it cannot be assigned to another VLL on another port within the port group.

Possible values: VLAN ID (see above), Ethernet or POS port number

Default value: N/A

vll-peer

Specifies the peer at the other end of a VLL.

EXAMPLE:

```
NetIron(config-mpls)# vll foundry-sf-to-sj 40000
NetIron(config-mpls-vll)# vll-peer 192.168.2.100
```

Syntax: vll-peer <ip-addr> [<static-local-vc-label> <static-remote-vc-label>]

Possible values: The IP address of the peer must match that of a destination for a tunnel LSP configured on the device.

Static local and remote VC label values are optional. If configured, <static-local-vc-label> is the VC label value expected for packets forwarded to the local physical port from the VLL peer, and <static-remote-vc-label> is the VC label applied to packets sent to the remote VLL peer.

Acceptable values for <static-local-vc-label> are 800000 – 1048575. If the label value you specify has already been assigned, a message is displayed requesting a different value.

Default value: N/A

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
NetIron(config-mpls)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
NetIron(config-mpls)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 31
MPLS LDP Commands

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
NetIron(config-mpls)# end
NetIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
NetIron(config-mpls)# exit
NetIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

hello-interval

Controls how often the device sends out LDP Hello messages. Hello messages are used to maintain LDP sessions between the device and its LDP peers. You can set the interval for LDP Link Hello messages (LDP Hello messages multicast to all routers on the sub-net).

EXAMPLE:

```
NetIron(config-mpls)# ldp
NetIron(config-mpls-ldp)# hello-interval 10
```

Syntax: hello-interval <seconds>

Possible values: 1 – 65535 seconds

Default value: 5 seconds

hello-interval target

Controls how often the device sends out LDP Targeted Hello messages (LDP Hello messages unicast to a specific address, such as a VLL peer).

EXAMPLE:

```
NetIron(config-mpls)# ldp
NetIron(config-mpls-ldp)# hello-interval target 10
```

Syntax: hello-interval target <seconds>

Possible values: 1 – 65535 seconds

Default value: 5 seconds

hello-timeout

Specifies how long the device waits for its LDP peers to send a Hello message. If the device does not receive a Hello message within this time, the LDP session with the peer can be terminated. The device includes the hold time in the Hello messages it sends out to its LDP peers.

EXAMPLE:

```
NetIron(config-mpls)# ldp
NetIron(config-mpls-ldp)# hello-timeout 20
```

Syntax: hello-timeout <seconds>

Possible values: 1 – 65535 seconds

Default value: 15 seconds

hello-timeout target

Specifies how long the device waits for its LDP peers to send a Targeted Hello messages.

EXAMPLE:

```
NetIron(config-mpls)# ldp
NetIron(config-mpls-ldp)# hello-timeout target 20
```

Syntax: hello-timeout target <seconds>

Possible values: 1 – 65535 seconds

Default value: 45 seconds

no

Disables other commands. To disable a command, place the word **no** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
NetIron(config-mpls)# quit
NetIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
NetIron(config-mpls)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
NetIron(config-mpls)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 32
MPLS VPLS Commands

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
NetIron(config-mpls)# end
NetIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
NetIron(config-mpls)# exit
NetIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

no

Disables other commands. To disable a command, place the word **no** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
NetIron(config-mpls)# quit
NetIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

untagged

Specifies an untagged endpoint for a VPLS instance:

EXAMPLE:

```
NetIron(config-mpls)# vpls v1 40000
NetIron(config-mpls-vpls-v1)# vlan 100
NetIron(config-mpls-vpls-v1-vlan-100)# untagged e 2/1
```

Syntax: untagged [pos | ethernet] <portnum>

Possible values: Ethernet or POS port number

Default value: N/A

vlan

Specifies a <vlan-id, port> endpoint for a VPLS instance.

EXAMPLE:

```
NetIron(config-mpls-vpls-v1)# vlan 200
NetIron(config-mpls-vpls-v1-vlan-200)# tagged e 3/11
```

Syntax: vlan <num>

Syntax: tagged [pos | ethernet] <slot/port>

Possible values: VLAN ID, Ethernet or POS port number. A VPLS instance can have multiple ports configured under the same VLAN ID, and can have ports configured under different VLAN IDs. The ports can belong to the same or different port groups. Another VPLS instance can reuse the same VLAN ID, if the ports belong to different port groups. Since the VLANs are configured under different VPLS instances, they are different VPLS VLANs, even though they use the same VLAN ID.

As with regular VLANs, if a port is currently a member of a non-default VLAN as an untagged port, it must be returned to the default VLAN before it can be assigned to a VPLS instance as a tagged port.

Default value: N/A

vpls-peer

Specifies the IP addresses of the remote VPLS peers within a VPLS instance.

EXAMPLE:

To specify three remote VPLS peers within a VPLS instance

```
NetIron(config-mpls-vpls-v1)# vpls-peer 192.168.2.100 192.168.2.101 192.168.2.102
```

Syntax: vpls-peer <ip-addr> [<ip-addr> ...]

Possible values: The IP address of each peer must match that of a destination for a tunnel LSP configured on the device; that is, the device must have at least one operational RSVP- or LDP-signalled LSP with the destination (the LSP’s “to” address) matching the VPLS peer’s IP address. An LSP terminating on the VPLS peer but configured with a different destination address would not be considered a match.

Default value: N/A

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
NetIron(config-mpls)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
NetIron(config-mpls)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 33
VLAN Commands

add-vlan

Adds a VLAN to a VLAN group.

NOTE: This command applies only to VLAN groups. See “vlan-group” on page 6-140.

EXAMPLE:

```
BigIron(config-vlan-group-1)# add-vlan 1001 to 1002
```

Syntax: add-vlan <vlan-id> [to <vlan-id>]

Possible values: Valid VLAN IDs

Default value: N/A

appletalk-cable-vlan

Configures an AppleTalk cable VLAN within a port-based VLAN.

EXAMPLE:

To configure AppleTalk cable VLAN 1 in port-based VLAN 10, then configure the routing parameters for the VLAN, enter the following commands.

```
BigIron(config-vlan-10)# appletalk-cable-vlan 1 name cable-one
BigIron(config-vlan-10)# static ethe 2/1 ethe 3/1 to 3/2
BigIron(config-vlan-10)# router-interface ve 1
BigIron(config-vlan-10)# interface ve 1
BigIron(config-vif-1)# appletalk cable-range 10 - 19
BigIron(config-vif-1)# appletalk address 10.1
BigIron(config-vif-1)# appletalk zone-name AA
BigIron(config-vif-1)# appletalk routing
```

Syntax: appletalk-cable-vlan <vlan-id> [name <string>]

The <vlan-id> can be from 1 – 8.

The **name** <string> parameter specifies a name and can be a string up to 32 characters long.

Possible values: VLAN ID 1 – 8; name up to 32 characters long

Default value: N/A

atalk-proto

Creates an AppleTalk protocol VLAN within a port-based VLAN when entered at the VLAN Level. All ports are assumed by default to be members of the VLAN when initially created. Protocol VLAN membership can be modified using the **dynamic**, **static**, or **exclude** commands.

EXAMPLE:

To create an AppleTalk protocol VLAN with permanent port membership of 9 and 13 (module 3) and no dynamic ports within an already defined port-based VLAN 2, enter the following commands.

```
BigIron(config)# vlan 2

BigIron(config-vlan-2)# atalk-proto

BigIron(config-vlan-atalk-proto)# static e 3/9 e 3/13

BigIron(config-vlan-atalk-proto) no dynamic
```

NOTE: If configuring this on a Layer 2 Switch, enter **vlan 2 by port** at the CONFIG Level versus **vlan 2**, as shown in the example above.

Syntax: atalk-proto [name <string>]

The name can be up to 16 characters long and can contain blanks. The name appears in VLAN show displays.

To specify a VLAN name, use the **name** keyword followed by a string. The **name** keyword and string are the last arguments in the command. For example, to name an AppleTalk VLAN, enter the following command:

```
BigIron(config)# atalk-proto name "Ship and Recv"
```

This example shows how to specify a name that contains a blank. Use double quotation marks before and after the name.

Possible values: N/A

Default value: N/A

decnet-proto

Creates a Decnet protocol VLAN within a port-based VLAN, when entered at the VLAN Level. All ports are assumed by default to be members of the VLAN when initially created. Protocol VLAN membership can be modified using the **dynamic**, **static**, or **exclude** commands.

EXAMPLE:

To create a Decnet protocol VLAN with permanent port membership of 15 and 16 with port 17 as dynamic member port (module 3), within VLAN 5, enter the following commands.

```
BigIron(config)# vlan 5

BigIron(config-vlan-5)# decnet-proto

BigIron(config-vlan-decnet-proto)# exclude e 3/1 to 3/14 e 3/18
```

NOTE: If configuring this on a Layer 2 Switch, enter **vlan 5 by port** at the CONFIG Level versus **vlan 5**, as shown in the example above.

Syntax: decnet-proto [name <string>]

The name can be up to 16 characters long and can contain blanks. The name appears in VLAN show displays.

To specify a VLAN name, use the **name** keyword followed by a string. The **name** keyword and string are the last arguments in the command. The name can contain blank spaces if you use double quotation marks before and after the name.

Possible values: N/A

Default value: N/A

default-gateway

Configures default gateways for the designated VLAN. You can configure up to five default gateways for the designated VLAN, and associate a metric with each one. The software uses the gateway with the lowest metric. The other gateways reside in the configuration but are not used. You can use one of the other gateways by modifying the configuration so that the gateway you want to use has the lowest metric.

If more than one gateway has the lowest metric, the software uses the gateway that appears first in the running-config.

NOTE: If you have already configured a default gateway globally and you do not configure a gateway in the VLAN, the software uses the globally configured gateway and gives the gateway a metric value of 1.

EXAMPLE:

```
BigIron(config)# vlan 10
BigIron(config-vlan-10)# untag ethernet 1/1 to 1/4
BigIron(config-vlan-10)# default-gateway 10.10.10.1 1
BigIron(config-vlan-10)# default-gateway 20.20.20.1 2
```

Syntax: [no] default-gateway <ip-addr> <metric>

Possible values:

The <ip-addr> parameter specifies the IP address of the gateway router.

The <metric> parameter specifies the metric (cost) of the gateway. You can specify a value from 1 – 5. There is no default. The software uses the gateway with the lowest metric.

Default value: N/A

default-vlan-id

When you enable port-based VLAN operation, all ports are assigned to VLAN 1 by default. As you create additional VLANs and assign ports to them, the ports are removed from the default VLAN. All ports that you do not assign to other VLANs remain members of default VLAN 1. This behavior ensures that all ports are always members of at least one VLAN.

You can change the VLAN ID for the default VLAN by entering the following command at the global CONFIG level of the CLI:

```
BigIron(config-vlan-2)# default-vlan-id 4095
```

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 – 4095.

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.

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged EXEC level, enter the following from any level of the CLI.

```
BigIron(config-vlan-decnet-proto)# end

BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the port-based VLAN level if configuring a protocol VLAN. If configuring a port-based VLAN, activity would be moved to the global level.

EXAMPLE:

```
BigIron(config-vlan-decnet-proto)# exit

BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

group-router-interface

Enables a VLAN group to use a virtual interface group.

NOTE: This command applies only to VLAN groups. See “vlan-group” on page 6-140.

EXAMPLE:

To configure a virtual interface group, enter commands such as the following:

```
BigIron(config)# vlan-group 1
BigIron(config-vlan-group-1)# group-router-interface
BigIron(config-vlan-group-1)# exit
BigIron(config)# interface group-ve 1
BigIron(config-vif-group-1)# ip address 10.10.10.1/24
```

These commands enable VLAN group 1 to have a group virtual interface, then configure virtual interface group 1. The software always associates a virtual 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 interface group also must have ID 1.

Syntax: group-router-interface

Possible values: N/A

Default value: N/A

ip-proto

Creates an IP protocol VLAN within a port-based VLAN, when entered at the VLAN Level.

When configuring on a Layer 2 Switch, all ports are dynamically allocated to the VLAN. You can modify port membership by using the **static** or **exclude** commands.

When configuring on a Layer 3 Switch, ports must be added to the VLAN with the **static** command if you configure routing information on the port. Otherwise, you can add ports dynamically.

EXAMPLE:

To assign ports 1, 2, 6 and 8 (module 2) to an IP protocol VLAN within VLAN 7, enter the following:

```
BigIron(config)# vlan 7

BigIron(config-vlan-7)# ip-proto

BigIron(config-vlan-ip-proto)# static e 3/1 to 3/2 e 3/6 e 3/8
```

NOTE: If configuring this on a Layer 2 Switch, enter **vlan 7 by port** at the CONFIG Level versus **vlan 7**, as shown in the example above.

NOTE: An IP protocol and IP sub-net VLAN cannot both be configured to operate on a Foundry device at the same time. This restriction is also true for IPX and IPX network VLANs.

Syntax: ip-proto [name <string>]

The name can be up to 16 characters long and can contain blanks. The name appears in VLAN show displays.

To specify a VLAN name, use the **name** keyword followed by a string. The **name** keyword and string are the last arguments in the command. The name can contain blank spaces if you use double quotation marks before and after the name.

Possible values: N/A

Default value: N/A

ip-subnet

Creates an IP sub-net protocol VLAN within a port-based VLAN, when entered at the VLAN Level. This allows you to define additional granularity than that of an IP protocol VLAN, by partitioning the broadcast domains by sub-net. In creating an IP sub-net VLAN, an IP address is used as identifier.

When configuring on a Layer 2 Switch, all ports are dynamically allocated to the VLAN. You can modify port membership by using the **static** or **exclude** commands.

When configuring on a Layer 3 Switch, ports must be added to the VLAN with the **static** command if you configure routing information on the port. Otherwise, you can add ports dynamically.

EXAMPLE:

To create an IP sub-net of IP address 192.75.3.0 with permanent port membership of 1 and 2 (module 2), within VLAN 10, enter the following commands.

```
BigIron(config)# vlan 10
BigIron(config-vlan-10)# ip-subnet 192.75.3.0 255.255.255.0
BigIron(config-vlan-ip-subnet)# static e 2/1 to 2/2
```

NOTE: If configuring this on a Layer 2 Switch, enter **vlan 10 by port** at the CONFIG Level versus **vlan 10**, as shown in the example above.

NOTE: An IP protocol and IP sub-net VLAN cannot both be configured to operate simultaneously on a Foundry device. This restriction is also true for IPX and IPX network VLANs.

Syntax: ip-subnet <ip-addr> <ip-mask> [<name>]

The name can be up to 16 characters long and can contain blanks. The name appears in VLAN show displays.

To specify a VLAN name, use the **name** keyword followed by a string. The **name** keyword and string are the last arguments in the command. The name can contain blank spaces if you use double quotation marks before and after the name.

Possible values: N/A

Default value: N/A

ipv6-proto

Configures 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.

EXAMPLE:

```
BigIron(config)# vlan 2
BigIron(config-vlan-2)# untag ethernet 1/1 to 1/8
BigIron(config-vlan-2)# ipv6-proto name V6
```



```
BigIron(config-ipv6-subnet)# static ethernet 1/1 to 1/6
BigIron(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>]

Possible values: See above

Default value: N/A

ipx-network

Creates an IPX network VLAN within a port-based VLAN, when entered at the VLAN Level. This allows you to define additional granularity than that of the IPX protocol VLAN, by partitioning the broadcast domains by IPX network number. In creating an IPX network VLAN, an IPX network number is used as identifier. The frame type must also be specified.

When configuring on a Layer 2 Switch, all ports are dynamically allocated to the VLAN. You can modify port membership by using the **static** or **exclude** commands.

When configuring on a Layer 3 Switch, ports must be added to the VLAN with the **static** command if you configure routing information on the port. Otherwise, you can add ports dynamically.

EXAMPLE:

To create an IPX network VLAN with a network number of 500 and frame type of 802.2 with permanent port membership of 10 and 14 (module 2) within port-based VLAN 15, enter the following commands.

```
BigIron(config)# vlan 15
BigIron(config-vlan-15)# ipx-network 500 ethernet_802.2
BigIron(config-vlan-ipx-proto)# static e 1/10 e 1/14
```

Syntax: ipx-network <ipx-network-number> <frame-type>

NOTE: If configuring this on a Layer 2 Switch, enter **vlan 15 by port** at the CONFIG Level versus **vlan 15**, as shown in the example above.

NOTE: An IPX network and IPX protocol VLAN cannot both be configured to operate simultaneously on a Foundry device. This restriction is also true for IP protocol and IP sub-net VLANs.

Possible values: Frame type: ethernet_ii, ethernet_802.2, ethernet_802.3, ethernet_snap

Default value: N/A

ipx-proto

Creates an IPX protocol VLAN within a port-based VLAN, when entered at the VLAN Level.

When configuring on a Layer 2 Switch, all ports are dynamically allocated to the VLAN. You can modify port membership by using the **static** or **exclude** commands.

When configuring on a Layer 3 Switch, ports must be added to the VLAN with the **static** command if you configure routing information on the port. Otherwise, you can add ports dynamically.

EXAMPLE:

To assign ports 1, 2, 6 and 8 (module 2) to an IPX protocol VLAN within port-based VLAN 22, enter the following:

```
BigIron(config)# vlan 22
BigIron(config-vlan-22)# ipx-proto
BigIron(config-vlan-ipx-proto)# static e 2/1 to 2/2 e 2/6 e 2/8
```


NOTE: If configuring this on a Layer 2 Switch, enter **vlan 22 by port** at the CONFIG Level versus **vlan 22**, as shown in the example above.

NOTE: An IPX protocol and IPX network VLAN cannot both be configured to operate simultaneously on a Foundry device. This restriction is also true for IP and IP sub-net VLANs.

Syntax: ipx-proto [<name>]

The name can be up to 16 characters long and can contain blanks. The name appears in VLAN show displays.

To specify a VLAN name, use the **name** keyword followed by a string. The **name** keyword and string are the last arguments in the command. The name can contain blank spaces if you use double quotation marks before and after the name.

Possible values: N/A

Default value: N/A

management-vlan

Makes a VLAN the designated management VLAN for the device. When you configure a VLAN to be the designated management VLAN, the management IP address you configure on the device is associated only with the ports in the designated VLAN. To establish a Telnet management session with the device, a user must access the device through one of the ports in the designated VLAN.

EXAMPLE:

```
BigIron(config)# vlan 10
BigIron(config-vlan-10)# untag ethernet 1/1 to 1/4
BigIron(config-vlan-10)# management-vlan
```

Syntax: [no] management-vlan

Possible values: N/A

Default value: N/A

metro-ring

Configures a metro ring for the Metro Ring Protocol (MRP).

EXAMPLE:

```
BigIron(config)# vlan 2
BigIron(config-vlan-2)# metro-ring 1
BigIron(config-vlan-2-mrp-1)# name CustomerA
BigIron(config-vlan-2-mrp-1)# master
BigIron(config-vlan-2-mrp-1)# ring-interface ethernet 1/1 ethernet 1/2
BigIron(config-vlan-2-mrp-1)# enable
```

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.

For information about the other commands in this example, see “Metro Ring Commands” on page 34-1.

Possible values: See above

Default value: N/A

multicast-flooding

Configures JetCore devices to perform hardware flooding for Layer 2 multicast and broadcast packets. Layer 2 multicast packets have a multicast address in the destination MAC address field.

EXAMPLE:

```
BigIron(config)#
BigIron(config)# vlan 2
BigIron(config-vlan-2)# multicast-flooding
BigIron(config-vlan-2)# exit
```

Syntax: multicast-flooding

After entering the **multicast-flooding** command for a VLAN, you must reboot the Foundry device to activate the feature.

Notes:

- This feature is supported only on JetCore devices and the 10 Gigabit Ethernet module.
- This feature cannot be enabled on an empty VLAN; the VLAN must already have ports assigned to it prior to enabling this feature.
- This feature is not supported on Layer 3 protocol-based VLANs.
- This feature is not supported on private VLANs.
- You cannot enable this feature on the designated management VLAN for the device.
- If you enable this feature on a VLAN that includes a trunk group, hardware flooding for Layer 2 multicast and broadcast packets occurs only on the trunk group's primary port. Multicast and broadcast traffic for the other ports in the trunk group is handled by software.

Possible values: N/A

Default value: N/A

netbios-proto

Creates a NetBIOS protocol VLAN within a port-based VLAN, when entered at the VLAN Level.

All ports are dynamically allocated to a NetBIOS VLAN when it is created. VLAN Membership can be modified using the **dynamic**, **static**, or **exclude** commands.

EXAMPLE:

To create a NetBIOS protocol VLAN with permanent port membership of 4 and 5 and ports 8 – 12 as dynamic member ports (module 2), within port-based VLAN 25, enter the following commands.

```
BigIron(config)# vlan 25
BigIron(config-vlan-25)# netbios-proto
BigIron(config-vlan-netbios-proto)# static e 2/4 e 2/5
BigIron(config- vlan-netbios-proto)# exclude e 2/1 to 2/3 e 2/6 e 2/7 e 2/13 to 2/18
```

NOTE: If configuring this on a Layer 2 Switch, enter **vlan 25 by port** at the CONFIG Level versus **vlan 25**, as show in the example above.

Syntax: netbios-proto [<name>]

The name can be up to 16 characters long and can contain blanks. The name appears in VLAN show displays.

To specify a VLAN name, use the **name** keyword followed by a string. The **name** keyword and string are the last arguments in the command. The name can contain blank spaces if you use double quotation marks before and after the name.

Possible values: N/A

Default value: N/A

no

Disables other commands. To disable a command, place the word ***no*** before the command.

other-proto

Creates an other-protocol VLAN within a port-based VLAN, when entered at the VLAN Level.

All ports of the device are by default dynamically assigned to a newly created "other-protocol" VLAN. VLAN Membership can be modified using the **dynamic**, **static**, or **exclude** commands.

You can use this option to define a protocol-based VLAN for protocols that do not require a singular protocol broadcast domain or are not currently supported on the Foundry device.

EXAMPLE:

On a 16-port device, ports 13 – 16 represent protocols Decnet and AppleTalk. You do not need to separate traffic by protocol into separate broadcast domains. Instead, create an other-protocol VLAN, with just those ports as members, within port-based VLAN 50.

```
BigIron(config)# vlan 50
BigIron(config-vlan-50)# other-proto
BigIron(config-vlan-other-proto)# static e 1/13 to 1/16
BigIron(config-vlan-other-proto)# exclude e 1/1 to 1/12
```

NOTE: If configuring this on a Layer 2 Switch, enter **vlan 50 by port** at the CONFIG Level versus **vlan 50**, as show in the example above.

Syntax: other-proto [<name>]

The name can be up to 16 characters long and can contain blanks. The name appears in VLAN show displays.

To specify a VLAN name, use the name keyword followed by a string. The **name** keyword and string are the last arguments in the command. The name can contain blank spaces if you use double quotation marks before and after the name.

Possible values: N/A

Default value: N/A

priority

This assigns a higher priority to a VLAN so that in times of congestion, it will receive precedence over other transmissions. Up to eight levels of priority can be assigned to a VLAN.

EXAMPLE:

```
BigIron(config)# vlan 25
BigIron(config-vlan-25)# priority 5
```

Here is the syntax on Chassis devices, the FastIron 4802, and the TurboIron/8:

Syntax: priority <0-7>

Here is the syntax on Stackable devices:

Syntax: priority high | normal

Possible values: See above

Default value: 0 or normal

pvlan mapping

Identifies the private VLANs for a primary private VLAN.

EXAMPLE:

To configure a primary private VLAN, enter commands such as the following:

```
BigIron(config)# vlan 7
BigIron(config-vlan-7)# untagged ethernet 3/2
BigIron(config-vlan-7)# pvlan type primary
BigIron(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: [no] pvlan mapping <vlan-id> ethernet <portnum>

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>).

Possible values: See above

Default value: None

pvlan type

Configures a private VLAN. 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.

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.

EXAMPLE:

To configure a community private VLAN, enter commands such as the following:

```
BigIron(config)# vlan 901
BigIron(config-vlan-901)# tagged ethernet 3/5 to 3/6
BigIron(config-vlan-901)# pvlan type community
```

These commands create port-based VLAN 901, add ports 3/5 and 3/6 to the VLAN as tagged ports, then specify that the VLAN is a community private VLAN.

To configure a primary private VLAN, enter commands such as the following:

```
BigIron(config)# vlan 7
BigIron(config-vlan-7)# untagged ethernet 3/2
BigIron(config-vlan-7)# pvlan type primary
BigIron(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: [no] pvlan type community | isolated | primary

The **pvlan type** command specifies that the 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.

Possible values: See above

Default value: None configured

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-ip-subnet)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

remove-vlan

Removes a VLAN from a VLAN group.

NOTE: This command applies only to VLAN groups. See “vlan-group” on page 6-140.

EXAMPLE:

```
BigIron(config-vlan-group-1)# remove-vlan 900 to 1000
```

Syntax: remove-vlan <vlan-id> [to <vlan-id>]

Possible values: Valid VLAN IDs

Default value: N/A

router-interface

Defines a router interface for a VLAN to allow traffic to be routed between VLANs.

EXAMPLE:

To configure a router interface for an IP sub-net VLAN, enter the following:

```
BigIron(config)# ip-subnet 192.75.3.0 255.255.255.0
BigIron(config-ip-subnet)# static e 5/1 to 5/3
BigIron(config-ip-subnet)# router-interface ve 3
```

NOTE: Once a router interface is assigned to a VLAN, it must be assigned an IP address at the interface level.

Syntax: router-interface ve <portnum>

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

spanning-tree

Spanning Tree bridge and port parameters are configurable using one command set at the global level for VLANs.

NOTE: On a device that has multiple port-based VLANs, you cannot configure STP parameters globally. Instead, you can configure the parameters on an individual spanning tree basis, in each port-based VLAN.

EXAMPLE:

Suppose you want to change the hello-time value of VLAN 3 from the default value. Enter the following commands:

```
BigIron(config)# vlan 3
BigIron(config-vlan-3)# span hello-time 8
```

NOTE: You do not need to configure values for the spanning tree parameters. All parameters have default values as noted below. Additionally, all values will be globally applied to all ports on the system or port-based VLAN for which they are defined.

Here is the syntax for global STP parameters.

Syntax: spanning-tree [forward-delay <value>] | [hello-time <value>] | [maximum-age <value>] | [priority <value>]

Here is the syntax for port STP parameters.

Syntax: spanning-tree atm | ethernet | pos <portnum> path-cost <value> | priority <value> | disable | enable

Possible values: see below

Bridge Parameters:

- Forward-delay: Possible values: 4 – 30 seconds. Default is 15 seconds.
- Max-age: Possible values: 6 – 40 seconds. Default is 20 seconds.
- Hello-time: Possible values: 1 – 10 seconds. Default is 2 seconds.
- Priority: Possible values: 1 – 65,535. Default is 32,768.

Port Parameters:

- Path: Possible values: 1 – 65,535. Default: The default depends on the port type:
  - 10 Mbps – 100
  - 100 Mbps – 19
  - Gigabit – 4
  - Gigabit – 2
  - OC-3c – 200
  - OC-12c – 80
  - OC-48c – 20
- Priority: possible values are 8 – 252, in increments of 4. Default is 128. A higher numerical value means a lower priority; thus, the highest priority is 8.

  NOTE: The range in software releases earlier than 07.5.01 is 0 – 255. 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.

- **disable | enable**: disables or enables STP on the specified port for this VLAN only. The STP state of the port in other VLANs is not affected.

Possible Values: See above

Default values: See above

spanning-tree 802-1w

Enables 802.1W on all ports of a device running the Spanning Tree Protocol. The 802.1W feature is Foundry's implementation of the IEEE 802.1W standard Rapid Spanning Tree Protocol (RSTP). Foundry's earlier implementation of RSTP, which was 802.1W Draft 3, provided only a subset of the IEEE 802.1W standard;

whereas, this implementation provides the full standard. This new implementation of the protocol is referred to as 802.1W. The implementation of the 802.1W Draft 3 is referred to as RSTP Draft 3. (See the command “spanning-tree single rstp” on page 6-129 to enable RSTP Draft 3.)

EXAMPLE:

To enable 802.1W on all ports in a port-based VLAN, enter commands such as the following:

```
BigIron(config)# vlan 10
BigIron(config-vlan-10)# spanning-tree 802-1w
```

Syntax: [no] spanning-tree 802-1w

Possible Values: N/A

Default values: Disabled

spanning-tree 802-1w <parameter>

Configures 802.1W bridge and port parameters once 802.1W has been enabled on the device. There are two versions of this command. The first one configures bridge parameters; the other configures port parameters.

Bridge Parameters:

Configuration changes using these parameters are applied to all ports on the bridge.

EXAMPLE:

To change 802.1W bridge parameters, enter a commands such as the following:

BigIron`(config)# spanning-tree 802-1w forward-delay 20 hello-time 3 max-age 10 priority 10`

The command in this example changes the parameter 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:

```
BigIron(config)# vlan 20
BigIron(config-vlan-20)# 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.

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 **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.

Possible values: See above

Default values: See above

Port Parameters:

These commands can be entered on individual ports or on multiple ports, such as all ports that belong to a VLAN.

EXAMPLE:

You can change the following 802.1W port parameters using the following methods.

```
BigIron(config)# vlan 10
BigIron(config-vlan-10)# spanning-tree 802-1w ethernet 1/5 path-cost 15 priority 64
```

Syntax: spanning-tree 802-1w ethernet <portnum> path-cost <value> | priority <value> | [admin-edge-port] | [admin-pt2pt-mac] | [force-migration-check]

The **ethernet | pos** <portnum> parameter specifies the interface used.

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 33.1 shows the recommended path cost values from the IEEE standards.

Table 33.1: 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
OC-3c	128,000	12,800 – 1,280,000
OC-12c	32,000	3,200 – 320,000
OC-48c	8,000	800 – 80,000
OC-192c	2,000	200 – 20,000

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 – 255. The default is 128. A higher numerical value means a lower priority. 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.

Possible Values: See above

Default values: See above

spanning-tree rstp

Enables 802.1W Draft 3 Rapid Spanning Tree on the VLAN.

To enable full 802.1W, see “spanning-tree 802-1w” on page 33-12.

NOTE: To enable 802.1W Draft 3 on a device that is running Single Spanning Tree, enter the **spanning-tree single rstp** command at the global CONFIG level of the CLI. See “spanning-tree single rstp” on page 6-129.

802.1W Draft 3 enhances STP by providing a fast failover mechanism for a root port that fails on a non-root bridge. 802.1W Draft 3 implementation provides a subset of the capabilities described in the 802.1W STP specification.

EXAMPLE:

To enable 802.1W Draft 3 in a port-based VLAN, enter commands such as the following:

```
BigIron(config)# vlan 10
BigIron(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 RSTP.

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:

```
BigIron(config-vlan-10)# no spanning-tree rstp
```

Possible values: N/A

Default value: Disabled

spanning-tree single 802-1w

Enables 802.1W on all ports of a single spanning tree.

EXAMPLE:

Enter a command such as the following:

```
BigIron(config-vlan-10)# spanning-tree single 802-1w
```

Syntax: [no] spanning-tree single 802-1w

Possible values: N/A

Default value: Disabled

static-mac-address

Allows you to define a static MAC addresses for a port on an Foundry device to ensure the device is not aged out. When defining the MAC address entry, you can also define the port’s priority and whether or not it is a router-type or host-type.

NOTE: Foundry recommends that you configure a static ARP entry to match the static MAC entry. In fact, the software automatically creates a static MAC entry when you create a static ARP entry. See “arp” on page 6-13.

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.

EXAMPLE:

To enter a static MAC address entry for port 5, that is also resident in port-based VLAN 4, enter the following:

```
BigIron(config)# vlan 4

BigIron(config-vlan-4)# static-mac-address 023.876.735 ethernet 5 high-priority
router-type
```

Here is the syntax for Chassis devices, the FastIron 4802, and the TurboIron/8:

Syntax: [no] static-mac-address <mac-addr> ethernet <portnum> [to <portnum> ethernet <portnum>] [normal-priority | high-priority] [host-type | router-type | fixed-host]

Here is the syntax for Stackable devices:

Syntax: static-mac-address <mac-addr> ethernet <portnum> [normal-priority | high-priority] [host-type | router-type]

Possible values: see above

Default values: priority 0 or normal-priority; host-type

super-span

Enables or disables the SuperSpan™ feature in the VLAN.

For information about this feature, see the “SuperSpan™“ section in the “Configuring Spanning Tree Protocol (STP) and IronSpan“ chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Use this command after you configure the SuperSpan boundary interfaces. (See “stp-boundary” on page 7-75.) You can enable SuperSpan globally or on an individual VLAN level. If you enable the feature globally, the feature is enabled on all VLANs. To enable or disable SuperSpan globally, see “super-span-global” on page 6-131.

NOTE: If you enable the feature globally, then create a new VLAN, the new VLAN inherits the global SuperSpan state. For example, if SuperSpan is globally enabled when you create a VLAN, SuperSpan also is enabled in the new VLAN.

EXAMPLE:

```
BigIron(config)# vlan 10
BigIron(config-vlan-10)# no super-span
```

Syntax: [no] super-span

In this example, the command disables SuperSpan in VLAN 10 but leaves the feature enabled in other VLANs.

Possible values: N/A

Default value: The global state of the SuperSpan feature

tagged

Once a port-based VLAN is created, port membership for that VLAN must be defined. To assign a port to a port-based VLAN, either the **tagged** or **untagged** command is used. When a port is tagged, it can be a member of multiple port-based VLANs.

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, that in turn, is resident in all VLANs whose members need access to the server.

EXAMPLE:

Suppose you want to make port 5 (module 5), a member of port-based VLAN 4, a tagged port. Enter the following:

```
BigIron(config)# vlan 4
BigIron(config-vlan-4)# tagged ethernet 5/5
```

Syntax: tagged ethernet | pos <portnum> [to <portnum> [ethernet <portnum>]]

Possible values: see above.

Default value: N/A

untagged

Once a port-based VLAN is created, port membership for that VLAN must be defined. To assign a port to a port-based VLAN, either the **tagged** or **untagged** command is used. When a port is 'untagged' it can be a member of only one VLAN.

EXAMPLE:

Suppose you want to assign all ports on a 16-port router except port 5 (module 3) as untagged to a VLAN. To assign ports 1 – 4 and 6 – 16 to VLAN 4, enter the following:

```
BigIron(config)# vlan 4
```

```
BigIron(config-vlan-4)# untagged ethernet 3/1 to 3/4 e 3/6 to 3/16
```

Syntax: [no] untagged ethernet | pos <portnum> [to <portnum> ethernet <portnum>]

Syntax: [no] untagged atm <slot/port.subinterface>

Possible values: see above.

Default value: N/A

uplink-switch

Configures a set of ports within a port-based VLAN as uplink ports for the VLAN. All broadcast and unknown-unicast traffic goes only to the uplink ports, not to the other ports in the VLAN.

EXAMPLE:

To configure a port-based VLAN containing uplink ports, enter commands such as the following:

```
BigIron(config)# vlan 10 by port
BigIron(config-vlan-10)# untag ethernet 1/1 to 1/24
BigIron(config-vlan-10)# untag ethernet 2/1 to 2/2
BigIron(config-vlan-10)# uplink-switch ethernet 2/1 to 2/2
```

Syntax: [no] uplink-switch ethernet <portnum> [to <portnum> | ethernet <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.

Possible values: see above.

Default value: N/A

vsrp vrid

Configures a

EXAMPLE:

```
BigIron(config)# vlan 200
BigIron(config-vlan-200)# tag ethernet 1/1 to 1/8
BigIron(config-vlan-200)# vsrp vrid 1
BigIron(config-vlan-200-vrid-1)# backup
BigIron(config-vlan-200-vrid-1)# activate
```

Syntax: [no] vsrp vrid <num>

The <num> parameter specifies the VRID and can be from 1 – 255.

For information about the other commands in this example, see “VSRP Commands” on page 35-1.

Possible values: see above.

Default value: N/A

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-vlan-4)# write memory
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-vlan-4)# write terminal
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 34
Metro Ring Commands

diagnostics

Enables Metro Ring Protocol (MRP) diagnostics for a ring on the master node.

EXAMPLE:

```
BigIron(config-vlan-2-mrp-1)# diagnostics
```

Syntax: [no] diagnostics

NOTE: This command is valid only on the master node.

Possible values: N/A

Default value: Disabled

enable

Enables an MRP ring.

EXAMPLE:

```
BigIron(config)# vlan 2
BigIron(config-vlan-2)# metro-ring 1
BigIron(config-vlan-2-mrp-1)# name CustomerA
BigIron(config-vlan-2-mrp-1)# master
BigIron(config-vlan-2-mrp-1)# ring-interface ethernet 1/1 ethernet 1/2
BigIron(config-vlan-2-mrp-1)# enable
```

Syntax: [no] enable

Possible values: N/A

Default value: Disabled

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged EXEC level, enter the following from any level of the CLI.

```
BigIron(config-vlan-2-mrp-1)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the port-based VLAN level if configuring a protocol VLAN. If configuring a port-based VLAN, activity would be moved to the global level.

EXAMPLE:

```
BigIron(config-vlan-2-mrp-1)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

hello-time

Changes the Ring Health Packet (RHP) hello time.

EXAMPLE:

```
BigIron(config-vlan-2-mrp-1)# hello-time 200
```

This command changes the hello time to 200 ms.

Syntax: [no] hello-time <ms>

Possible values: 100 – 1000 (one second)

Default value: 100 ms

master

Designates this device to be the master node for the MRP ring.

NOTE: The master node must be a NetIron or BigIron device. FastIron Layer 2 Switches can be member nodes but cannot be master nodes.

EXAMPLE:

```
BigIron(config)# vlan 2
BigIron(config-vlan-2)# metro-ring 1
BigIron(config-vlan-2-mrp-1)# master
```

Syntax: [no] master

NOTE: Enter this command only on one node in the ring.

Possible values: N/A

Default value: non-master node

name

Adds a name to an MRP ring.

EXAMPLE:

```
BigIron(config-vlan-2-mrp-1)# name CustomerA
```

Syntax: [no] name <string>

The <string> parameter specifies a name for the ring.

Possible values: The name 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”).

Default value: no name assigned

no

Disables other commands. To disable a command, place the word ***no*** before the command.

preforwarding-time

Changes the RHP preforwarding time.

EXAMPLE:

```
BigIron(config-vlan-2-mrp-1)# preforwarding-time 400
```

This command changes 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.

The <ms> specifies the number of milliseconds. 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.

Possible values: 200 – 5000 ms

Default value: 300 ms

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-vlan-2-mrp-1)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

ring-interface

Specifies the device interfaces attached to the ring. Each device has two interfaces to the ring, a primary interface and a secondary 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.

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.

EXAMPLE:

```
BigIron(config-vlan-2-mrp-1)# ring-interface ethernet 1/1 ethernet 1/2
```

Syntax: [no] ring-interface ethernet | pos <primary-if> ethernet | pos <secondary-if>

The **ethernet | pos** <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 | pos** <secondary-if> parameter specifies the secondary interface.

You can use two Ethernet interfaces, two POS interfaces, or a combination of Ethernet and POS.

Possible values: See above

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-vlan-2-mrp-1)# write memory
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-vlan-2-mrp-1)# write terminal
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 35
VSRP Commands

activate

Activates a VSRP VRID.

NOTE: This command is equivalent to the **enable** command.

EXAMPLE:

```
BigIron(config)# vlan 200
BigIron(config-vlan-200)# tag ethernet 1/1 to 1/8
BigIron(config-vlan-200)# vsrp vrid 1
BigIron(config-vlan-200-vrid-1)# backup
BigIron(config-vlan-200-vrid-1)# activate
```

Syntax: [no] activate

Possible values: N/A

Default value: Disabled

advertise backup

Enables a Backup to send Hello messages to the Master.

By default, Backups do not send Hello messages to advertise themselves to the Master. You can enable these messages if desired.

EXAMPLE:

```
BigIron(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 messages to the Master every 60 seconds by default. (This is the default Hello interval.)

Possible values: N/A

Default value: Disabled

backup

Configures the device as a VSRP Backup for a VSRP VRID.

Optional parameters enable you to change the backup priority used for VSRP master election and the track priority, which can affect the device's VSRP priority.

NOTE: The **backup** 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.

EXAMPLE:

```
BigIron(config)# vlan 200
BigIron(config-vlan-200)# tag ethernet 1/1 to 1/8
BigIron(config-vlan-200)# vsrp vrid 1
BigIron(config-vlan-200-vrid-1)# backup
BigIron(config-vlan-200-vrid-1)# activate
```

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# backup priority 75
```

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# backup track-priority 2
```

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.

The **track-priority** <value> parameter specifies the 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.

Possible values: See above

Default value: See above

backup-hello-interval

Changes the Backup Hello interval.

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# backup-hello-interval 180
```

Syntax: [no] backup-hello-interval <num>

The <num> parameter specifies the message interval.

Possible values: 60 – 3600 seconds

Default value: 60 seconds

dead-interval

Changes 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.

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# dead-interval 30
```

Syntax: [no] dead-interval <num>

The <num> parameter specifies the interval.

Possible values: 1 – 84 seconds

Default value: 3 seconds (three times the default Hello interval)

disable

Disables a VSRP VRID.

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# disable
```

Syntax: disable

Possible values: N/A

Default value: Disabled

enable

Enables a VSRP VRID.

NOTE: This command is equivalent to the **activate** command.

EXAMPLE:

```
BigIron(config)# vlan 200
BigIron(config-vlan-200)# tag ethernet 1/1 to 1/8
BigIron(config-vlan-200)# vsrp vrid 1
BigIron(config-vlan-200-vrid-1)# backup
BigIron(config-vlan-200-vrid-1)# enable
```

Syntax: enable

Possible values: N/A

Default value: Disabled

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged EXEC level, enter the following from any level of the CLI.

```
BigIron(config-vlan-200-vrid-1)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the port-based VLAN level if configuring a protocol VLAN. If configuring a port-based VLAN, activity would be moved to the global level.

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

hello-interval

Changes the Backup Hello interval.

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# backup-hello-interval 180
```

Syntax: [no] backup-hello-interval <num>

The <num> parameter specifies the message interval.

Possible values: 60 – 3600 seconds

Default value: 60 seconds

hold-down-interval

Changes 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.

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# hold-down-interval 4
```

Syntax: [no] hold-down-interval <num>

The <num> parameter specifies the hold-down interval.

Possible values: 1 – 84 seconds

Default value: 2 seconds

include-port

Removes a port from a VSRP VRID.

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.

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# no include-port ethernet 1/2
```

Syntax: [no] include-port ethernet | pos <portnum>

The **ethernet | pos** <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.

Possible values: See above

Default value: All ports in the VRID's VLAN are included

initial-ttl

Changes the TTL for a VSRP VRID.

A VSRP Hello packet's TTL specifies how many hops the packet can traverse before being dropped. A hop can be a Layer 3 router 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.

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# initial-ttl 5
```


Syntax: [no] initial-ttl <num>

The <num> parameter specifies the TTL.

Possible values: 1 – 255

Default value: 2

ip address

Specifies an IP address to back up.

NOTE: This command is equivalent to the **ip-address** command.

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 "Configuring VRRP and VRRPE" chapter in the *Foundry Enterprise Configuration and Management Guide*.

NOTE: The VRID IP address must be in the same sub-net 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.

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# ip address 10.10.10.1
```

Syntax: [no] ip address <ip-addr>

Possible values: An IP address

Default value: None

ip-address

Specifies an IP address to back up.

NOTE: This command is equivalent to the **ip address** command (above).

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# ip-address 10.10.10.1
```

Syntax: [no] ip-address <ip-addr>

Possible values: An IP address

Default value: None

no

Disables other commands. To disable a command, place the word ***no*** before the command.

non-preempt-mode

Disables preemption on a Backup.

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.

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# non-preempt-mode
```

Syntax: [no] non-preempt-mode

Possible values: N/A

Default value: Pre-emption is enabled

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

save-current-values

Configures a Backup to save the VSRP timer values received from the Master instead of the timer values configured on the Backup.

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.

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# save-current-values
```

Syntax: [no] save-current-values

Possible values: N/A

Default value: Disabled; each Backup saves the configured timer values to its startup-config file when you save the device's configuration

show

Displays a variety of configuration and statistical information about the device. See "Show Commands" on page 43-1.

track-port

Configures a VRID to track an interface.

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.

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# track-port e 2/4
```

Syntax: [no] track-port ethernet <portnum> | pos <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. See "backup" on page 35-1.

Possible values: See above

Default value: None configured

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# write memory
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-vlan-200-vrid-1)# write terminal
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 36
Topology Group Commands

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged EXEC level, enter the following from any level of the CLI.

```
BigIron(config-topo-group-2)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global CONFIG level.

EXAMPLE:

```
BigIron(config-topo-group-2)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

master-vlan

Adds the master VLAN to a topology group.

EXAMPLE:

```
BigIron(config)# topology-group 2
BigIron(config-topo-group-2)# master-vlan 2
```

Syntax: [no] master-vlan <vlan-id>

NOTE: 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.

Possible values: See above

Default value: N/A

member-group

Adds a VLAN group to a topology group.

EXAMPLE:

```
BigIron(config)# topology-group 2
BigIron(config-topo-group-2)# master-vlan 2
BigIron(config-topo-group-2)# member-vlan 3
BigIron(config-topo-group-2)# member-vlan 4
BigIron(config-topo-group-2)# member-vlan 5
BigIron(config-topo-group-2)# member-group 2
```

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.

Possible values: See above

Default value: N/A

member-vlan

Adds a member VLAN to a topology group.

EXAMPLE:

```
BigIron(config)# topology-group 2
BigIron(config-topo-group-2)# master-vlan 2
BigIron(config-topo-group-2)# member-vlan 3
BigIron(config-topo-group-2)# member-vlan 4
BigIron(config-topo-group-2)# member-vlan 5
BigIron(config-topo-group-2)# member-group 2
```

Syntax: [no] member-vlan <vlan-id>

The <vlan-id> parameter specifies a VLAN ID. The VLAN must already be configured.

Possible values: See above

Default value: N/A

no

Disables other commands. To disable a command, place the word ***no*** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-topo-group-2)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-topo-group-2)# write memory
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-topo-group-2)# write terminal
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 37
STP Group Commands

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged EXEC level, enter the following from any level of the CLI.

```
BigIron(config-stp-group-1)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global CONFIG level.

EXAMPLE:

```
BigIron(config-stp-group-1)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

master-vlan

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.

EXAMPLE:

```
BigIron(config)# vlan 2
BigIron(config-vlan-2)# spanning-tree priority 1
BigIron(config-vlan-2)# tagged ethernet 1/1 ethernet to 1/4
BigIron(config-vlan-2)# vlan 3
BigIron(config-vlan-3)# tagged ethernet 1/1 ethernet to 1/4
BigIron(config-vlan-3)# vlan 4
BigIron(config-vlan-4)# tagged ethernet 1/1 ethernet to 1/4
BigIron(config)# stp-group 1
BigIron(config-stp-group-1)# master-vlan 2
```

```
BigIron(config-stp-group-1)# member-vlan 3 to 4
```

These commands configure three port-based VLANs, then add VLAN 2 to STP group 1 as a master VLAN. The STP settings in this VLAN are used for all the VLANs in the STP group. In this case, the STP settings in VLAN 2 are used for VLANs 2, 3, and 4.

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] master-vlan <num>

Possible values: A valid VLAN ID

Default value: N/A

member-group

Adds VLANs to an STP group. The VLANs inherit the STP settings of the master VLAN in the group.

EXAMPLE:

```
BigIron(config)# vlan-group 1 vlan 5 to 1000
BigIron(config-vlan-group-1)# tagged 1/1 to 1/2
BigIron(config-vlan-group-1)# exit
BigIron(config)# stp-group 1
BigIron(config-stp-group-1)# master-vlan 2
BigIron(config-stp-group-1)# member-group 1
```

These commands add VLAN 2 as a master VLAN for the STP group, and add all the VLANs in VLAN group 1 as members of the STP group.

Syntax: [no] member-vlan <num> [to <num>]

Possible values: A valid VLAN group ID

Default value: N/A

member-vlan

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.

EXAMPLE:

```
BigIron(config)# stp-group 1
BigIron(config-stp-group-1)# master-vlan 2
BigIron(config-stp-group-1)# member-vlan 3 to 4
```

These commands add VLAN 2 as a master VLAN for the STP group, and add VLANs 3 and 4 as members of the STP group.

Syntax: [no] member-group <num>

Possible values: A valid VLAN ID

Default value: N/A

no

Disables other commands. To disable a command, place the word ***no*** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-stp-group-1)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-stp-group-1)# write memory
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-stp-group-1)# write terminal
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 38
GVRP Commands

block-applicant

Disables VLAN advertising on a port enabled for GVRP.

EXAMPLE:

```
BigIron(config-gvrp)# block-applicant ethernet 1/24 ethernet 6/24 ethernet 8/17
```

This command disables advertising of VLAN information on ports 1/24, 6/24, and 8/17.

Syntax: [no] block-applicant all | ethernet <portnum> [ethernet <portnum> | to <portnum>]

NOTE: Leaveall messages are still sent on the GVRP ports.

Possible values: See above

Default value: Advertising is enabled

block-learning

Disables VLAN learning on a port enabled for GVRP.

EXAMPLE:

```
BigIron(config-gvrp)# block-learning ethernet 6/24
```

This command disables learning of VLAN information on port 6/24.

NOTE: The port still advertises VLAN information unless you also disable VLAN advertising.

Syntax: [no] block-learning all | ethernet <portnum> [ethernet <portnum> | to <portnum>]

Possible values: See above

Default value: Learning is enabled

default-timers

Resets the GVRP Join, Leave, and Leaveall timers to their default values.

EXAMPLE:

```
BigIron(config-gvrp)# default-timers
```

Syntax: default-timers

Possible values: N/A

Default values:

- Join – 200 ms
- Leave – 600 ms
- Leaveall – 10000 ms

enable

Enables GVRP on specific interfaces.

EXAMPLE:

```
BigIron(config)# gvrp-enable
BigIron(config-gvrp)# enable all
```

The first command globally enables support for the feature and changes the CLI to the GVRP configuration level. The second command enables GVRP on all ports on the device.

The following command enables GVRP on ports 1/24, 6/24, and 8/17:

```
BigIron(config-gvrp)# enable ethernet 1/24 ethernet 6/24 ethernet 8/17
```

Syntax: [no] gvrp-enable

Syntax: [no] enable all | ethernet <portnum> [ethernet <portnum> | to <portnum>]

The **all** parameter enables GVRP on all ports.

The **ethernet** <portnum> [**ethernet** <portnum> | **to** <portnum>] parameter enables GVRP on the specified list or range of Ethernet ports.

- To specify a list, enter each port as **ethernet** <portnum> followed by a space. For example, to enable GVRP on three Ethernet ports, enter the following command: **enable ethernet 1/24 ethernet 6/24 ethernet 8/17**
- To specify a range, enter the first port in the range as **ethernet** <portnum> followed by **to** followed by the last port in the range. For example, to add ports 1/1 – 1/8, enter the following command: **enable ethernet 1/1 to 1/8**

You can combine lists and ranges in the same command. For example: **enable ethernet 1/1 to 1/8 ethernet 1/24 ethernet 6/24 ethernet 8/17**

Possible values: See above

Default value: Disabled

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged EXEC level, enter the following from any level of the CLI.

```
BigIron(config-gvrp)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global CONFIG level.

EXAMPLE:

```
BigIron(config-gvrp)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

join-timer

Changes the GVRP Join, Leave, and Leaveall timers.

EXAMPLE:

```
BigIron(config-gvrp)# join-timer 1000 leave-timer 3000 leaveall-timer 15000
```

This command changes the Join timer to 1000 ms, the Leave timer to 3000 ms, and the Leaveall timer to 15000.

Syntax: [no] join-timer <ms> leave-timer <ms> leaveall-timer <ms>

NOTE: When you enter this command, all the running GVRP timers are canceled and restarted using the new times specified by the command.

Possible values:

- Join timer – from 200 ms to one third the value of the Leave timer
- Leave timer – from three times the Join timer to one fifth the value of the Leaveall timer
- Leaveall timer – from five times the Leave timer to maximum value allowed by software (configurable from 300000 – 1000000 ms)

  NOTE: To change the maximum value for the Leaveall timer, see “gvrp-max-leaveall-timer” on page 6-31.

Timer Configuration Requirements

- All timer values must be in multiples of 100 ms.
- The Leave timer must be >= 3* the Join timer.
- The Leaveall timer must be >= 5* the Leave timer.
- The GVRP timers must be set to the same values on all the devices that are exchanging information using GVRP.

Default value: Join – 200 ms, Leave – 600 ms, Leaveall – 10000 ms

NOTE: To reset the timers to their default values, use the default-timers command. See “default-timers” on page 38-1.

no

Disables other commands. To disable a command, place the word ***no*** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-gvrp)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-gvrp)# write memory
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-gvrp)# write terminal
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 39
802.1X Port Security Commands

enable

Enables 802.1X port security on the device. You can enable 802.1X port security on all interfaces at once, on individual interfaces, or on a range of interfaces.

EXAMPLE:

For example, to enable 802.1X port security on all interfaces on the device:

```
BigIron(config-dot1x)# enable all
```

To enable 802.1X port security on interface 3/11:

```
BigIron(config-dot1x)# enable ethernet 3/11
```

To enable 802.1X port security on interfaces 3/11 through 3/16, enter the following command:

```
BigIron(config-dot1x)# enable ethernet 3/11 to 3/16
```

Syntax: [no] enable all

Syntax: [no] enable <portnum>

Syntax: [no] enable <portnum> to <portnum>

Possible values: Interface number

Default value: By default, 802.1X port security is disabled.

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
BigIron(config-dot1x)#
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
BigIron(config-dot1x)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

maxreq

Specifies the maximum number of EAP-request/identity frame retransmissions the device sends to an 802.1X-enabled Client. If no EAP-response/identity frame is received from the Client after the specified number of EAP-request/identity frame retransmissions, the device restarts the authentication process with the Client.

EXAMPLE:

To configure the device to retransmit an EAP-request/identity frame to a Client a maximum of three times, enter the following command:

```
BigIron(config-dot1x)# maxreq 3
```

Syntax: maxreq <value>

Possible values: 1 – 10 retransmissions

Default value: 2 retransmissions

no

Disables other commands. To disable a command, place the word **no** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-dot1x)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

re-authentication

Configures the device to periodically re-authenticate Clients connected to 802.1X-enabled interfaces. When you enable periodic re-authentication, the device re-authenticates Clients every 3,600 seconds by default. You can optionally specify a different re-authentication interval with the **timeout re-authperiod** command.

EXAMPLE:

```
BigIron(config-dot1x)# re-authentication
```

Syntax: [no] re-authentication

Possible values: N/A

Default value: N/A

servertimeout

Specifies the amount of time the device waits for a response from a RADIUS server before retransmitting the message to the RADIUS server.

EXAMPLE:

```
BigIron(config-dot1x)# servertimeout 45
```

Syntax: servertimeout <seconds>

Possible values: 0 – 4294967295 seconds

Default value: 30 seconds

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

supptimeout

Specifies the amount of time the device waits for a response from an 802.1X-enabled Client before retransmitting the EAP-Request frame to the Client.

EXAMPLE:

```
BigIron(config-dot1x)# supptimeout 45
```

Syntax: supptimeout <seconds>

Possible values: 0 – 4294967295 seconds

Default value: 30 seconds

timeout quiet-period

If the Foundry device is unable to authenticate an 802.1X-enabled Client, specifies how long the device waits before trying again.

EXAMPLE:

```
BigIron(config-dot1x)# timeout quiet-period 30
```

Syntax: [no] timeout quiet-period <seconds>

Possible values: 0 – 4294967295 seconds

Default value: 60 seconds

timeout re-authperiod

When periodic re-authentication is enabled, specifies how often the device re-authenticates Clients. The re-authentication interval is a global setting, applicable to all 802.1X-enabled interfaces. If you want to re-authenticate Clients connected to a specific port manually, use the **dot1x re-authenticate** command.

EXAMPLE:

```
BigIron(config-dot1x)# timeout re-authperiod 2000
```

Syntax: [no] timeout re-authperiod <seconds>

Possible values: 0 – 4294967295 seconds

Default value: 3,600 seconds

timeout security-hold-time

Specifies the amount of time an interface is placed in the unauthorized state if multiple Clients try to connect on the interface (provided that the **multiple-hosts** command is not used).

NOTE: When the **port-control** parameter on an 802.1X-enabled interface is set to **force-authorized**, the Foundry device allows connections from multiple Clients, regardless of whether the **multiple-hosts** parameter is used in the interface's configuration.

EXAMPLE:

```
BigIron(config-dot1x)# timeout security-hold-time 120
```

Syntax: [no] timeout security-hold-time <seconds>

Possible values: 0 – 4294967295 seconds

Default value: 60 seconds

timeout tx-period

Specifies how long the device waits for a Client to send back an EAP-response/identity frame after the device has sent an EAP-request/identity frame. If the Client does not send back an EAP-response/identity frame during this time, the device retransmits the EAP-request/identity frame.

EXAMPLE:

```
BigIron(config-dot1x)# timeout tx-period 60
```

Syntax: [no] timeout tx-period <seconds>

Possible values: 0 – 4294967295 seconds

Default value: 30 seconds

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-port-security)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-port-security)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 40
MAC Port Security Commands

age

Configures the device to age out secure MAC addresses after a specified amount of time.

EXAMPLE:

To set the port security age timer to 10 minutes on all interfaces:

```
BigIron(config)# port security
BigIron(config-port-security)# age 10
```

To set the port security age timer to 10 minutes on a specific interface:

```
BigIron(config)# int e 7/11
BigIron(config-if-e100-7/11)# port security
BigIron(config-port-security-e100-7/11)# age 10
```

Syntax: [no] age <minutes>

Possible values: 0 – 1440 minutes

Default value: 0 minutes (never age out secure MAC addresses)

autosave

Configures the device to automatically save learned MAC addresses to the startup-config file at specified intervals.

EXAMPLE:

To automatically save learned secure MAC addresses on the device every twenty minutes:

```
BigIron(config-port-security)# autosave 20
```

Syntax: [no] autosave <minutes>

Possible values: 15 – 1440 minutes

Default value: By default, secure MAC addresses are not autosaved to the startup-config file.

enable

Enables or disables the port security feature, either globally or on specified interfaces.

EXAMPLE:

To enable the feature on all interfaces at once:

```
BigIron(config)# port security
BigIron(config-port-security)# enable
```

To disable the feature on all interfaces at once:

```
BigIron(config)# port security
BigIron(config-port-security)# no enable
```

To enable the feature on a specific interface:

```
BigIron(config)# int e 7/11
BigIron(config-if-e100-7/11)# port security
BigIron(config-port-security-e100-7/11)# enable
```

Syntax: [no] enable

Possible values: N/A

Default value: By default, the port security feature is disabled on all interfaces.

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged level, enter the following from any level of the CLI.

```
BigIron(config-port-security)#
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level. In this case, activity will be moved to the global level.

EXAMPLE:

```
BigIron(config-port-security)# exit
NetIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

maximum

Adjusts the maximum number of MAC addresses that can be secured.

EXAMPLE:

To configure interface 7/11 to have a maximum of 10 secure MAC addresses:

```
BigIron(config)# int e 7/11
BigIron(config-if-e100-7/11)# port security
BigIron(config-if-e100-7/11)# maximum 10
```

Syntax: maximum <number-of-addresses>

Possible values: The <number-of-addresses> parameter can be set to a number from 0 – (64 + the total number of global resources available) The total number of global resources is 2048 or 4096, depending on flash memory size. Setting the parameter to 0 prevents any addresses from being learned.

Default value: 1 MAC address

no

Disables other commands. To disable a command, place the word **no** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-port-security)# quit
NetIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

secure

Specifies a secure MAC address on an interface.

EXAMPLE:

```
BigIron(config)# int e 7/11
BigIron(config-if-e100-7/11)# port security
BigIron(config-port-security-e100-7/11)# secure 0050.DA18.747C
```

Syntax: [no] secure <mac-address>

Possible values: MAC address

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

violation restrict

Configures the device to drop packets from a violating MAC address when a security violation occurs, and allow packets from secure addresses. A security violation can occur when a user tries to plug into a port where a MAC address is already locked, or the maximum number of secure MAC addresses has been exceeded.

EXAMPLE:

```
BigIron(config)# int e 7/11
BigIron(config-if-e100-7/11)# port security
BigIron(config-port-security-e100-7/11)# violation restrict
```

Syntax: violation restrict

Possible values: N/A

Default value: N/A

violation shutdown

Shuts down the port for a specified amount of time when a security violation occurs.

EXAMPLE:

```
BigIron(config)# int e 7/11
BigIron(config-if-e100-7/11)# port security
BigIron(config-port-security-e100-7/11)# violation shutdown 5
```

Syntax: violation shutdown <minutes>

Possible values: You can specify from 0 – 1440 minutes. Specifying 0 shuts down the port permanently when a security violation occurs.

NOTE: When using this feature with a 24-port 10/100 module (part number B24E) only the **violation shutdown** command is supported. The **violation restrict** command is not supported on the B24E.

Default value: N/A

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-port-security)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration of the Foundry device on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-port-security)# wr term
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 41
Real Server Commands

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged EXEC level, enter the following from any level of the CLI.

```
BigIron(config-rs-S2)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level.

EXAMPLE:

```
BigIron(config-rs-S2)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

no

Disables a command. To disable a command, place the word ***no*** before the command.

port <num> disable

Disables the application health check for the port. By default, when you create a profile for the port (using the server port <num> command), the health check is enabled.

EXAMPLE:

```
BigIron(config-rs-S2)# port http disable
```

Syntax: port http | <num>

The **http** parameter is for port 80. If your server uses a different port number for HTTP, enter the port number.

Possible values: N/A

Default value: N/A

port <num> keepalive

Enables the HTTP health check for a web site you configured a Layer 3 Switch to assist with Geographically-distributed Server Load Balancing (SLB). Globally-distributed SLB allows the same web site (and same IP address) to reside on multiple servers, which usually are in geographically dispersed locations. To specify the web sites that the Layer 3 Switch is assisting, use the **server real...** command. See “server real-name” on page 6-112.

The health check is disabled by default.

NOTE: By default, the HTTP health check sends a HEAD request to the web server for its default web page (“1.0”). If the server responds with a status code from 200 – 299, the server passes the health check. You can customize the URL requested by the health check and also the status codes that the Layer 3 Switch determines satisfactory replies to a health check. See “port <num> url” on page 41-3 and “port <num> status_code” on page 41-2.

EXAMPLE:

To configure a Layer 3 Switch to assist a web site at IP address 209.157.22.249 and enable the HTTP health check for the web site, enter the following commands:

```
BigIron(config)# server real S2 209.157.22.249

BigIron(config-rs-S2)# port http
```

Syntax: [no] port <num> | http [keepalive <interval> <retries>]

The **http** parameter is for port 80. If your server uses a different port number for HTTP, enter the port number.

The <interval> parameter specifies the number of seconds between health checks sent by the Layer 3 Switch. You can specify a number from 2 – 60 seconds. The default is 5 seconds.

The <retries> parameter specifies how many times the Layer 3 Switch will resend a health check if the web site does not respond. You can specify from 1 – 5 retries. The default is 2.

Possible values: N/A

Default value: enabled

port <num> status_code

Changes the range of status codes that the Layer 3 Switch considers acceptable as replies to an HTTP health check. This command applies only when you are configuring a Layer 3 Switch to assist Foundry ServerIrons, third-party SLBs, or directly-connected web servers with globally-distributed Server Load Balancing. See the "Route Health Injection" chapter of the *Foundry Enterprise Configuration and Management Guide*.

EXAMPLE:

To add a web server, enable the HTTP health check for the server, and change the HTTP status codes that the Layer 3 Switch considers successful replies to the health check, enter the following commands.

```
BigIron(config)# server real S2 209.157.22.249

BigIron(config-rs-S2)# port http keepalive

BigIron(config-rs-S2)# port http status_code 200 299
```

Syntax: [no] port http status_code <range> [<range>[<range>[<range>]]]

The **http** parameter is for port 80. If your server uses a different port number for HTTP, enter the port number.

The default status code range for HTTP health checks is 200 – 299. You can specify up to four discrete ranges. To specify a single message code for a range, enter the code twice. For example, to specify 200 only, enter the following command: **port http status_code 200 200**.

Possible values: 100 – 505

Default value: 200 – 299

port <num> url

Changes the URL or request method for HTTP health checks. This command applies only when you are configuring a Layer 3 Switch to assist Foundry ServerIrons, third-party SLBs, or directly-connected web servers with Globally-distributed Server Load Balancing. See the "Route Health Injection" chapter of the *Foundry Enterprise Configuration and Management Guide*.

By default, the HTTP health check sends a HEAD request to the web server for its default web page ("1.0").

EXAMPLE:

To add a web server, enable the HTTP health check for the server, and change the URL requested by the health check to “Scully-Files”, enter the following commands.

```
BigIron(config) server real S2 209.157.22.249
BigIron(config-rs-S2)# port http keepalive
BigIron(config-rs-S2)# port http url "/Scully-Files.html"
```

EXAMPLE:

To change the request from a HEAD to a GET, enter the following command.

```
BigIron(config-rs-S2)# port http url GET "/Scully-Files.html"
```

Syntax: [no] port http url “[GET | HEAD] [/]<URL-page-name>”

The **http** parameter is for port 80. If your server uses a different port number for HTTP, enter the port number.

Possible values: 100 – 505

Default value: 200 – 299

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-rs-S2)# quit
BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-rs-S2)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-rs-S2)# wr t
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 42
Application Port Commands

NOTE: An ***application port*** is a Layer 4 TCP or UDP port. For example, port 80 is the well-known port number for the HTTP application used by Web browsers. The commands in this chapter apply to the route health injection (Global IP) feature described in the "Configuring Route Health Injection" chapter of the *Foundry Enterprise Configuration and Management Guide*.

end

Moves activity to the privileged EXEC level from any level of the CLI except the user EXEC level.

EXAMPLE:

To move to the privileged EXEC level, enter the following from any level of the CLI.

```
BigIron(config-port-80)# end
BigIron#
```

Syntax: end

Possible values: N/A

Default value: N/A

exit

Moves activity up one level from the current level.

EXAMPLE:

```
BigIron(config-port-80)# exit
BigIron(config)#
```

Syntax: exit

Possible values: N/A

Default value: N/A

no

Disable a commands. To do so, place the word ***no*** before the command.

quit

Returns you from any level of the CLI to the User EXEC mode.

EXAMPLE:

```
BigIron(config-port-80)# quit

BigIron>
```

Syntax: quit

Possible values: N/A

Default value: N/A

show

Displays a variety of configuration and statistical information about the device. See “Show Commands” on page 43-1.

tcp keepalive

Modifies health check parameters for an application port.

EXAMPLE:

```
BigIron(config-port-80)# tcp keepalive 10 3
```

Syntax: tcp keepalive <interval> <retries> | disable | enable

The <interval> parameter specifies the number of seconds between health checks sent by the Layer 3 Switch. You can specify a number from 2 – 60 seconds. The default is 5 seconds.

The <retries> parameter specifies how many times the Layer 3 Switch will resend a health check if the web site does not respond. You can specify from 1 – 5 retries. The default is 2.

The **disable** parameter disables the health check.

The **enable** parameter re-enables the health check. When you add the port (using the server port <num> command), the health check is automatically enabled for the port.

Possible values: see above

Default value: see above

write memory

Saves the running configuration into the startup-config file.

EXAMPLE:

```
BigIron(config-port-80)# wr mem
```

Syntax: write memory

Possible values: N/A

Default value: N/A

write terminal

Displays the running configuration on the terminal screen.

NOTE: This command is equivalent to the **show running-config** command.

EXAMPLE:

```
BigIron(config-port-80)# wr t
```

Syntax: write terminal

Possible values: N/A

Default value: N/A

Chapter 43
Show Commands

show 802-1w

Displays a summary of 802.1W information.

EXAMPLE:

```
BigIron(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 43.1.

Table 43.1: 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.
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 43.1: 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 This 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
State	The port's current 802.1W state. A port can have one of the following states: • Forwarding • Discarding • Learning • Disabled
Designated Cost	The best root path cost that this port received, including the best root path cost that it can transmit.

Table 43.1: CLI Display of 802.1W Summary (Continued)

This Field...	Displays...
Designated Bridge	The ID of the bridge that sent the best RST BPDU that was received on this port.

Possible values: N/A

Default values: N/A

show 802-1w detail

Display detailed information about 802-1W.

EXAMPLE:

```
BigIron(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.
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.

This Field...	Displays...
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
State	The port's current 802.1W state. A port can have one of the following states: • Forwarding • Discarding • Learning • Disabled
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.
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.

This Field...	Displays...
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 43-2.) • 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 3.) • 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.
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).

show aaa

Displays information about all TACACS+ and RADIUS servers identified on the device.

EXAMPLE:

```
BigIron# show aaa
Tacacs+ key: foundry
Tacacs+ retries: 1
Tacacs+ timeout: 15 seconds
Tacacs+ dead-time: 3 minutes
Tacacs+ Server: 207.95.6.90 Port:49:
                opens=6 closes=3 timeouts=3 errors=0
                packets in=4 packets out=4
no connection

Radius key: networks
Radius retries: 3
Radius timeout: 3 seconds
Radius dead-time: 3 minutes
Radius Server:  207.95.6.90 Auth Port=1645 Acct Port=1646:
                opens=2 closes=1 timeouts=1 errors=0
                packets in=1 packets out=4
no connection
```

Syntax: show aaa

Possible values: N/A

Default value: N/A

show access-list

Displays information for ACLs.

EXAMPLE:

Here is an example of the output for hardware-based ACLs on a JetCore device running software release 07.6.01 or later.

```
BigIron(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)
```

The following is an example of the output when a comment text has been configured for an ACL.

```
BigIron# 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

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.

Possible values: see above

Default value: N/A

show acl-on-arp

Shows the ACLs that have been configured to filter ARP requests.

EXAMPLE:

```
BigIron(config)# show acl-on-arp
Port  ACL ID  Filter Count
2     103     10
3     102     23
4     101     12
```

Syntax: show acl-on-arp [ethernet [<portnum>] | loopback [<num>] | ve [<num>]]

If port number or the interface number is not specified, all ports on the device that use ACLs for ARP filtering will be included in the display.

The Filter Count column shows how many ARP packets have been dropped on the interface since the last time the count was cleared.

Possible values: see above

Default value: N/A

show appletalk arp

Displays the ARP Table for the AppleTalk routing protocol.

EXAMPLE:

```
Index     Node Address     Mac Address      Port
1         10.30            00e0.5200.0000   1
```

Syntax: show appletalk arp

Possible values: N/A

Default value: N/A

show appletalk cache

Displays the forwarding table for the AppleTalk routing protocol. You can clear this cache by entering the CLI command, **clear appletalk cache**.

EXAMPLE:

```
BigIron> show appletalk cache
Total number of cache entries: 8
D:Dynamic  P:Permanent  F:Forward  U:Us  W:Wait ARP  K:Drop
     Destination       Next Hop      MAC              Type  Fid Vlan
1    6499.193          6300.22       0000.c541.bc71   DF    9    1
2    6401.0            6300.22       0000.c541.bc71   DF    9    1
3    6300.177          0.0           0000.0000.0000   PU    0
4    6300.22           0.0           0000.c541.bc71   DF    9    1
5    450.0             0.0           0000.0000.0000   PU    0
6    400.0             0.0           0000.0000.0000   PU    0
7    6300.0            0.0           0000.0000.0000   PU    0
8    450.177           0.0           0000.0000.0000   PU    0
```

Syntax: show appletalk cache

Possible values: N/A

Default value: N/A

show appletalk globals

Displays the global configuration parameters for the AppleTalk routing protocol.

EXAMPLE:

```
BigIron> show appletalk globals
AppleTalk Routing Global Settings:
enabled: Routing
disabled: Glean Packets
rtmp-update-interval: 10
zip-query-interval:10, arp-retransmit-interval: 1, arp-retransmit-count: 2
QOS Priority 0 Sockets: 1 - 254
QOS Priority 1 Sockets: None
```

Syntax: show appletalk globals

show appletalk interface

Displays the AppleTalk configuration for an individual interface or all interfaces.

EXAMPLE:

To view the configuration for all interfaces, enter **show appletalk interface**, as shown in the example below. To view the configuration of a specific interface, enter **show appletalk interface ethernet** <portnum>. To view the configuration of a virtual interface (VE), enter **show appletalk interface ve** <num>.

```
BigIron> show appletalk interface
Interface Ethernet 15
  port state: UP
  routing: Enabled
  operation mode:Seed Router
  address: 100.50,  cable-range: 100 - 100 arp-age 10
  Zone Filter List:
  Action: Permit  Zone name: sales, no RTMP Filtering
  Additional Zones Action: Permit, No RTMP Filtering
Interface Ethernet 16
  port state: DOWN
routing: Disabled
  operation mode:Routing not enabled.
  address: 200.50,  cable-range: 200 - 400 arp-age 10
  Zone Filter List:  No zone filters are configured.
Interface Ve 3
  members: ethe 1 to 3
  active: ethe 1
  port state: UP
  routing: Enabled
  operation mode: Seed Router
  address: 200.50,  cable-range: 200 - 200 arp-age 10
  Zone List:
  Finance
  Zone Filter List:  No zone filters are configured.
```

Syntax: show appletalk interface [ethernet <portnum> | ve <num>]

The **ethernet** <portnum> parameter lets you specify specific interface.

The **ve** <num> parameter lets you specify a virtual interface (VE).

Possible values: N/A

Default value: N/A

show appletalk route

Displays the AppleTalk routing table.

You can clear learned routes stored in the routing table by entering the **clear appletalk route** command.

EXAMPLE:

```
BigIron> show appletalk route
Index     Cable Range     Next Hop     Distance     State     Port
1         6300  - 6400    0.0          0            0         2
2         6401  - 6500    6300.22      1            0         2
3         400   - 499     0.0          0            0         1
4         500   - 599     450.10       1            0         1
5         600   - 699     450.10       2            0         1
6         200   - 300     450.10       2            0         1
7         1000  - 1100    450.10       2            0         1
8         1200  - 1299    450.10       2            0         1
9         7000  - 8000    450.10       1            0         1
```

NOTE: Please note the following regarding the information displayed in the AppleTalk routing table:

Index: Identifies the entry.

Cable Range: Shows the network numbers to which the route information applies.

Next Hop: Shows the address of the next hop router to which packets for that destination will be sent.

Distance: Indicates the number of hops away that the destination is from this router

State: Indicates the state of the entry. The possible states that may be displayed in this field are listed below with the numerical value that will appear in the table:

- Good route: 0
- Suspect route: 2
- Bad Route: 4

Port: References the port number upon which the next hop router is found.

Syntax: show appletalk route

Possible values: N/A

Default value: N/A

show appletalk traffic

Displays statistical information for RTMP, ZIP, AEP, DDP and AARP packets.

EXAMPLE:

```
BigIron> show appletalk traffic
RTMP Statistics:
  Received: 16038, Transmitted: 16032, Filtered: 0
ZIP Statistics:
  Query Received:16, Transmitted:6, GZL Received: 2, Transmitted: 1
NetInfo Statistics:
  Received: 10   , Reply:8
AEP Statistics:
  Request Received: 0, Request Transmitted: 0
  Reply Received: 0,   Reply Transmitted: 0
DDP Statistics:
  Received: 55468,  Transmitted: 55445,  Forwarded: 39372
  In-Delivered: 16092, Dropped-No-Route:0, Dropped-Bad-Hop-Counts: 0
  Dropped-Other-Reasons: 0
AARP Statistics:
Received: 14, Transmitted: 22
```

NOTE: Note the following regarding the information displayed in the AppleTalk traffic table.

RTMP Statistics: Provides a count of all RTMP packets received, transmitted and filtered on the router.

ZIP statistics: Provides a count of requests for zone information (Recv. Query) the system receives as well as a count of those ZIP queries made to other routers (Query, Transmitted). The 'Recv GZL' count lists those Get Zone List requests received from other routers and the 'Transmitted' field lists those GZL requests transmitted to other routers.

NetInfo Statistics: The received and reply values of this field refer to the number of zone and network number requests made and received by the router.

AEP Statistics: Provides a count of those AppleTalk Echo Protocol (pings) requests received or transmitted and a count of the replies received or transmitted.

DDP Statistics: Displays the total count of those DDP packets transmitted, received and forwarded from the router; those packets received and forwarded up the AppleTalk protocol stack (in-delivered) and those packets dropped due to an unknown route (no-route), those packets that exceeded maximum hop count and those that were dropped due to unknown MAC address (other-reasons).

AARP Statistics: Displays the total count of those AARP packets received and transmitted by the router.

Syntax: show appletalk traffic

Possible values: N/A

Default value: N/A

show appletalk zone

Displays the network numbers and zones learned on the network. You can clear all information stored in the zone table by entering the **clear appletalk route** command.

EXAMPLE:

```
BigIron> show appletalk zone
Index       Cable   Range      Zonename
1           6300   - 6400      QA
2           6300   - 6400      QARouter
3           6401   - 6500      QA1
4           6401   - 6500      QALab2
5           400    - 499       account
6           1200   - 1299      sales
7           1000   - 1100      engineering
```



```
8           1000   - 1100      foundrynetwork
9           1000   - 1100      foundrynetworks1
10          200    - 300       marketing
11          600    - 699       management
12          500    - 599       gigabit
13          7000   - 8000      gatethernet0
```

Syntax: show appletalk zone

Possible values: N/A

Default value: N/A

show aps

Displays information about a POS APS configuration.

EXAMPLE:

```
BigIron# show aps
POS2/1 working group 1 channel 1 Enabled Selected
```

Syntax: show aps

In this example, the output indicates that POS interface 2/1 is the working interface for channel 1 in APS group 1, and the interface is active. If there is a tilde next to Selected (for example, `~Selected`) it means the interface is not active.

Possible values: N/A

Default value: N/A

show arp

Displays the ARP cache of the device. See the "Configuring IP" chapter of the *Foundry Enterprise Configuration and Management Guide* for information about the fields in this display.

NOTE: For Layer 3 Switches, this command displays dynamic entries and static entries. If you want to display only the static entries on a Layer 3 Switch, see “show ip static-arp” on page 43-78.

NOTE: For Layer 2 Switches, this command does not display the 'Type' column, but does display a VLAN ID column. Layer 2 Switches do not have static entries, so the Type column is not needed.

EXAMPLE:

```
BigIron# 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 <portnum> | mac-address <xxxx.xxxx.xxxx> [<mask>] | <ip-addr> [<ip-mask>]] [<num>]

Specify the MAC address mask as “f”s and “0”s, where “f”s are significant bits. Specify IP address masks in standard decimal mask format (for example, 255.255.0.0).

The **ethernet** <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.

Here are some examples of how to use these commands.

The following command displays all ARP entries for MAC addresses that begin with “abcd”:

```
BigIron# show arp mac-address a.b.c.d ffff.0000.0000
```

The following command displays all IP address entries for IP addresses that begin with "209.157":

```
BigIron# show arp 209.157.0.0 255.255.0.0
```

Possible values: See above.

Default value: N/A

show atm map

Displays the point-to-multipoint mappings for an ATM point-to-multipoint interface.

EXAMPLE:

```
BigIron(config)# show atm map
      IP Address          VCI   -  VPI         Type        Age       Port
1     10.1.1.1            0     -  202         Static      None      4/2
2     10.1.1.2            0     -  204         Static      None      4/2
3     10.1.1.3            0     -  205         Static      None      4/2
```

Syntax: show atm map

For information about the fields in this display, see the "Using Asynchronous Transfer Mode Modules" chapter in the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Possible values: N/A

Default value: N/A

show atm-state

Displays general ATM module information.

EXAMPLE:

```
BigIron> show atm-state
================================================
ATM MODULE (6) App CPU in running mode:
        CPU 0 in state of ATM_STATE_RUNNING
---------------
Module 6 App CPU 1, SW: Version 07.2.05T91
Compiled on Apr 25 2001 at 18:16:41 labeled as A2R07205
DRAM 134M, BRAM 33554K, FPGA Version 0000
Code Flash 4M: Primary (429266 bytes, 07.2.05T91),
               Secondary (426573 bytes, 07.2.05T91)
Boot Flash 131K, Boot Version 06.00.00
The system uptime is 0 day 0 hour 31 minute 6 second
General Status: 4 ipc msg rec, 2 ipc msg sent
```

Syntax: show atm-state

Possible values: N/A

Default value: N/A

show atm vc

Displays ATM VC statistics.

EXAMPLE:

```
BigIron(config-subif-4/1.1)# show atm vc

Port   VPI VCI  InPkts      OutPkts      InBytes    OutBytes

6/1.1   1    100 69626        1129         6330289      68211
6/1.2   1    101 214784       307          19170286     28584
6/1.3   2    200 4910         353          439675       34612
6/1.4   2    201 296          221596       28529        19768722
6/1.5   2    202 220          269          23252        29349
6/2.1   1    13  1040         515          389135       45016
```

Syntax: show atm vc [slot <slotnum> | interface <portnum> | <slotnum>/<portnum>.sub-interface <subif>]

You can use the command's optional parameters to refine the display request.

- The **slot** <slotnum> parameter displays VC statistics for the ATM interfaces on the ATM module in the specified chassis slot.
- The **interface** <portnum> parameter displays VC statistics for the specified ATM interface.
- The **sub-interface** <subif> parameter displays VC statistics for the specified sub-interface number.

Fro information about the fields in this display, see the "Using Asynchronous Transfer Mode Modules" chapter in the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Possible values: See above

Default value: N/A

show cam

Displays the Layer 2, Layer 3, and Layer 4 entries in a Chassis device's Content Addressable Memory (CAM).

EXAMPLE:

To display Layer 2 entries in the CAM, enter a command such as the following at any level of the CLI:

```
BigIron> show cam ethernet 3/1
 Slot    Index                MAC       Age  Source Port    VLAN    Out Port
  3      1024       00a0.23d2.04ed        2     ethe 3/16      3       ethe 3/14
  3      1025       00a0.23d7.05ed        1     ethe 3/1       1       ethe 3/3
  3      1026       00a0.23d2.08ed        4     ethe 3/2       1       ethe 3/8
  3      1027       00a0.25d2.08ed        2     ethe 3/5       1       ethe 3/9
```

Syntax: show cam ethernet <portnum> [mac-address <mac-addr>]

To show information for a specific MAC address, enter the address as in the following example:

```
BigIron> show cam ethernet 3/1 00a0.23d2.04ed
 Slot    Index                MAC       Age  Source Port     VLAN    Out Port
  3       1024      00a0.23d2.04ed        2     ethe  3/16      3       ethe 3/14
```

This command shows the following information.

Table 43.2: CLI Display of Layer 2 CAM Information

This Field...	Displays...
Slot	The chassis slot number.
Index	The entry number for this CAM entry.
MAC	The MAC address of the entry.
Age	The age of the entry.
Source Port	The port on which the MAC address was learned.
VLAN	The port-based VLAN that contains the entry's MAC address.
Out Port	The port on which traffic from the MAC address gets forwarded.

To display Layer 3 entries in the CAM, enter the following command at any level of the CLI:

```
BigIron> show cam ip 3/1
Slot    Index          IP_Address                MAC       Age    Out Port  VLAN
  3        1          198.38.5.4/24     0800.5afc.ea21        3    ethe 3/4    2
  3        2          198.38. 5.10/24   00a0.24d2.04ed        2    ethe 3/10   3
  3        3            198.38.7.6/24   00c0.2638.ac9c        0    ethe 3/2    2
```

Syntax: show cam ip <portnum> [<ip-addr> <ip-mask> | stat]

To show information for a specific MAC address, enter the address as in the following example:

```
BigIron> show cam ip  3/1 198.38.5.4 255.255.255.0
Slot   Index           IP_Address                MAC       Age    Out Port  VLAN
  3        1         198.38.5.4/24     0800.5afc.ea21        3    ethe 3/4    2
```

This command shows the following information.

Table 43.3: CLI Display of Layer 3 CAM Information

This Field...	Displays...
Slot	The chassis slot number.
Index	The entry number for this CAM entry.
IP Address	The IP address of the entry.
MAC	The MAC address of the entry.
Age	The age of the entry.
Out Port	The port on which traffic from the IP address gets forwarded.
VLAN	The port-based VLAN that contains the entry's MAC address.

To display Layer 4 entries in the CAM, enter the following command at any level of the CLI. This is an example of the command entered on a Jetcore device. The first CAM entry is flow-based, while the next three are rule-based. (For IronCore devices, only flow-based CAM entries are displayed.)

```
4802 Router# show cam l4 3
Sl Index       Src IP_Addr  SPort      Dest IP_Addr  DPort Prot Age    Out Port
 1 20480      10.10.10.1/32  1750     192.168.2.1/32    23  TCP  73   Use L2/L3
 1 21674      11.11.11.1/24    80   192.168.2.253/24    69  UDP dis   Use L2/L3
     Port Ranges -->           95
 1 21676      11.11.11.1/24    96   192.168.2.253/24    69  UDP dis   Use L2/L3
     Port Ranges -->           99
 1 21678               Any   N/A               Any   N/A  Any dis   Discard
```

The three rule-based CAM entries correspond to the following ACL:

```
access-list 101 permit udp 11.11.11.1/24 range 80 99 192.168.2.253/24 eq tftp
```

Syntax: show cam l4 <portnum>

This command shows the following information.

Table 43.4: CLI Display of Layer 4 CAM Information

This Field...	Displays...
Sl	The chassis slot number.

Table 43.4: CLI Display of Layer 4 CAM Information (Continued)

This Field...	Displays...
Index	The entry number for this CAM entry.
Src IP_Addr	The source IP address for the CAM entry.
SPort	The port on which the source IP address was learned.
Port Ranges -->	For the SPort column, indicates the range of source ports for the CAM entry; for example, ports 80 – 95 or 96 – 99.
Dest IP_Addr	The destination IP address for the CAM entry.
DPort	The port on which the destination IP address was learned.
Prot	The Layer 4 protocol.
Age	The age of the CAM entry.
Out Port	Information about how the packet is to be forwarded. This can be one of the following: Discard — Indicates that the CAM entry's corresponding PRAM entry is programmed to drop the packet. <Fid value> — Indicates that the CAM entry's corresponding PRAM entry is programmed to use the specified FID value. mgmt — Indicates that the CAM entry's corresponding PRAM entry is programmed to use the CPU FID. Use L2/L3 — Indicates that the CAM entry's corresponding PRAM entry is programmed to allow the forwarding decision to be made at Layer 2 or Layer 3. IronCore devices always display “Use L2/L3” in the Out Port field.

Possible values: See above

Default value: N/A

show cam-partition

Displays CAM partitioning information for each master DMA.

EXAMPLE:

```
SW-BI8000-doc# show cam-partition brief

==== SLOT 1 CAM PARTITION ====

DMA: 0 (0x00)
Number of CAM devices per DMA:  8
Number of hw entries per CAM:  0x00800
Total size of CAM = 1Mbits
complete CAM index range per DMA:
  (sw) 1 - 16383  (1 - 0x03fff), total entries: 16383 (0x03fff)
  (hw) 0 - 16383  (0 - 0x03fff), total entries: 16384 (0x04000)
Percentage of CAM hardware entries for each partition:
  Level3 l3 = 2047 (0.124938Mbits)      (12.493896%)
  Level3 l3 = 2048 (0.125Mbits) (12.5%)
  Level3 l3 = 8192 (0.5Mbits)   (50%)
  Level4    = 4096 (0.25Mbits)  (25%)

DMA: 2 (0x02)
Number of CAM devices per DMA:  8
Number of hw entries per CAM:  0x00800
Total size of CAM = 1Mbits
complete CAM index range per DMA:
  (sw) 1 - 16383  (1 - 0x03fff), total entries: 16383 (0x03fff)
  (hw) 0 - 16383  (0 - 0x03fff), total entries: 16384 (0x04000)
Percentage of CAM hardware entries for each partition:
  Level3 l3 = 2047 (0.124938Mbits)      (12.493896%)
  Level3 l3 = 2048 (0.125Mbits) (12.5%)
  Level3 l3 = 8192 (0.5Mbits)   (50%)
  Level4    = 4096 (0.25Mbits)  (25%)
```



```
SW-BI8000-doc# show cam-partition detail

==== SLOT 1 CAM PARTITION ====

DMA: 0 (0x00)
Number of CAM devices per DMA:  8
Number of hw entries per CAM:  0x00800
Total size of CAM = 1Mbits
complete CAM index range per DMA:
  (sw) 1 - 16383  (1 - 0x03fff), total entries: 16383 (0x03fff)
  (hw) 0 - 16383  (0 - 0x03fff), total entries: 16384 (0x04000)
Percentage of CAM hardware entries for each partition:
  Level3 l3 = 2047 (0.124938Mbits)      (12.493896%)
  Level3 l3 = 2048 (0.125Mbits) (12.5%)
  Level3 l3 = 8192 (0.5Mbits)   (50%)
  Level4    = 4096 (0.25Mbits)  (25%)

DMA: 2 (0x02)
Number of CAM devices per DMA:  8
Number of hw entries per CAM:  0x00800
Total size of CAM = 1Mbits
complete CAM index range per DMA:
  (sw) 1 - 16383  (1 - 0x03fff), total entries: 16383 (0x03fff)
  (hw) 0 - 16383  (0 - 0x03fff), total entries: 16384 (0x04000)
Percentage of CAM hardware entries for each partition:
  Level3 l3 = 2047 (0.124938Mbits)      (12.493896%)
  Level3 l3 = 2048 (0.125Mbits) (12.5%)
  Level3 l3 = 8192 (0.5Mbits)   (50%)
  Level4    = 4096 (0.25Mbits)  (25%)

L3 level 3 index range:
  (sw) 1 - 2047         (0x00001 - 0x007ff), free 2047 (0x007ff)
  (hw) 1 - 2047         (0x00001 - 0x007ff)
L3 level 2 index range:
  (sw) 2048 - 4095      (0x00800 - 0x00fff), free 2048 (0x00800)
  (hw) 2048 - 4095      (0x00800 - 0x00fff)
L3 index range:
  (sw) 4096 - 12287     (0x01000 - 0x02fff), free 8189 (0x01ffd)
  (hw) 4096 - 12287     (0x01000 - 0x02fff)
L4 pool 0 index range:
  (sw) 12288 - 14335    (0x03000 - 0x037ff), free 2044 (0x007fc)
  (hw) 12288 - 14335    (0x03000 - 0x037ff)
L2/L4 pool 1 index range:
  (sw) 14336 - 16383    (0x03800 - 0x03fff), free 2047 (0x007ff)
  (hw) 14336 - 16383    (0x03800 - 0x03fff)
```

Syntax: show cam-partition brief | detail

Syntax: show cam-partition module <module> brief | detail

Possible values: Module number

Default value: N/A

show chassis

Displays the status of power supplies and fans in the chassis.

EXAMPLE:

```
BigIron# show chassis
power supply 1 ok
power supply 2 not present
power supply 3 not present
power supply 4 not present
power supply 1 to 4 from bottom to top
fan 1 (rear/back panel, top fan) ok
fan 2 (rear/back panel, bottom fan) ok
fan 3 (top panel, fan) bad
fan 4 (top panel, fan) ok
Current temperature : 32.5 C degrees
Warning level : 45 C degrees, shutdown level : 55 C degrees
Boot Prom MAC: 00e0.52a9.bb00
```

Syntax: show chassis

NOTE: In a 15-slot chassis, the state of fans 5 and 6 is listed as "unmanaged". The software is unable to read the status of these fans.

Possible values: N/A

Default value: N/A

show clock

Displays the current settings for the on-board time counter and Simple Network Time Protocol (SNTP) clock, if configured.

EXAMPLE:

```
BigIron# show clock
```

Syntax: show clock [detail]

Possible values: N/A

Default value: N/A

show configuration

Lists the operating configuration of a Foundry device and allows you to check configuration changes before saving them to flash.. In software releases 07.6.03 and later, this command also displays the size of the running-config file.

EXAMPLE:

```
BigIron# show configuration
```

Syntax: show configuration

Possible values: N/A

Default value: N/A

show default

Displays the defaults for system parameters.

If you specify "default" but not the optional "values", the default states for parameters that can either be enabled or disabled are displayed. If you also specify "values", the default values for parameters that take a numeric value are displayed.

You can reconfigure the system parameters displayed by the “values” option using the system-max command. See “system-max” on page 6-131.

EXAMPLE:

Here are some examples of the information displayed by these commands. The first example shows the information displayed by the **show default** command on a BigIron Layer 3 Switch.

NOTE: If the information scrolls off the screen, you can enable page-display mode. See “page-display” on page 5-39.

```
BigIron# show default
spanning tree disabled
auto sense port speed      port untagged              port flow control on
no username assigned       no password assigned       boot sys flash primary
system traps enabled       sntp disabled              radius disabled
rip disabled               ospf disabled              bgp disabled

when ip routing enabled :
ip irdp enabled            ip load-sharing enabled    ip proxy arp enabled
ip rarp enabled            ip bcast forward enabled
dvmrp disabled             pim/dm disabled
vrrp disabled             fsrp disabled

when rip enabled :
rip type:v2 only           rip poison rev enabled

ipx disabled               appletalk disabled
```

EXAMPLE:

Here is an example of the same command entered on a FastIron II Layer 2 Switch:

```
FastIronII> show default
spanning tree disabled
auto sense port speed      port untagged              port flow control on
no username assigned       no password assigned       boot sys flash primary
system traps enabled       sntp disabled              radius disabled
rip disabled               ospf disabled              bgp disabled

when ip routing enabled :
ip irdp enabled            ip load-sharing enabled    ip proxy arp enabled
ip rarp enabled            ip bcast forward enabled
dvmrp disabled             pim/dm disabled
vrrp disabled              fsrp disabled

ipx disabled               appletalk disabled
```


EXAMPLE:

The following is an example of the command output when you use the **values** option on a BigIron Layer 3 Switch.

```
BigIron#sh 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

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               8000       64000      8000
ip-static-arp        2048       4096       2048
atalk-route          1024       3072       1024
atalk-zone-port      64         255        64
atalk-zone-sys       768        2048       768
multicast-route      16         8192       16
dvmrp-route          2048       128000     2048
multicast-flow       1024       2048       1024
dvmrp-mcache         512        2048       512
pim-mcache           1024       4096       1024
igmp-max-group-addr  8192       16000      8192
ip-cache             128000     400000     128000
ip-filter-port       512        8192       512
ip-filter-sys        1024       16384      1024
ipx-forward-filter   256        1024       256
ipx-rip-entry        3072       32768      3072
ipx-rip-filter       256        1024       256
ipx-sap-entry        6144       32768      6144
ipx-sap-filter       256        1024       256
l3-vlan              32         4095       32
ip-qos-session       2048       64000      2048
l4-real-server       1024       2048       1024
l4-virtual-server    256        512        256
l4-server-port       2048       4096       2048
mac                  8000       64000      8000
ip-route             128000     400000     128000
ip-static-route      512        4096       512
vlan                 32         4095       32
spanning-tree        32         128        32
mac-filter-port      32         1024       32
mac-filter-sys       64         2048       64
```



```
ip-subnet-port       24         128        24
session-limit        262144     1000000    262144
view                 10         65535      10
virtual-interface    255        4095       255
l2-mac               65536      2000000    65536
vpls-mac             65536      2000000    65536
vpls-session         32768      500000     32768
vpls-num             2048       16384      2048
```

EXAMPLE:

Here is an example of the information displayed by the same command entered on a FastIron II:

```
FastIronII> show default values
sys log buffers:50         mac age time:300 sec       bootp relay max hops:4
telnet sessions:5
ip arp age:20 min          igmp group memb.:140 sec   igmp query:60 sec
ip ttl:64 hops             max #ip addr per intf:24 ip metric:1
rip type:v2 only           rip poison rev enabled
ospf dead:40 sec           ospf hello:10 sec          ospf priority:1
ospf retrans:5 sec         ospf transit delay:1 sec
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    Minimum    Maximum    Default
arp                  1000       4000       1000
atalk-route          128        256        128
atalk-zone-port      8          64         32
atalk-zone-sys       64         256        64
dvmrp                0          2048       512
igmp                 0          1024       255
ip-cache             8000       16000      8000
ip-filter-port       16         64         16
ip-filter-sys        64         256        64
ipx-forward-filter   0          128        32
ipx-rip-entry        64         1024       256
ipx-rip-filter       0          128        32
ipx-sap-entry        128        1536       512
ipx-sap-filter       0          128        32
l3-vlan              0          128        8
ip-qos-session       128        1024       128
mac                  4000       8000       4000
ip-route             4000       8000       4000
ip-static-route      64         256        64
vlan                 8          128        8
mac-filter-port      32         64         32
mac-filter-sys       64         128        64
```

Syntax: show default [values]

Possible values: N/A

Default value: N/A

show dot1x

Displays information about the 802.1X configuration on the Foundry device.

EXAMPLE:

```
BigIron# show dot1x
PAE Capability:    Authenticator Only
system-auth-control: Enable
re-authentication: Disable
quiet-period:    60 Seconds
tx-period:    30 Seconds
supptimeout:    30 Seconds
servertimeout:     30 Seconds
maxreq:    2
re-authperiod:    3600 Seconds
security-hold-time: 60 Seconds
Protocol Version:    1
```

Syntax: show dot1x

Possible values: N/A

Default value: N/A

show dot1x config

Displays information about the 802.1X configuration on an individual port.

EXAMPLE:

```
BigIron# show dot1x config e 3/1
Port 3/1 Configuration:
Authenticator PAE state:    CONNECTING
Backend Authentication state:     IDLE
AdminControlledDirections:    BOTH
OperControlledDirections:    BOTH
AuthControlledPortControl:     Auto
AuthControlledPortStatus:    unauthorized
quiet-period:    60 Seconds
tx-period:    30 Seconds
supptimeout:    30 Seconds
servertimeout:     30 Seconds
maxreq:    2
re-authperiod:    3600 Seconds
security-hold-time: 60 Seconds
re-authentication: Disable
multiple-hosts: Disable
Protocol Version:    1
```

Syntax: show dot1x config <portnum>

Possible values: Port number

Default value: N/A

show dot1x statistics

Display 802.1X statistics for an individual port.

EXAMPLE:

```
BigIron# show dot1x statistics e 3/3

Port 3/3 Statistics:
RX EAPOL Start:    0
RX EAPOL Logoff:    0
RX EAPOL Invalid:    0
RX EAPOL Total:    0
RX EAP Resp/Id:    0
RX EAP Resp other than Resp/Id:    0
RX EAP Length Error:    0
Last EAPOL Version:    0
Last EAPOL Source:    0007.9550.0B83
TX EAPOL Total:    217
TX EAP Req/Id:    163
TX EAP Req other than Req/Id:    0
```

Syntax: show dot1x statistics <portnum>

Possible values: Port number

Default value: N/A

show fdp entry

Displays information about the Foundry device's Foundry or Cisco neighbors.

NOTE: To obtain the information, you must enable the Foundry Discovery Protocol (FDP), enable the Foundry device to intercept Cisco Discovery Protocol (CDP) packets, or both. See "fdp run" on page 6-29 and "cdp run" on page 6-19.

EXAMPLE:

To display CDP entries for all neighbors, enter the following command:

```
BigIron# show fdp entry *
Device ID: Router
Entry address(es):
  IP address: 207.95.6.143
Platform: cisco RSP4,  Capabilities: Router
Interface: Eth 1/1,  Port ID (outgoing port): FastEthernet5/0/0
Holdtime : 124 seconds
Version :
Cisco Internetwork Operating System Software
IOS (tm) RSP Software (RSP-JSV-M), Version 12.0(5)T1,  RELEASE SOFTWARE
(fc1)
Copyright (c) 1986-1999 by cisco Systems, Inc.
Compiled Thu 19-Aug-99 04:12 by cmong
```


EXAMPLE:

To display CDP entries for a specific device, specify the device ID. Here is an example.

```
BigIron# show fdp entry Router1
Device ID: Router1
Entry address(es):
  IP address: 207.95.6.143
Platform: cisco RSP4,  Capabilities: Router
Interface: Eth 1/1,  Port ID (outgoing port): FastEthernet5/0/0
Holdtime : 156 seconds
Version :
Cisco Internetwork Operating System Software
IOS (tm) RSP Software (RSP-JSV-M), Version 12.0(5)T1,  RELEASE SOFTWARE
(fc1)
Copyright (c) 1986-1999 by cisco Systems, Inc.
Compiled Thu 19-Aug-99 04:12 by cmong
```

Syntax: show fdp entry * | <device-id>

The ***** | <device-id> parameter specifies the device ID. If you enter *****, the detailed updates for all neighbor devices are displayed. If you enter a specific device ID, the update for that device is displayed.

Possible values: N/A

Default value: N/A

show fdp interface

Displays information about the Foundry device's Foundry neighbors.

NOTE: To obtain the information, you must enable the Foundry Discovery Protocol (FDP). See "fdp run" on page 6-29. This command is not supported for CDP.

EXAMPLE:

To display FDP information for an interface, enter a command such as the following:

```
BigIronA# show fdp interface ethernet 2/3
FastEthernet2/3 is up, line protocol is up
  Encapsulation ethernet
  Sending FDP packets every 5 seconds
  Holdtime is 180 seconds
```

This example shows information for Ethernet port 2/3. The port sends FDP updates every 5 seconds. Neighbors that receive the updates can hold them for up to 180 seconds before discarding them.

Syntax: show fdp interface [ethernet | pos | atm <portnum>]

The **ethernet | pos | atm** <portnum> parameter lists the information only for the specified interface.

Possible values: N/A

Default value: N/A

show fdp neighbors

Displays information about the Foundry device's Foundry or Cisco neighbors.

NOTE: To obtain the information, you must enable the Foundry Discovery Protocol (FDP), enable the Foundry device to intercept Cisco Discovery Protocol (CDP) packets, or both. See "fdp run" on page 6-29 and "cdp run" on page 6-19.

EXAMPLE:

```
BigIron# show fdp neighbors
Capability Codes: R - Router, T - Trans Bridge, B - Source Route Bridge
                  S - Switch, H - Host, I - IGMP, r - Repeater
(*) indicates a Cisco device

   Device ID      Local Int    Holdtm Capability Platform    Port ID
   -------------- ------------ ------ ---------- ----------- -------------
(*)Router         Eth 1/1      124    R          cisco RSP4
FastEthernet5/0/0
```

EXAMPLE:

```
BigIron# show fdp neighbors detail
Device ID: Router
Entry address(es):
  IP address: 207.95.6.143
Platform: cisco RSP4,  Capabilities: Router
Interface: Eth 1/1,  Port ID (outgoing port): FastEthernet5/0/0
Holdtime : 150 seconds
Version :
Cisco Internetwork Operating System Software
IOS (tm) RSP Software (RSP-JSV-M), Version 12.0(5)T1,  RELEASE SOFTWARE
(fc1)
Copyright (c) 1986-1999 by cisco Systems, Inc.
Compiled Thu 19-Aug-99 04:12 by cmong
```

Syntax: show fdp neighbor [ethernet | pos | atm <portnum>] [detail]

The **ethernet | pos | atm** <portnum> parameter lists the information only for updates received on the specified interface.

The **detail** parameter lists detailed information for each device.

Possible values: N/A

Default value: N/A

show fdp traffic

Displays Cisco Discovery Protocol (CDP) packet statistics.

NOTE: To obtain the information, you must enable the Foundry device to intercept Cisco Discovery Protocol (CDP) packets. See “cdp run” on page 6-19.

EXAMPLE:

```
BigIronA# show fdp traffic
CDP/FDP counters:
  Total packets output: 6, Input: 5
  Hdr syntax: 0, Chksum error: 0, Encaps failed: 0
  No memory: 0, Invalid packet: 0, Fragmented: 0
  Internal errors: 0
```

Syntax: show fdp traffic

Possible values: N/A

Default value: N/A

show flash

Displays the version of the software image saved in the primary and secondary flash of a Foundry device.

EXAMPLE:

```
BigIron# show flash
Active management module:
Code Flash Type: AMD 29F032B, Size: 64 * 65536 = 4194304, Unit: 2
Boot Flash Type: AMD 29F040, Size: 8 * 65536 = 524288
Compressed Pri Code size = 1357024, Version 07.1.05T51 (b2s07105.car)
Compressed Sec Code size = 1357024, Version 07.1.05T51 (b2s07105.car)
Maximum Code Image Size Supported: 2817536 (0x002afe00)
Boot Image size = 138692, Version 07.01.05 (boot7105.bin)
```

Syntax: show flash

Possible values: N/A

Default value: N/A

show interfaces

Displays information about interfaces on the Foundry device, including their state, duplex mode, STP state, priority and MAC address.

NOTE: If you have configured virtual routing interfaces (also called virtual interfaces or VEs) within port-based VLANs on a Layer 3 Switch, all ports within all virtual interfaces on the device share the same MAC address. See the second example in “show interfaces brief” on page 43-30.

EXAMPLE:

```
BigIron# show interfaces
GigabitEthernet1/1 is up, line protocol is up
  Hardware is GigabitEthernet, address is 00e0.52a9.bb00 (bia 00e0.52a9.bb00)
  Configured speed 1Gbit, actual unknown, configured duplex fdx, actual unknown
  Member of L2 VLAN ID 1, port is untagged, port state is BLOCKING
  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
  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
  0 packets output, 0 bytes, 0 underruns
  Transmitted 0 broadcasts, 0 multicasts, 0 unicasts
  0 output errors, 0 collisions, DMA transmitted 0 packets
```


NOTE: The MTU listed by this command is the transit MTU, which is the maximum packet size for transit traffic that the device can receive or transmit on this port. Transit traffic is traffic that is not addressed to the device itself but is instead switched or routed from one port to another through the device. To display the IP MTU instead, enter the **show ip interface** <port-type> <portnum> command.

Syntax: show interfaces [atm | ethernet | pos <portnum>] | [loopback <num>] | [slot <slot-num>] | [ve <num>]

Possible values: N/A

Default value: N/A

show interfaces brief

Shows a summary of Layer 2 information for all interfaces.

NOTE: If you have configured virtual routing interfaces (also called virtual interfaces or VEs) within port-based VLANs on a Layer 3 Switch, all ports within all virtual interfaces on the device share the same MAC address. See the second example below.

EXAMPLE:

```
BigIron# show interfaces brief
Port Link State     Dupl Speed Trunk Tag Priori MAC            Name
1/1  Down None      None None  None  No  level0 00e0.52f0.4f00
1/2  Down None      None None  None  No  level0 00e0.52f0.4f01
1/3  Down None      None None  None  No  level0 00e0.52f0.4f02
1/4  Down None      None None  None  No  level0 00e0.52f0.4f03
1/5  Down None      None None  None  No  level0 00e0.52f0.4f04
1/6  Down None      None None  None  No  level0 00e0.52f0.4f05
1/7  Down None      None None  None  No  level0 00e0.52f0.4f06
1/8  Down None      None None  None  No  level0 00e0.52f0.4f07
```

Syntax: show interfaces [ethernet | pos <portnum>] | [loopback <num>] | [slot <slot-num>] | [ve <num>] | [brief]

EXAMPLE:

This example shows information displayed on a Layer 3 Switch that contains virtual interfaces. Notice that all the ports within the virtual interfaces have the same MAC address as the first port on the device (port 1/1). This is true even if you configure multiple virtual interfaces in different VLANs. The MAC address for all the ports in the virtual interfaces is always the MAC address of port 1/1 (or port 1). The shared MAC address does not create networking conflicts on the Foundry device because the Foundry device maintains separate forwarding tables for each port-based VLAN. Therefore, the Foundry device can uniquely recognize an interface even if it shares a MAC address

with other interfaces. In this example, port 1/1 and all the ports within three virtual interfaces have MAC address 00e0.5295.b600. For completeness, this example includes the commands for creating the virtual interfaces.

```
BigIron(config)# vlan 2 by port
BigIron(config-vlan-2)# untag ethernet 1/3 to 1/4
BigIron(config-vlan-2)# router-interface ve 2
BigIron(config-vlan-2)# exit
BigIron(config)# vlan 3 by port
BigIron(config-vlan-3)# untag ethernet 1/7 to 1/8
BigIron(config-vlan-3)# router-interface ve 3
BigIron(config-vlan-3)# exit
BigIron(config)# vlan 8 by port
BigIron(config-vlan-8)# untag ethernet 1/5 ethernet 2/1 to 2/5
BigIron(config-vlan-8)# router-interface ve 8
BigIron(config-vlan-8)# exit
BigIron(config)# interface ve 2
BigIron(config-vif-2)# ip address 2.2.2.2/24
BigIron(config-vif-2)# exit
BigIron(config)# interface ve 3
BigIron(config-vif-3)# ip address 3.3.3.3/24
BigIron(config-vif-3)# exit
BigIron(config)# interface ve 8
BigIron(config-vif-8)# ip address 8.8.8.8/24
BigIron(config-vif-8)# exit

BigIron(config)# show interfaces brief
Port Link State     Dupl Speed Trunk Tag Priori MAC            Name
1/1  Down None      None None  None  No  level0 00e0.5295.b600
1/2  Down None      None None  None  No  level0 00e0.5295.b601
1/3  Down None      None None  None  No  level0 00e0.5295.b600
1/4  Down None      None None  None  No  level0 00e0.5295.b600
1/5  Down None      None None  None  No  level0 00e0.5295.b600
1/6  Down None      None None  None  No  level0 00e0.5295.b605
1/7  Down None      None None  None  No  level0 00e0.5295.b600
1/8  Down None      None None  None  No  level0 00e0.5295.b600
2/1  Down None      None None  None  No  level0 00e0.5295.b600
2/2  Down None      None None  None  No  level0 00e0.5295.b600
2/3  Down None      None None  None  No  level0 00e0.5295.b600
2/4  Down None      None None  None  No  level0 00e0.5295.b600
2/5  Down None      None None  None  No  level0 00e0.5295.b600
2/6  Down None      None None  None  No  level0 00e0.5295.b625
2/7  Down None      None None  None  No  level0 00e0.5295.b626
2/8  Down None      None None  None  No  level0 00e0.5295.b627
2/9  Down None      None None  None  No  level0 00e0.5295.b628
2/10 Down None      None None  None  No  level0 00e0.5295.b629
2/11 Down None      None None  None  No  level0 00e0.5295.b62a
2/12 Down None      None None  None  No  level0 00e0.5295.b62b
```

Possible values: N/A

Default value: N/A

show gvrp

Displays GVRP configuration information.

EXAMPLE:

```
BigIron(config)# show gvrp
GVRP is enabled on the system

GVRP BASE VLAN ID          : 4093
GVRP MAX Leaveall Timer    : 300000 ms

GVRP Join Timer            : 200 ms
GVRP Leave Timer           : 600 ms
GVRP Leave-all Timer       : 10000 ms

============================================================================
Configuration that is being used:

 block-learning ethe 1/3
 block-applicant ethe 2/7 ethe 2/11
 enable ethe 1/1 to 1/7 ethe 2/1 ethe 2/7 ethe 2/11

============================================================================

Spanning Tree: SINGLE SPANNING TREE
Dropped Packets Count: 0

============================================================================

Number of VLANs in the GVRP Database: 15
Maximum Number of VLANs that can be present: 4095

============================================================================
```

Syntax: show gvrp [ethernet <port-num>]

For information about this display, see the "Configuring GARP VLAN Registration Protocol (GVRP)" chapter in the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Possible values: N/A

Default value: N/A

show gvrp statistics

Displays GVRP statistics for a port.

EXAMPLE:

```
BigIron(config)# show gvrp statistics ethernet 2/1
PORT 2/1 Statistics:
 Leave All Received                    : 147
 Join Empty Received                   : 4193
 Join In Received                      : 599
 Leave Empty Received                  : 0
 Leave In Received                     : 0
 Empty Received                        : 588
 Leave All Transmitted                 : 157
 Join Empty Transmitted                : 1794
 Join In Transmitted                   : 598
 Leave Empty Transmitted               : 0
 Leave In Transmitted                  : 0
 Empty Transmitted                     : 1248
 Invalid Messages/Attributes Skipped   : 0
 Failed Registrations                  : 0
```

Syntax: show gvrp statistics all | ethernet <port-num>

For information about this display, see the "Configuring GARP VLAN Registration Protocol (GVRP)" chapter in the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Possible values: N/A

Default value: N/A

show gvrp vlan

Displays GVRP information about all the VLANs on the device.

EXAMPLE:

```
BigIron(config)# show gvrp vlan brief

Number of VLANs in the GVRP Database: 7
Maximum Number of VLANs that can be present: 4095

        [VLAN-ID]                          [MODE]       [VLAN-INDEX]

                1                  STATIC-DEFAULT                 0
                7                          STATIC                 2
               11                          STATIC                 4
             1001                         DYNAMIC                 7
             1003                         DYNAMIC                 8
             4093           STATIC-GVRP-BASE-VLAN                 6
             4094         STATIC-SINGLE-SPAN-VLAN                 5

==============================================================================
```


EXAMPLE:

To display detailed information for a specific VLAN, enter a command such as the following:

```
BigIron(config)# show gvrp vlan 1001

VLAN-ID: 1001, VLAN-INDEX: 7, STATIC: NO, DEFAULT: NO, BASE-VLAN: NO
Timer to Delete Entry Running: NO
Legend: [S=Slot]

Forbidden Members: None

Fixed Members: None

Normal(Dynamic) Members: (S2)  1
```

Syntax: show gvrp vlan all | brief | <vlan-id>

For information about these displays, see the "Configuring GARP VLAN Registration Protocol (GVRP)" chapter in the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Possible values: N/A

Default value: N/A

show interface ethernet <portnum> | ve <num> rate-limit

Displays configuration information and statistics for Adaptive Rate Limiting.

EXAMPLE:

```
BigIron(config-if-e1000-1/1)# show interface ethernet 1/1 rate-limit
Input
matches: access-group 101
params: 10000000 bps, 125000 limit, 187500 extended limit
conform 0 packets, 0 bytes; action: set-prec-transmit 5
exceeded 0 packets, 0 bytes; action: set-prec-transmit 0
last packet: 0ms ago, current burst: 0 bytes
last cleared: 0 days 00:08:05 ago, conformed 0 bps, exceeded 0 bps
Output
matches: access-group 103
params: 1000000 bps, 100000 limit, 100000 extended limit
conform 0 packets, 0 bytes; action: set-prec-transmit 5
exceeded 0 packets, 0 bytes; action: drop
last packet: 0ms ago, current burst: 0 bytes
last cleared: 0 days 00:00:04 ago, conformed 0 bps, exceeded 0 bps
```

NOTE: In compliance with RFC 1757, the input and output rates include the interpacket gap (sometimes called the interframe gap). The interpacket gap is the minimum idle period between transmission of frames and provides a recovery period to allow devices to prepare to receive the next frame. The minimum interpacket gap is 96 bit times, which is 9.6 microseconds for 10 Mbps Ethernet, 960 nanoseconds for 100 Mbps Ethernet, and 96 nanoseconds for 1 Gbps Ethernet.

The input and output rates are displayed in the "5 minute input rate" and "5 minute output rate" lines, not shown in the example above.

Syntax: show interface ethernet <portnum> | ve <num> rate-limit

Possible values: N/A

Default value: N/A

show ip

For Foundry Layer 2 Switches, this command displays the switch IP address and mask, its default router, the IP address of a TFTP server where configuration or image files are stored, if defined; and the file names of image and configuration files saved on that TFTP server.

For Foundry Layer 3 Switches, this command displays the global parameters for IP—specifically, router ID, IP TTL, ARP age values as well as all protocols and IP features enabled on the router. This command also displays all active filters.

EXAMPLE:

```
Global Settings
  ttl: 64, arp-age: 10, bootp-relay-max-hops: 4
  router-id : 10.1.1.1
  enabled : UDP-Broadcast-Forwarding  IRDP  Proxy-ARP  RARP  RIP  VRRP
  disabled: BGP4 Load-Sharing  RIP-Redist  OSPF  DVMRP  FSRP
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

Possible values: N/A

Default value: N/A

show ip access-lists

Displays the configured IP Access Control Lists (ACLs).

show ip acl-traffic

Displays statistics for packets permitted or denied by flow-based (CPU-based) ACLs.

EXAMPLE:

```
BigIron(config)# show ip acl-traffic

ICMP inbound packets received 400
ICMP inbound packets permitted 200
ICMP inbound packets denied 200
```

Syntax: show ip acl-traffic

The command lists a separate set of statistics for each of the following IP protocols:

- ICMP
- IGMP
- IGRP
- IP
- OSPF
- TCP
- UDP
- Protocol number, if an ACL is configured for a protocol not listed above

For TCP and UDP, a separate set of statistics is listed for each application port.

Possible values: N/A

Default value: N/A

show ip as-path-access-lists

Displays the configured IP AS-path ACLs, used for BGP4 filtering.

show ip bgp <ip-addr>

Displays routes that match a specified address and mask.

EXAMPLE:

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:

```
BigIron(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>

Possible values: see above

Default value: N/A

show ip bgp attribute-entries

Shows information entries in a Layer 3 Switch's BGP4 route attributes table. 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.

See the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide* for information about the fields in this display.

EXAMPLE:

```
BigIron# 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
```

Syntax: show ip bgp attribute-entries

Possible values: N/A

Default value: N/A

show ip bgp config

Displays the active BGP4 configuration information contained in the running-config. Use this command when you want to display only the active BGP4 configuration information, instead of the device's entire running-config.

EXAMPLE:

To display the device's active BGP4 configuration, enter the following command at any level of the CLI:

```
BigIron# 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 "rtr 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

Possible values: N/A

Default value: N/A

show ip bgp dampened-paths

Lists all the routes that have been dampened by the BGP4 route flap dampening feature.

EXAMPLE:

```
BigIron# show ip bgp dampened-paths
```

Syntax: show ip bgp dampened-paths

Possible values: N/A

Default value: N/A

show ip bgp filtered-routes

Displays the routes that the Layer 3 Switch has filtered out but retained for use by the soft reconfiguration feature.

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.

EXAMPLE:

To display the routes that have been filtered out, enter the following command at any level of the CLI:

```
BigIron# 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.

Possible values: See above

Default value: N/A

show ip bgp flap-statistics

Displays route flap dampening statistics. See the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide* for information about the fields in this display.

EXAMPLE:

```
BigIron# 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.

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.

NOTE: You also can display all the dampened routes by entering the following command: **show ip bgp dampened-paths**.

Possible values: See above

Default value: N/A

show ip bgp neighbors

Shows information about a Layer 3 Switch's BGP4 neighbors (peer BGP4 routers). See the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide* for information about the fields in this display.

EXAMPLE:

To display summary route information for a neighbor, enter a command such as the following at any level of the CLI:

```
BigIron(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
```


To display information for a specific neighbor, enter a command such as the following:

```
BigIron(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
```

NOTE: The display shows all the configured parameters for the neighbor. Only the parameters that have values different from their defaults are shown.

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:

```
BigIron# show ip bgp neighbors 20.20.20.2 advertised-routes
      There are 18690 routes advertised to neighbor 20.20.20.2
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
      Prefix             Next Hop        Metric     LocPrf     Weight Status
1     223.223.223.223/32 20.20.20.1                            0          BE
        AS_PATH: 65001
2     223.223.223.224/32 20.20.20.1                            0          BE
        AS_PATH: 65001
3     223.223.223.225/32 20.20.20.1                            0          BE
        AS_PATH: 65001
```


You also can enter a specific route, as in the following example:

```
BigIron# show ip bgp neighbors 20.20.20.2 advertised-routes 192.169.25.0/24
Status A:AGGREGATE B:BEST b:NOT-INSTALLED-BEST E:EBGP I:IBGP L:LOCAL
       Prefix             Next Hop        Metric     LocPrf     Weight Status
1      192.169.25.0/24    20.20.20.1                            0      BE
         AS_PATH: 65001
```

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 the description of the **soft-reconfiguration** option in “neighbor” on page 13-12.

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.

Some of the options accept the **detail** option. This option displays detailed information. For an example, see the "Displaying BGP4 Neighbor Information" section in the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

Possible values: see above

Default value: information for all neighbors is displayed

show ip bgp peer-group

Shows configuration information for peer groups.

NOTE: Only the parameters that are not set to their default values are listed. If a parameter setting is not listed, then that parameter is set to its default value.

EXAMPLE:

```
BigIron# 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>]

Possible values: see above

Default value: information for all peer groups is displayed

show ip bgp routes

Shows the BGP4 routes in a Layer 3 Switch's BGP4 route table. See the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide* for information about the fields in this display.

EXAMPLE:

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:

```
BigIron(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            : 0
  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
```

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:

```
BigIron(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
       H:HISTORY I:IBGP L:LOCAL M:MULTIPATH S:SUPPRESSED
       Network          ML Next Hop        Metric     LocPrf     Weight Status
1      4.2.42.0         24 192.168.4.211              101        0      B
         AS_PATH: 5
2      4.2.43.0         24 192.168.4.211              101        0      B
         AS_PATH: 5
3      7.7.7.0          24 192.168.4.211   0          101        0      b
         AS_PATH: 5
4      38.38.38.0       24 192.168.4.211   0          101        0      B
         AS_PATH: 5
13     102.0.0.0        24 200.1.1.10      12         101        32768  BL
         AS_PATH: 5
```

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 the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

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.

Possible values: see above

Default value: all routes are displayed

show ip bgp summary

Shows a summary of BGP4 configuration information for a Layer 3 Switch. See the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide* for information about the fields in this display.

EXAMPLE:

```
BigIron# 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
```

Syntax: show ip bgp summary

Possible values: N/A

Default value: N/A

show ip cache

Displays the IP host table showing indices to MAC addresses and the IP address of the next hop for Foundry Layer 3 Switches.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip cache

Total number of cache entries: 243

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  Fid
1     207.95.95.1      0.0.0.0       0000.0000.0000   PU    0
2     111.111.100.111  0.0.0.0       0000.0000.0000   PU    0
3     207.95.45.1      0.0.0.0       0000.0000.0000   PU    0
4     207.195.1.255    0.0.0.0       0000.0000.0000   PU    0
5     207.95.133.255   0.0.0.0       0000.0000.0000   PU    0

. . .  entries 6-242 not shown

243   207.95.42.1      0.0.0.0       0000.0000.0000   PU    0
```

Syntax: show ip cache [<ip-addr>] | [<num>]

The optional <num> parameter lets you display the table beginning with a specific entry number.

Possible values: N/A

Default value: N/A

show ip cache flow

Displays the main flow cache used by NetFlow on a Velocity Management Module.

EXAMPLE:

```
BigIron(config)# show ip cache flow
Flow Cache Active Timeout is 60 minutes
Flow Cache Inactive Timeout is 60 seconds

Flow Cache is Active for the following interfaces:
ethernet 3/1
ethernet 4/1

Input If     Src Prefix  Output If     Dst Prefix    Packets    Octets
     129       20.1.1.1        193       10.1.1.1         10       650
     129       20.1.1.1        193       10.1.1.1          8       600
     193       10.1.1.1        129       20.1.1.1         10       688
     193       10.1.1.1        129       20.1.1.1          7       602
```

Syntax: show ip cache flow

For information about the fields in this display, see the "Remote Network Monitoring" appendix in the *Foundry Enterprise Configuration and Management Guide*.

Possible values: N/A

Default value: N/A

show ip cache flow aggregation

Displays statistics and entries for a NetFlow aggregate cache.

EXAMPLE:

```
BigIron(config)# show ip cache flow aggregation as
IP Flow Aggregation Switching Cache, 2883584 bytes
1 active, 65535 inactive
70 ager polls, 0 flow alloc failures (cache space overflow)
Active Timeout 60 minutes, Inactive Timeout 60 seconds

Src AS  Dst AS  Input If  Output If     Flows     Pkts     Octets
    20       10       193        129         2       17       1290
    10       20       129        193         2       18       1250
```

This example shows information for the AS aggregate cache.

Here are examples of the output for the other types of aggregate caches. The following example shows the destination-prefix aggregate cache.

```
BigIron(config)# show ip cache flow aggregation destination-prefix
IP Flow Aggregation Switching Cache, 2883584 bytes
3 active, 65533 inactive
65 ager polls, 0 flow alloc failures (cache space overflow)
Active Timeout 60 minutes, Inactive Timeout 60 seconds

Output If     Dst Prefix  Msk    AS     Flows     Pkts     Octets
      193       10.1.1.1   24    20         2       18       1250
      129       20.1.1.1   24    10         2       17       1290
```

The following example shows the prefix aggregate cache.

```
BigIron(config)# show ip cache flow aggregation prefix
IP Flow Aggregation Switching Cache, 2883584 bytes
3 active, 65533 inactive
88 ager polls, 0 flow alloc failures (cache space overflow)
Active Timeout 60 minutes, Inactive Timeout 60 seconds

Input If     Src Prefix  Msk  Output If     Dst Prefix  Msk     Flows    Pkts
     129       20.1.1.1   24        193       10.1.1.1   24         2      18
     193       10.1.1.1   24        129       20.1.1.1   24         2      17
```

The following example shows the protocol-port aggregate cache.

```
BigIron(config)# show ip cache flow aggregation protocol-port
IP Flow Aggregation Switching Cache, 2883584 bytes
3 active, 65533 inactive
97 ager polls, 0 flow alloc failures (cache space overflow)
Active Timeout 60 minutes, Inactive Timeout 60 seconds

Protocol  Source Port  Dest Port     Flows     Pkts     Octets
       6         1360         23         1       10        688
       6           23       1360         1        8        600
       6         1361         23         1        7        602
       6           23       1361         1       10        650
```

The following example shows the source-prefix aggregate cache.

```
BigIron(config)# show ip cache flow aggregation source-prefix
IP Flow Aggregation Switching Cache, 2883584 bytes
3 active, 65533 inactive
59 ager polls, 0 flow alloc failures (cache space overflow)
Active Timeout 60 minutes, Inactive Timeout 60 seconds

Input If     Src Prefix  Msk    AS     Flows     Pkts     Octets
     193       10.1.1.1   24    20         2       17       1290
     129       20.1.1.1   24    10         2       18       1250
```


Most of the fields in these displays are the same for all the aggregate cache types, while a few fields are particular to each cache type. The **show ip cache flow aggregation** commands show the following information.

Syntax: show ip cache flow aggregation as | destination-prefix | prefix | protocol-port | source-prefix

For information about the fields in this display, see the "Remote Network Monitoring" appendix in the *Foundry Enterprise Configuration and Management Guide*.

Possible values: N/A

Default value: N/A

show ip client-pub-key

Displays the currently loaded public keys.

EXAMPLE:

```
BigIron# show ip client-pub-key

1024 65537 162566050678380006149460550286514061230306797782065166110686648548574
949573392322599631573796819248476346145327421786527672319957469414416047146826800
064453679033330420291249056907718288654183965655676902543288147725297813592782167540629478392662275128774861815448523997023618173312328476660721888873946758201
 user@csp_client

1024 35 152676199889856769693556155614587291553826312328095300428421494164360924
762074755452346792684432337622953129794188335259756957757051018052125410080748772658611985742270289700411216885214507408796984064240845174271455859236169370590874837875599405503479603024287131312793895007927438074972787423695977635251943 root@unix_machine

There are 2 authorized client public keys configured
```

Syntax: show ip client-pub-key

Possible values: N/A

Default value: N/A

show ip community-access-lists

Displays the configured IP community ACLs, which are used for BGP4 filtering.

show ip dr-aggregate

Displays cached default routes.

EXAMPLE:

To display the default route cache entries, enter the following command at any level of the CLI:

```
BigIron(config)# show ip dr-aggregate
```

Syntax: show ip dr-aggregate [<ip-addr>]

If you specify an IP address, only the entries for that destination are displayed.

Here is an example of the information displayed by this command.

```
BigIron(config)# show ip dr-aggregate
Total number of cache entries: 2
Start index: 1  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     22.22.22.22    /8  207.95.6.60     0044.052e.4302  DF    1/1   1     0
2     207.96.7.7     /12 207.95.6.60     0044.052e.4302  DF    1/1   1     0
```

This example shows two entries. The prefix associated with each entry is displayed. Notice that the prefix lengths in this example are different for each entry. The software selects a prefix length long enough to make the default network route entry unambiguous, so that is does not conflict with other cache entries.

To display the entry for a specific destination, enter the destination address, as shown in the following example.

```
BigIron(config)# show ip dr-aggregate 207.96.7.7
Total number of cache entries: 2
Start index: 1  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     207.96.7.7     /12 207.95.6.60     0044.052e.4302  DF    1/1   1     0
```

This example shows the second entry from the previous example, but the entry row number is 1. The row number identifies the row number in the displayed output. In addition, notice that the Total number of cache entries field shows 2, as in the previous example. The number in this field indicates the total number of default route aggregation entries in the forwarding cache.

Possible values: N/A

Default value: N/A

show ip dvmrp

Displays the global and interface settings for DVMRP on a Foundry Layer 3 Switch.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip dvmrp
Global Settings
  prune age: 180, neighbor timeout: 40
  probe interval: 10, report interval: 60
  route expire interval: 200, route discard interval: 340
  triggered update interval: 5, graft retransmit interval: 10
Interface Ethernet 1
TTL Threshold: 1 Metric: 1
Local Address: 192.094.005.001
. . .
Interface Ethernet 16
TTL Threshold: 1 Metric: 1
Local Address: 193.095.016.001
```

Syntax: show ip dvmrp

Possible values: N/A

Default value: N/A

show ip dvmrp flowcache

Displays all active IP DVMRP flows for a Foundry Layer 3 Switch. A ***flow*** is a cached forwarding entry.

EXAMPLE:

```
BigIron# show ip dvmrp flow-cache
```

Syntax: show ip flow-cache

Possible values: N/A

Default value: N/A

show ip dvmrp graft

Displays active DVMRP grafts. Information shown is port, source network, group address, neighbor router and age for a Foundry Layer 3 Switch configured for DVMRP operation.

This command is not supported on a Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip dvmrp graft
```

Syntax: show ip dvmrp graft

Possible values: N/A

Default value: N/A

show ip dvmrp group

Displays network address, mask and gateway and associated IP multicast group membership and port for a Foundry Layer 3 Switch configured for DVMRP operation.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip dvmrp group
```

Syntax: show ip dvmrp group [<group-address>]

Possible values: <group-address> is a multicast group address.

Default value: N/A

show ip dvmrp interface

Displays the interface DVMRP settings, TTL threshold and metric for all sub-nets (interfaces) for a Foundry Layer 3 Switch configured for DVMRP operation.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip dvmrp interface
Interface Ethernet 1
TTL Threshold: 1 Metric: 1 Enabled: Querier
```

Syntax: show ip dvmrp interface [ethernet <portnum> | ve <num>]

Possible values: The **ethernet** <portnum> parameter lets you specify a router port.

The **ve** <num> parameter lets you specify a virtual interface (VE).

Default value: N/A

show ip dvmrp mcache

Displays the DVMRP multicast cache for a Foundry Layer 3 Switch configured for DVMRP operation.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip dvmrp mcache
F:Fast S:Slow P:Prune L:Leaf
      SourceNet      GroupAddress    Type  PortMask & PruneMask
 1 207.095.002.000   226.000.000.019   P 15   F15.      P12
 2 207.095.002.000   226.000.000.021   P 15   F15.      P12
```

Syntax: show ip dvmrp mcache [<ip-addr>]

Possible values: The <ip-addr> parameter displays information for a specific source IP address.

Default value: N/A

show ip dvmrp nbr

Displays all neighbor DVMRP routers and the Foundry ports to which they are attached, for Foundry Layer 3 Switches configured for DVMRP operation.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip dvmrp nbr
Port   Neighbor          GenId      Age  UpTime
 11    207.095.018.001  -12198       40      900
Port   Neighbor          GenId      Age  UpTime
 12    207.095.009.040       0       40      900
Port   Neighbor          GenId      Age  UpTime
 14    207.095.008.030       0       40      130
```

Syntax: show ip dvmrp nbr

Possible values: N/A

Default value: N/A

show ip dvmrp prune

Displays active prunes on the network for a Foundry Layer 3 Switch configured for DVMRP operation.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip dvmrp prune
Port  SourceNetwork     GroupAddress      NbrRouter         Age UpTime
 11    207.095.002.000  226.000.000.027   207.095.018.001   180  0
 11    207.095.002.000  226.000.000.026   207.095.018.001   180  0
 11    207.095.002.000  226.000.000.025   207.095.018.001   180  0
```

Syntax: show ip dvmrp prune

Possible values: N/A

Default value: N/A

show ip dvmrp route

Displays network address, mask and gateway and associated IP multicast group membership and ports for a Foundry Layer 3 Switch with DVMRP configured.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip dvmrp route
```

Syntax: show ip dvmrp route [<ip-addr>]

Possible values: The <ip-addr> parameter displays information for a specific source IP address.

Default value: N/A

show ip dvmrp traffic

Displays all active DVMRP traffic on a Foundry Layer 3 Switch.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip dvmrp traffic
Port   Probe            Graft            Prune
       [Rx   Tx Discard] [Rx Tx Discard] [Rx   Tx Discard]
10      0      95    0     0     0    0     0      0        0
12     95      95    0     0     0    0    21      0        0
13     95      95    0     0     9    0     0      72       0
Tot   195     285    0     0     9    0    21      72       0
```

Syntax: show ip dvmrp traffic

Possible values: N/A

Default value: N/A

show ip filter-cache

Displays all active IP filter definitions for a Foundry Layer 2 Switch operating with Layer 3 switching.

This command is not supported on Foundry Layer 3 Switches.

EXAMPLE:

```
FastIron# show ip filter-cache
```

Syntax: show ip filter-cache

Possible values: N/A

Default value: N/A

show ip flow-cache

Displays all active IP flows for a Foundry Layer 3 Switch. A ***flow*** is a cached forwarding entry.

EXAMPLE:

```
BigIron# show ip flow-cache
```

Syntax: show ip flow-cache [<ip-addr>]

Possible values: IP address

Default value: N/A

show ip flow export

Displays configuration information and statistics for NetFlow.

EXAMPLE:

```
BigIron(config)# show ip flow export
Flow export is enabled
Version 5 flow records
Autonomous System information not included in export datagrams
COLLECTOR 1:
  Exporting using source interface Ethernet 3/1
  Collector IP address 10.1.1.1
  Collector UDP port 3065

COLLECTOR 2:
  Exporting using source interface Ethernet 3/2
  Collector IP address 10.1.1.2
  Collector UDP port 3066

COLLECTOR 3:
  Exporting using source interface Ethernet 3/3
  Collector IP address 20.1.2.3
  Collector UDP port 3000

Cache for protocol-port aggregation:
Aggregation flow export is enabled
Exporting using source interface Ethernet 3/5
Exporting flows to  10.2.2.2 (3233)

Cache for destination-prefix aggregation:
Aggregation flow export is enabled
Exporting using source interface Ethernet 3/4
Exporting flows to  10.2.2.2 (3235)

20000 flows exported in 15800 udp datagrams
0 flows failed to export
0 export packets were dropped
```

For information about the fields in this display, see the "Remote Network Monitoring" appendix in the *Foundry Enterprise Configuration and Management Guide*.

Syntax: show ip flow export

Possible values: N/A

Default value: N/A

show ip fsrp

Displays the current settings of FSRP on a Foundry Layer 3 Switch.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip fsrp
FSRP Interfaces currently defined:
Ethernet Interface: 1
ip fsrp ip address 192.147.200.165
ip fsrp virtual router ip address 192.147.200.100
ip fsrp other router ip address 192.147.200.170
ip fsrp state  Active
ip fsrp preference level 50
ip fsrp track port 3
ip fsrp keep alive time 15
```



```
ip fsrp router dead interval 30
```

Syntax: show ip fsrp

Possible values: N/A

Default value: N/A

show ip interface

Displays interface configuration details for all interfaces or a specified interface on a Layer 3 Switch router.

EXAMPLE:

To view all IP interfaces and their configuration, enter the following:

```
BigIron(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
```

EXAMPLE:

To view a specific interface configuration, enter a command such as the following:

```
BigIron# 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
```

NOTE: The MTU listed by this command is the IP MTU, which is the maximum size IP packet that the device can support for traffic addressed to the device itself. To display the transit MTU instead, enter the **show interfaces** command.

Syntax: show ip interface [ethernet <portnum>] | [loopback <num>] | [pos <portnum>] | [ve <num>]

Possible values: N/A

Default value: N/A

show ip mbgp <ip-addr>[/<prefix>]

Displays a specific MBGP route.

For information about the fields in this command's display, see the description of the **show ip bgp** <ip-addr> command in the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

show ip mbgp attribute-entries

Displays MBGP route attributes.

For information about the fields in this command's display, see the description of the **show ip bgp attribute-entries** command in the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

show ip mbgp config

Shows the MBGP configuration commands in the running-config.

EXAMPLE:

```
BigIron# show ip mbgp config
Current BGP configuration:
router bgp
 aggregate-address  192.1.0.0 255.255.0.0
 aggregate-address  192.1.0.0 255.255.0.0 nlri unicast multicast
 aggregate-address  207.95.0.0 255.255.0.0 nlri unicast multicast
 aggregate-address  207.95.0.0 255.255.0.0 summary-only
 as-path-filter  20 permit .
 local-as 20
 neighbor nj peer-group nlri unicast multicast
 neighbor 7.7.7.1 remote-as 30 nlri unicast multicast
 neighbor 7.7.7.1 shutdown
 neighbor 15.15.15.2 remote-as 40 nlri unicast multicast
 neighbor 38.38.38.1 remote-as 65097 nlri unicast multicast
 neighbor 1.1.1.1 peer-group nj
 neighbor 1.1.1.1 remote-as 2 nlri unicast multicast
 neighbor 10.8.20.6 remote-as 20 nlri unicast multicast
 neighbor 10.8.20.6 update-source loopback 1
 neighbor 10.8.20.6 route-map out newlocal
 neighbor 8.8.8.1 remote-as 40 nlri unicast multicast
 network 162.162.162.0 255.255.255.0 nlri unicast multicast
 redistribute connected route-map setcon
end
```

Syntax: show ip mbgp config

NOTE: This command displays exactly the same information as the **show ip bgp config** command. Each command displays both the BGP and MBGP configuration commands that are in the running-config.

Possible values: N/A

Default value: N/A

show ip mbgp dampened-paths

Displays MBGP paths that have been dampened by route flap dampening.

For information about the fields in this command's display, see the description of the **show ip bgp dampened-paths** command in the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

show ip mbgp filtered-routes

Displays MBGP routes that have been filtered out.

For information about the fields in this command's display, see the description of the **show ip bgp filtered-routes** command in the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

show ip mbgp flap-statistics

Displays route flap dampening statistics for MBGP routes.

For information about the fields in this command's display, see the description of the **show ip bgp flap-statistics** command in the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

show ip mbgp neighbors

Displays information about MBGP neighbors.

EXAMPLE:

```
BigIron(config-bgp-router)# show ip mbgp neighbor 7.7.7.2
    Total number of BGP Neighbors: 6
1   IP Address: 1.1.1.1, AS: 2 (EBGP), RouterID: 0.0.0.0
    State: CONNECT, Time: 1h27m5s, KeepAliveTime: 60, HoldTime: 180
       PeerGroup: nj
    Messages:    Open    Update  KeepAlive Notification Refresh-Req
       Sent    : 0       0       0         0            0
       Received: 0       0       0         0            0
    Last Connection Reset Reason:Unknown
    Notification Sent:     Unspecified
    Notification Received: Unspecified
    Neighbor NLRI Negotiation:
      Peer configured for Unicast and Multicast Routes
    TCP Connection state: ESTABLISHED
       Byte Sent:   1346, Received: 1714918
       Local host:  7.7.7.1, Local  Port: 179
       Remote host: 7.7.7.2, Remote Port: 8179
       ISentSeq:      12122  SendNext:      13469  TotUnAck:          0
       TotSent:        1347  ReTrans:           0  UnAckSeq:      13469
       IRcvSeq:   886310126  RcvNext:   888025045  SendWnd:       16384
       TotalRcv:    1714919  DupliRcv:        601  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. 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 Neighbor NLRI Negotiation section (shown in bold type) lists the types of routes that this Layer 3 Switch can exchange with the MBGP neighbor.

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".

NOTE: The display shows all the configured parameters for the neighbor. Only the parameters that have values different from their defaults are shown.

Syntax: show ip mbgp neighbors [<ip-addr>]

The <ip-addr> parameter specifies the neighbor's IP address.

For information about the fields in this command's display, see the description of the **show ip bgp neighbors** command in the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

Possible values: N/A

Default value: N/A

show ip mbgp peer-group

Displays information about MBGP peer groups.

For information about the fields in this command's display, see the description of the **show ip bgp peer-group** command in the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

show ip mbgp routes

Displays MBGP routes.

EXAMPLE:

```
BigIron(config-bgp-router)# show ip mbgp routes
Total number of BGP Routes: 3389
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      10.10.2.0/24       38.38.38.1                 100        0      BE
         AS_PATH: 65097 65356
2      12.0.0.0/8         38.38.38.1                 100        0      BE
         AS_PATH: 65097 683 6509 24 1800 1239 5511 5511 2200 1305
3      12.4.125.0/24      38.38.38.1                 100        0      BE
         AS_PATH: 65097 683 11537 1239
4      12.6.92.0/24       38.38.38.1                 100        0      BE
         AS_PATH: 65097 683 11537 1239
5      12.150.219.0/24    38.38.38.1                 100        0      BE
         AS_PATH: 65097 683 11537 1239
6      24.144.0.0/18      38.38.38.1                 100        0      BE
         AS_PATH: 65097 683 11537 1239
7      24.221.128.0/19    38.38.38.1                 100        0      BE
         AS_PATH: 65097 683 11537 1239
8      24.221.160.0/19    38.38.38.1                 100        0      BE
         AS_PATH: 65097 683 11537 1239
9      35.0.0.0/8         38.38.38.1                 100        0      BE
         AS_PATH: 65097 683 11537 237
```

Syntax: show ip mbgp routes

For information about the fields in this command's display, see the description of the **show ip bgp routes** command in the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

Possible values: N/A

Default value: N/A

show ip mbgp summary

Displays summary MBGP configuration information and statistics.

EXAMPLE:

```
BigIron# show ip mbgp summary
  BGP4 Summary
  Router ID: 10.8.20.1   Local AS Number : 20
  Confederation Identifier : not configured
  Confederation Peers:
  Maximum Number of Paths Supported for Load Sharing : 1
  Number of Neighbors Configured : 6
  Number of Routes Installed : 3389
  Number of Routes Advertising to All Neighbors : 16936
  Number of Attribute Entries Installed : 750
  Neighbor Address  AS#   State   Time     Rt:Accepted Filtered Sent   ToSend
  1.1.1.1           2     CONN    0h 0m18s    0        0        0      3387
  7.7.7.1           30    ADMDN   0h16m56s    0        0        0      3387
  8.8.8.1           40    CONN    0h 0m18s    0        0        0      3387
  10.8.20.6         20    CONN    0h 0m 9s    0        0        0      3387
  15.15.15.2        40    ESTAB   0h16m35s    0        0        3387   0
  38.38.38.1        65097 ESTAB   0h16m44s    3388     0        1      0
```

Syntax: show ip mbgp summary

NOTE: This command's display looks similar to the display for the **show ip bgp config** command. However, the **show ip mbgp config** command lists only the MBGP neighbors, whereas the show ip bgp config command lists only the BGP neighbors.

For information about the fields in this display, see the description of the **show ip bgp summary** command in the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

Possible values: N/A

Default value: N/A

show ip mroute

Displays information about IP multicast routes.

EXAMPLE:

```
BigIron(config)# show ip mroute
Total number of Mroutes: 3389
Start index: 1  D:Connected  R:RIP  S:Static  O:OSPF *:Candidate default
0    10.10.2.0        255.255.255.0     38.38.38.1       e3/5     20
1    12.0.0.0         255.0.0.0         38.38.38.1       e3/5     20
2    12.4.125.0       255.255.255.0     38.38.38.1       e3/5     20
3    12.6.92.0        255.255.255.0     38.38.38.1       e3/5     20
4    12.150.219.0     255.255.255.0     38.38.38.1       e3/5     20
5    24.144.0.0       255.255.192.0     38.38.38.1       e3/5     20
6    24.221.128.0     255.255.224.0     38.38.38.1       e3/5     20
7    24.221.160.0     255.255.224.0     38.38.38.1       e3/5     20
8    35.0.0.0         255.0.0.0         38.38.38.1       e3/5     20
9    62.4.0.0         255.255.224.0     38.38.38.1       e3/5     20
10   62.4.64.0        255.255.224.0     38.38.38.1       e3/5     20
```

Possible values: N/A

Default value: N/A

show ip msdp peer

Displays MSDP peer information.

EXAMPLE:

```
BigIron(config-msdp-router)# show ip msdp peer

        Total number of MSDP Peers: 2

        IP Address           State
1       206.251.17.30        ESTABLISHED
        Keep Alive Time  Hold Time
        60               90

                             Message Sent         Message Received
        Keep Alive           2                    3
        Notifications        0                    0
        Source-Active        0                    640
        Last Connection Reset Reason:Reason Unknown
        Notification Message Error Code Received:Unspecified
        Notification Message Error SubCode Received:Not Applicable
        Notification Message Error Code Transmitted:Unspecified
        Notification Message Error SubCode Transmitted:Not Applicable
        TCP Connection state: ESTABLISHED
          Local host:  206.251.17.29, Local  Port: 8270
          Remote host: 206.251.17.30, Remote Port: 639
          ISentSeq:      16927  SendNext:     685654  TotUnAck:          0
          SendWnd:       16384  TotSent:      668727  ReTrans:           1
          IRcvSeq:    45252428  RcvNext:    45252438  RcvWnd:        16384
          TotalRcv:         10  RcvQue:            0  SendQue:           0
```

Syntax: show ip msdp peer

Possible values: N/A

Default value: N/A

show ip msdp sa-cache

Displays the Source Actives in the MSDP cache.

EXAMPLE:

```
BigIron(config-msdp-router)# show ip msdp sa-cache

Total Entry 4096, Used 1800 Free 2296
Index   SourceAddr  GroupAddr              Age
1     (100.100.1.254, 232.1.0.95), RP:206.251.17.41, Age:0
2     (100.100.1.254, 237.1.0.98), RP:206.251.17.41, Age:30
3     (100.100.1.254, 234.1.0.48), RP:206.251.17.41, Age:30
4     (100.100.1.254, 239.1.0.51), RP:206.251.17.41, Age:30
5     (100.100.1.254, 234.1.0.154), RP:206.251.17.41, Age:30
6     (100.100.1.254, 236.1.0.1), RP:206.251.17.41, Age:30
7     (100.100.1.254, 231.1.0.104), RP:206.251.17.41, Age:90
8     (100.100.1.254, 239.1.0.157), RP:206.251.17.41, Age:30
9     (100.100.1.254, 236.1.0.107), RP:206.251.17.41, Age:30
10    (100.100.1.254, 233.1.0.57), RP:206.251.17.41, Age:90
```


Syntax: `show ip msdp sa-cache`

Possible values: N/A

Default value: N/A

show ip msdp summary

Displays summary MSDP information.

EXAMPLE:

```
BigIron(config-msdp-router)# show ip msdp summary

MSDP Peer Status Summary
 KA: Keepalive SA:Source-Active NOT: Notification
 Peer Address     State          KA          SA          NOT
                                In     Out   In    Out   In   Out
 206.251.17.30    ESTABLISH   3      3     0     640   0     0
 206.251.17.41    ESTABLISH   0      3     651   0     0     0
```

Syntax: `show ip msdp summary`

Possible values: N/A

Default value: N/A

show ip multicast

Indicates whether IP multicast traffic reduction is enabled on a Foundry Layer 2 Switch, lists the operating mode (active or passive) and lists the active multicast receivers and the learned multicast groups.

This command is not supported on Foundry Layer 3 Switches.

EXAMPLE:

```
BigIron-SW(config)# show ip pim
IP multicast is enabled - Passive
VLAN ID 22
Active 5.5.5.1 Router Ports  3/4 3/10 5/3
Total number of Multicast Group: 1
   1 Multicast Group: 239.255.162.1, Port: 3/4 3/10 5/3
    IGMP Group Port:
    PIMv2 Group Port: 3/4 3/10 5/3
```

For information about the fields in this display, see the "Configuring IP Multicast Traffic Reduction" chapter in the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Syntax: show ip multicast

Possible values: N/A

Default value: N/A

show ip nat statistics

Displays Network Address Translation (NAT) statistics.

EXAMPLE:

```
BigIron(config)# show ip nat statistics

Total translations: 10 (0 static, 10 dynamic)
Hits: 10 Misses: 1
Expired translations: 1
Dynamic mappings:
 pool rtrpool: mask = 255.255.255.255
     start 192.168.2.79 end 192.168.2.79
     total addresses 1 overloaded
IP Fragments: saved 0, restored 0, timed out 0
Sess: Total 524288, Avail 524243, NAT 22

Inside global      Last Inside Local  xmit pkts  xmit bytes rx pkts    rx bytes  cnt
192.168.2.79       10.10.100.18       62         4012       42         4285      10
```

Syntax: show ip nat statistics

Possible values: N/A

Default value: N/A

show ip nat translation

Displays the currently active NAT translations.

EXAMPLE:

```
BigIron(config)# show ip nat translation
Pro Inside global      Inside local       Outside local      Outside global
--- 209.157.1.69     10.10.10.69       207.195.2.12      207.195.2.12
--- 209.157.1.72     10.10.10.2        207.195.4.69      207.195.4.69
```

Syntax: show ip nat translation

Possible values: N/A

Default value: N/A

show ip net-aggregate

Displays the entries in the Content Addressable Memory (CAM), when the **ip net-aggregate** feature is enabled. See “ip net-aggregate” on page 6-63.

EXAMPLE:

```
BigIron(config)# show ip net-aggregate
```

For an example and information about the command’s display, see the "Configuring IP" chapter of the *Foundry Enterprise Configuration and Management Guide*.

Possible values: N/A

Default value: N/A

show ip ospf area

Displays for all active OSPF areas, the following information:

- type of area—stub or normal
- cost (for stub area only)
- number of times the SPF (shortest path first) calculation is performed for the area

- number of area borders within the area
- number of AS boundary routers within the area
- number of link state advertisements (LSA) in the link state database of the area
- sum of LSA checksums in the area

NOTE: This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# 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>] database link-state
advertise | link-state-id | network | nssa | router | router-id <ip-addr> | sequence-number <num> | status <index> | summary]]

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.

The **database link-state** parameter lets you display information about the link state database:

- **advertise** displays link state by advertisement
- **link-state-id** displays link state by link-state ID
- **network** displays link state by network link
- **nssa** displays link state by NSSA
- **router** displays link state by router link
- **router-id** <ip-addr> displays link state by router ID
- **sequence-numbers** <num> displays link state by sequence number
- **status** <index> displays link state status
- **summary** displays link state by summary link

Possible values: N/A

Default value: N/A

show ip ospf border-routers

Shows entries for ABR and ASBR routers.

EXAMPLE:

```
BigIron# 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.

Possible values: IP address

Default value: N/A

show ip ospf config

Displays global and interface runtime configuration details for OSPF on a Foundry Layer 3 Switch.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip ospf config
Router OSPF: Enabled
Redistribution: Disabled
default OSPF Metric: 10
OSPF Area currently defined:
Area-ID          Area-Type Cost
0.0.0.0          normal    0
OSPF Interfaces currently defined:
Ethernet Interface: 1
ip ospf cost 1
ip ospf dead-interval 40
ip ospf hello-interval 10
ip ospf priority 1
ip ospf retransmit-interval 5
ip ospf transmit-delay 1
ip ospf area 0.0.0.0
Ethernet Interface: 2
ip ospf cost 1
ip ospf dead-interval 40
ip ospf hello-interval 10
ip ospf priority 1
ip ospf retransmit-interval 5
ip ospf transmit-delay 1
ip ospf area 0.0.0.0
```

Syntax: show ip ospf config

Possible values: N/A

Default value: N/A

show ip ospf database external-link-state

Displays information about external link state advertisements stored in the database.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron> show ip ospf database 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
```

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 database external-link-state** command to display the table. See the "Configuring OSPF" chapter of the *Foundry Enterprise Configuration and Management Guide* 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.

Possible values: see above

Default value: N/A

show ip ospf database link-state opaque-area

Displays information about OSPF-TE LSAs used with MPLS traffic engineering.

EXAMPLE:

```
NetIron# show ip ospf database link-state opaque-area

Area ID         Type    LS ID           Adv Rtr         Seq(Hex)   Age   Cksum
0               OpAr    1.0.0.0         3.3.3.3         80000006  1337  0x1a19
  Area-opaque TE LSA
  1 - router address (len 4): 3.3.3.3

Area ID      Type    LS ID           Adv Rtr         Seq(Hex)   Age      Cksum
0            OpAr    1.0.0.2         2.2.2.2         80000007  1333     0x88f1
  Area-opaque TE LSA
  2 - link (len 100):
    1 - link type (len 1): point-to-point(1)
    2 - link ID (len 4): 1.1.1.1
    3 - local i/f ip addr (len 4): 10.1.1.2
    4 - remote i/f ip addr (len 4): 10.1.1.1
    5 - TE metric (len 4):
    6 - max BW (len 4): 2372 Mbits/sec
    7 - max reservable BW (len 4): 2372 Mbits/sec
    8 - unreserved BW (len 32):
      Priority 0: 2372 Mbits/sec
      Priority 1: 2372 Mbits/sec
      Priority 2: 2372 Mbits/sec
      Priority 3: 2372 Mbits/sec
      Priority 4: 2372 Mbits/sec
      Priority 5: 2372 Mbits/sec
      Priority 6: 2372 Mbits/sec
      Priority 7: 2372 Mbits/sec
    9 - color (len 4): 0
```

Syntax: show ip ospf database link-state opaque-area

Possible values: N/A

Default value: N/A

show ip ospf general

Displays global status information about OSPF for a Foundry Layer 3 Switch, specifically:

- count of external Link State Advertisements (LSA)
- sum of external LSA checksums
- number of new LSAs originated by the router
- number of new LSAs received by the router

NOTE: This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip ospf gen
External LSA Counter          0
External LSA Checksum Sum     0000
Originate New LSA Counter     4
Rx New LSA Counter            4
```

Syntax: show ip ospf general

Possible values: N/A

Default value: N/A

show ip ospf interface

Displays information about all or a specific OSPF interface.

This command is not supported on Foundry Layer 2 Switches.

The following information is provided:

- OSPF interface parameters
- State of the interface
- IP address of the designated router
- IP address of the backup designated router

EXAMPLE:

```
BigIron# show ip ospf interface
Indx Port   IP Address   Area ID    OSPF Mode   Priority
1    1       2.0.0.1      0.0.0.0   enabled       1
Transit(sec)  Retrans(sec) Hello(sec) Dead(sec)   cost
     1          5            10          40         1
Type        D. Router    Backup D. Router events  state
broadcast    2.0.0.1     2.0.0.2    1            DRouter
Authentication-Key: None
```

Syntax: show ip ospf interface [<ip-addr>]

The <ip-addr> parameter displays the OSPF interface information for the specified IP address.

Possible values: N/A

Default value: N/A

show ip ospf database link-state

Displays the router, network, summary and summary ASBR link state advertisements. The **status** parameter provides a detailed display. The **advertise** parameter provides a summary.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip ospf database link-state status
Index: 1  Area ID: 0.0.0.0
Age(sec) Type     LS ID    Router     Seq(hex) Chksum(hex)
565     Summary  192.147.200.0   192.147.80.3  80000001   781f
```

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 LSA table. To determine an LSA packet's position in the table, enter the **show ip ospf database link-state** command to display the table. See the "Configuring OSPF" chapter of the *Foundry Enterprise Configuration and Management Guide* 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.

Possible values: N/A

Default value: N/A

show ip ospf neighbor

Displays information about all neighbor routers or a specific neighbor router.

NOTE: For releases prior to 07.6.03, this command displays the fields as shown in the "show ip ospf neighbor detail" on page 43-67.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron> 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
```

Syntax: show ip ospf neighbor [router-id <ip-addr>] | [<num>]

The **router-id** <num> parameter displays only the neighbor entries for the specified router.

The <num> parameter displays the table beginning at the specified entry number.

Possible values: see above

Default value: N/A

show ip ospf neighbor detail

NOTE: This command is available in software releases 07.6.03 and later.

Displays additional information than that displayed in the **show ip ospf neighbor** command, including:

- Op - The sum of the option bits in the Options field of the Hello packet
- Cnt - The number of LSAs that were retransmitted

```
BigIron# show ip ospf neighbor detail
Port        Address         Pri State      Neigh Address   Neigh ID    Ev Op Cnt
v2          20.20.20.20     1   FULL/BDR   20.20.20.10     2.2.2.2     11 2  0
3/9         30.30.30.30     1   FULL/BDR   30.30.30.20     2.2.2.2     11 2  0
```

Syntax: show ip ospf neighbor detail

Possible values: N/A

Default value: N/A

show ip ospf redistribute

Displays the routes that have been redistributed into OSPF.

EXAMPLE:

```
BigIron# 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:

```
BigIron# show ip ospf redistribute route 3.1.0.0 255.255.0.0
  3.1.0.0 255.255.0.0 static
```

Possible values: see above

Default value: N/A

show ip ospf routes

Displays the OSPF route table. See the "Configuring OSPF" chapter of the *Foundry Enterprise Configuration and Management Guide* for information about the fields in this display.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

To display OSPF route information, enter the following command at any CLI level:

```
BigIron> 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.

Possible values: see above

Default value: N/A

show ip ospf trap

Displays the list of all OSPF traps and their current state of enabled or disabled.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron(config)# 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
Originate MaxAge LSA Trap:  Enabled
Link State Database Overflow Trap:  Enabled
Link State Database Approaching Overflow Trap:  Enabled
```

Syntax: show ip ospf trap

Possible values: N/A

Default value: N/A

show ip ospf virtual-link

Displays transit area, router ID and transit specifics for an OSPF virtual link on a Foundry Layer 3 Switch.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip ospf virtual-link 1
Indx Transit Area   Router ID       Transit(sec) Retrans(sec) Hello(sec)
1    192.147.60.0   192.147.180.30   1            5            10
Dead(sec)      events           state        Authentication-Key
40             0                down         None
```

Syntax: show ip ospf virtual-link [<num>]

The <num> parameter displays the table beginning at the specified entry number.

Possible values: see above

Default value: N/A

show ip ospf virtual-neighbor

Displays the OSPF virtual neighbor information.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ip ospf virtual-neighbor 3
```

Syntax: show ip ospf virtual-neighbor [<num>]

The <num> parameter displays the table beginning at the specified entry number.

Possible values: see above

Default value: N/A

show ip pim

Displays IP multicast information on a Layer 2 Switch.

NOTE: This command applies to Layer 2 Switches only. To display IP PIM information on a Layer 3 Switch, use the other **show ip pim** commands listed in this chapter.

EXAMPLE:

```
BigIron-SW(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
```

For information about the fields in this display, see the "Configuring IP Multicast Traffic Reduction" chapter in the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Syntax: show ip pim

Possible values: N/A

Default value: N/A

show ip pim bsr

Shows Bootstrap router (BSR) information for PIM Sparse.

EXAMPLE:

To display BSR information, enter the following command at any CLI level:

```
BigIron(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.

```
BigIron(config-pim-router)# show ip pim bsr
PIMv2 Bootstrap information
 local BSR address  = 207.95.7.1
 local BSR priority = 5
```


See the "Configuring IP Multicast Protocols" chapter of the *Foundry Enterprise Configuration and Management Guide* for an explanation of the information shown by this command.

Syntax: show ip pim bsr

Possible values: see above

Default value: N/A

show ip pim flowcache

Displays all active PIM flows for a Foundry Layer 3 Switch. A ***flow*** is a cached forwarding entry.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron(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
```

See the "Configuring IP Multicast Protocols" chapter of the *Foundry Enterprise Configuration and Management Guide* for an explanation of the information shown by this command.

Syntax: show ip pim flowcache

Possible values: N/A

Default value: N/A

show ip pim group

Displays all active PIM groups by interface—both physical and virtual—for a Foundry Layer 3 Switch. Physical ports are displayed as numerals only. Virtual interfaces are preceded with a 'v' as in the example below.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron(config)# show ip pim group
Index   Group                                       Port
1       224.2.230.64                                v01
2       239.255.0.1                                 v01
```

See the "Configuring IP Multicast Protocols" chapter of the *Foundry Enterprise Configuration and Management Guide* for an explanation of the information shown by this command.

Syntax: show ip pim group

Possible values: N/A

Default value: N/A

show ip pim interface

Lists all active interfaces configured for a Foundry Layer 3 Switch.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron(config)# sh ip pim interface
```



```
Interface Ethernet 1
TTL Threshold: 1, Enabled
Local Address: 207.95.18.20
Interface Ethernet 3
TTL Threshold: 1, Enabled
Local Address: 207.95.5.1
```

Syntax: show ip pim interface [ethernet <portnum> | ve <num>]

The **ethernet** <portnum> parameter lets you specify a router port.

The **ve** <num> parameter lets you specify a virtual interface (VE).

Possible values: N/A

Default value: N/A

show ip pim mcache

Displays all forwarding entries for a Foundry Layer 3 Switch with PIM enabled.

This command is not supported on Foundry Layer 2 Switches.

In the example below, the source, group pair is defined for ports 2 and 3 as listed in hex in the PortMask column.

EXAMPLE:

```
BigIron(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 [<source> <group>]

Possible values: N/A

Default value: N/A

show ip pim nbr

Displays all PIM neighbor routers for physical, virtual and tunnel interfaces.

Port numbers preceded by a ‘T’ are tunnel interfaces, ‘E’ refers to physical interfaces and ‘VE’ refers to routed interfaces within a VLAN.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron(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

See the “Configuring IP Multicast Protocols” chapter of the *Foundry Enterprise Configuration and Management Guide* for an explanation of the information shown by this command.

Syntax: show ip pim nbr

Possible values: N/A

Default value: N/A

show ip pim prune

Shows those prune states that are active on a Foundry Layer 3 Switch with PIM enabled.

Port numbers preceded by a ‘T’ are tunnel interfaces, ‘E’ refers to physical interfaces and ‘VE’ refers to routed interfaces within a VLAN.

This command is not supported on a Foundry Layer 2 Switch.

EXAMPLE:

```
BigIron(config)# show ip pim nbr
Port       SourceNet      Group         Nbr                Age
T16        207.95.5.0     239.255.0.2   207.95.6.10        0
```

Syntax: show ip pim prune

Possible values: N/A

Default value: N/A

show ip pim rp-candidate

Displays candidate Rendezvous Point (RP) information for PIM Sparse.

EXAMPLE:

To display candidate RP information, enter the following command at any CLI level:

```
BigIron(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.

```
BigIron(config-pim-router)# show ip pim rp-candidate

This system is not a Candidate-RP.
```

See the "Configuring IP Multicast Protocols" chapter of the *Foundry Enterprise Configuration and Management Guide* for an explanation of the information shown by this command.

Syntax: show ip pim rp-candidate

Possible values: N/A

Default value: N/A

show ip pim rp-hash

Shows RP information for a specific PIM Sparse group.

EXAMPLE:

To display RP information for a PIM Sparse group, enter the following command at any CLI level:

```
BigIron(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
```

See the "Configuring IP Multicast Protocols" chapter of the *Foundry Enterprise Configuration and Management Guide* for an explanation of the information shown by this command.

Syntax: show ip pim rp-hash <group-addr>

The <group-addr> parameter is the address of a PIM Sparse IP multicast group.

Possible values: N/A

Default value: N/A

show ip pim rp-map

Shows PIM Sparse RP-to-group mappings.

To display RP-to-group-mappings, enter the following command at any CLI level:

```
BigIron(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
```

See the "Configuring IP Multicast Protocols" chapter of the *Foundry Enterprise Configuration and Management Guide* for an explanation of the information shown by this command.

Syntax: show ip pim rp-map

Possible values: N/A

Default value: N/A

show ip pim rp-set

Shows the RP set list on a Layer 3 Switch configured as a PIM Sparse router. In software releases 07.6.03 and later, this command also displays the ACLs used to filter static RP groups.

EXAMPLE:

To display the RP set list, enter the following command at any CLI level:

```
BigIron(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
```

See the "Configuring IP Multicast Protocols" chapter of the *Foundry Enterprise Configuration and Management Guide* for an explanation of the information shown by this command.

Syntax: show ip pim rp-set

Possible values: N/A

Default value: N/A

show ip pim sparse

Shows global PIM Sparse parameters.

EXAMPLE:

To display PIM Sparse configuration information, enter the following command at any CLI level:

```
BigIron(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
```

See the "Configuring IP Multicast Protocols" chapter of the *Foundry Enterprise Configuration and Management Guide* for an explanation of the information shown by this command.

Syntax: show ip pim sparse

Possible values: N/A

Default value: N/A

show ip pim traffic

Displays active PIM interfaces and their statistics for a Foundry Layer 3 Switch.

Port numbers preceded by a 'T' are tunnel interfaces, 'E' refers to physical interfaces and 'VE' refers to routed interfaces within a VLAN.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron(config)# show ip pim traffic
Port       Hello          Join           Prune          Graft          Assert
          [Rx   Tx]     [Rx   Tx]      [Rx   Tx]     [Rx   Tx]     [Rx   Tx]
e5          0    2         0    0          0    0         0    0         0    0
t1          538  540       0    0          3    775       0    4         0    0
ve1         0    541       0    0          0    0         0    0         0    0
ve3         0    541       0    0          0    0         0    0         0    0
Total       538  2163      0    0          33   775       0    4         0    0
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
```

This example shows output for regular PIM (dense mode) and PIM Sparse. The regular PIM statistics are listed first, followed by the PIM Sparse statistics. Rows are displayed only for the type of PIM configured on the Layer 3 Switch. See the "Configuring IP Multicast Protocols" chapter of the *Foundry Enterprise Configuration and Management Guide* for an explanation of the information shown by this command.

Syntax: show ip pim traffic

Possible values: N/A

Default value: N/A

show ip policy

Displays the configured global and local session policies defined using the **ip policy** command.

This command does not apply to Layer 3 Switches.

EXAMPLE:

```
Index        Priority       Protocol      Socket  Type
  1          high           tcp           pop3    global
  2          high           udp           dns     global
```

Syntax: show ip policy

Possible values: N/A

Default value: N/A

show ip prefix-lists

Displays the configured IP prefix lists.

show ip rip

Displays the RIP filters defined for a Foundry Layer 3 Switch and its neighbor router.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron(config)# show ip rip
RIP Route Filter Table
Index                   Action              Route IP Address     Sub-net Mask
1                       Permit              192.58.5.3           255.255.255.0
RIP Neighbor Filter Table
Index                   Action              Neighbor IP address
1                       Permit              195.98.7.2
```

Syntax: show ip rip

Possible values: N/A

Default value: N/A

show ip route

Displays active IP routes on a Foundry Layer 3 Switch. See the "Configuring IP" chapter of the *Foundry Enterprise Configuration and Management Guide* for information about the fields in this display.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron> show ip route

Total number of IP routes: 514
Starting index: 1  B:BGP D:Directly-Connected  R:RIP  S:Static  O:OSPF
IA:OSPF inter area  E1:OSPF external type 1  E2:OSPF external type 2

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
```

Syntax: show ip route [<ip-addr> [<ip-mask>] [longer] [none-bgp]] | <num> | bgp | direct | ospf | rip | static | tunnel]

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 **tunnel** option displays only routes using an MPLS LSP.

Possible values: see above

Default value: N/A

show ip ssh

Displays information about the SSH management sessions in effect on the device. Up to five SSH connections can be active on the Foundry device. For information about this display and about using SSH, see the "Configuring Secure Shell" chapter.

EXAMPLE:

```
BigIron# show ip ssh
Connection     Version     Encryption     State     Username
    1            1.5         ARCFOUR       0x82      neville
    2            1.5           IDEA        0x82      lynval
    3            1.5           3DES        0x82      terry
    4            1.5           none        0x00
    5            1.5           none        0x00
```

Syntax: show ip ssh

Possible values: N/A

Default value: N/A

show ip static-arp

Displays the static ARP table.

EXAMPLE:

```
BigIron# 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 <portnum> | mac-address <xxxx.xxxx.xxxx> [<mask>] | <ip-addr> [<ip-mask>]] [<num>]

The **ethernet** <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.

Possible values: N/A

Default value: N/A

show ip tcp connections

Displays information about each TCP connection on the device, including the local IP address, local port number, remote IP address, remote port number and the state of the connection. In addition, the command displays the percentage of free memory for each of the internal TCP buffers.

EXAMPLE:

```
BigIron# show ip tcp connections
Local IP address : port  <-> Remote IP address : port  TCP state
TCP: 10.10.10.25 : 23  <-> 10.10.10.15      : 2465  ESTABLISHED
TCP: 10.10.10.25 : 80  <-> 10.10.10.30      : 4026     FIN-WAIT-2
TCP: 10.10.10.25 : 22  <-> 10.10.10.50      : 3578     ESTABLISHED
TCP: 10.10.10.25 : 23  <-> 10.10.10.15      : 2468  ESTABLISHED
TCP: 10.10.10.25 : 23  <-> 10.10.10.15      : 2466  ESTABLISHED
Total 5 TCP connections

TCP MEMORY USAGE PERCENTAGE
FREE TCB = 96 percent
FREE TCP QUEUE BUFFER = 62 percent
FREE TCP SEND BUFFER = 25 percent
FREE TCP RECEIVE BUFFER = 100 percent
FREE TCP OUT OF SEQUENCE BUFFER = 100 percent
```

Syntax: show ip tcp connections

Possible values: N/A

Default value: N/A

show ip tcp status

Displays detailed information about a specified TCP connection, including the sequence and ACK numbers, window sizes, and available buffer sizes.

EXAMPLE:

```
BigIron# show ip tcp status 10.10.10.25 23 10.10.10.15 2465
TCP: TCB = 0x210de40a
TCP: 10.10.10.25:23 <-> 10.10.10.15:2465: state: ESTABLISHED
  Send: initial sequence number = 1453320
  Send: first unacknowledged sequence number = 1532710
  Send: current send pointer = 1532710
  Send: next sequence number to send = 1532710
  Send: remote received window = 0
  Send: total unacknowledged sequence number = 3773
  Send: total used buffers 43
  Receive: initial incoming sequence number = 17806845
  Receive: expected incoming sequence number = 17846856
  Receive: received window = 16384
  Receive: bytes in receive queue = 0
  Receive: congestion window = 1460
```

Syntax: show ip tcp status <local IP address> <local port> <remote IP address> <remote port>

Possible values: See above

Default value: N/A

show ip traffic

Displays IP (including ICMP, UDP, TCP, and RIP) traffic statistics for a Foundry device.

EXAMPLE:

```
BigIron# show ip traffic
IP Statistics
  464 received, 2267 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 sequence, 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:
  54 total, 0 errors, 0 unreachable, 0 time exceed
  0 parameter, 0 source sequence, 0 redirect, 0 echo,
  0 echo reply, 0 timestamp, 0 timestamp rely, 0 addr mask
  0 addr mask reply, 54 irdp advertisement, 0 irdp solicitation
```

NOTE: This example is an excerpt, not a complete display.

Syntax: show ip traffic

Possible values: N/A

Default value: N/A

show ip vrrp

Displays VRRP information.

EXAMPLE:

```
BigIron(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
```

To display detailed information for a Layer 3 Switch, enter the following command at any level of the CLI:

```
BigIron(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.

```
BigIron(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 of VRRP statistics.

```
BigIron(config-if-e1000-1/5-vrid-1)# show ip vrrp stat

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
```

Syntax: show ip vrrp brief | ethernet <portnum> | ve <num> | stat

The **brief** parameter displays the summary information. If you do not use this parameter, detailed information is displayed instead.

The **ethernet** <portnum> parameter specifies an Ethernet port. If you use this parameter, the command displays VRRP information only for the specified port.

The **ve** <num> parameter specifies a virtual interface. If you use this parameter, the command displays VRRP information only for the specified virtual interface.

The **stat** parameter displays statistics.

Possible values: N/A

Default value: N/A

show ip vrrp-extended

Displays VRRPE information.

EXAMPLE:

```
BigIron(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
```


Here is an example of detailed information for a VRRPE Backup.

```
BigIron(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
```

Here is an example of VRRPE statistics.

```
BigIron(config-if-e1000-1/5-vrid-1)# show ip vrrp-extended stat

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
```

Syntax: show ip vrrp-extended brief | ethernet <portnum> | ve <num> | stat

The **brief** parameter displays the summary information. If you do not use this parameter, detailed information is displayed instead.

The **ethernet** <portnum> parameter specifies an Ethernet port. If you use this parameter, the command displays VRRPE information only for the specified port.

The **ve** <num> parameter specifies a virtual interface. If you use this parameter, the command displays VRRPE information only for the specified virtual interface.

The **stat** parameter displays statistics.

Possible values: N/A

Default value: N/A

show ip vrrp vrid

Displays information about the settings configured for a specified VRRP Virtual Router ID (VRID).

EXAMPLE:

```
BigIron(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.

Possible values: N/A

Default value: N/A

show ipx

Displays IPX global parameters for a Foundry Layer 3 Switch.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ipx

Global Settings

IPX Routing Mode: Enabled

IPX NetBIOS (type 20): Disallowed
```

Syntax: show ipx

Possible values: N/A

Default value: N/A

show ipx cache

Displays summary by port, network number, forwarding (Next Hop Router), MAC address, out filter status and frame type for a Foundry device.

EXAMPLE:

```
BigIron# show ipx cache

Total number of IPX cache entries 3
Forwarding
Index  Network   Router      Out-Filter  Frame-Type        Port
1      11110007  0000.0000.0000  off      ethernet_802.3    7
2      11110005  0000.0000.0000  off      ethernet_802.3    5
3      32D564FA  00a0.24bf.89ca  off      ethernet_802.3    5
```

Syntax: show ipx cache [<num(hex)>]

Possible values: The optional <num(hex)> parameter lets you specify an IPX network number.

Default value: N/A

show ipx interface

Lists network number, MAC address, and port state and frame type for all interfaces or a specific IPX interface on a Foundry Layer 3 Switch.

To display data on all interfaces, enter the command **show ipx interface**.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

To display data for interface 5, enter the following:

```
BigIron# show ipx interface ethernet 3/5

Interface Ethernet 3/5
  MAC address: 00e0.5284.0b44  Port state: UP
  IPX network:       0000ABCD  Frame type: ethernet_snap   Allow NetBIOS: NO
  rip-interval: 60  rip-max-packet-size: 432  rip-multiplier: 3
  sap-interval: 60  sap-max-packet-size: 480  sap-multiplier: 3
```

Syntax: show ipx interface [ethernet <portnum> | ve <num>]

The **ethernet** <portnum> parameter lets you specify a router port.

The **ve** <num> parameter lets you specify a virtual interface (VE).

Possible values: see above

Default value: N/A

show ipx route

Displays active IPX routes noting hop, tick and port for a Foundry Layer 3 Switch.

EXAMPLE:

```
BigIron# show ipx route
Total number of IPX route entries 3
Forwarding
Index   Network   Router          Hops  Ticks Port
1       11110007  0000.0000.0000     0     1     7
2       32D564FA  00a0.24bf.89ca     1     2     5
3       11110005  0000.0000.0000     0     1     5
```


Syntax: show ipx route [<num(hex)>]

Possible values: The optional <num(hex)> parameter lets you specify an IPX network number.

Default value: N/A

show ipx servers

Displays IPX servers defined for a Foundry Layer 3 Switch.

This command is not supported on Foundry Layer 2 Switches.

EXAMPLE:

```
BigIron# show ipx servers
Total number of IPX server entries 3
Index  Network   Node            Socket    Type      Hops
1      32D564FA  0000.0000.0001   0005     026B        1
         Server-name: FoundryD
2      32D564FA  0000.0000.0001   4006     0278        1
         Server-name: FoundryM
3      32D564FA  0000.0000.0001   0451     0004        1
         Server-name: Foundry-MPR2
```

Syntax: show ipx servers [<name>]

Possible values: The optional <name> parameter lets you specify a server name.

Default value: N/A

show ipx traffic

Displays a port summary of total IPX packets forwarded. It also breaks down the packets by transmit and receive. Totals for dropped and filtered packets are also shown.

This command is supported on both Foundry Layer 2 Switches and Layer 3 Switches.

EXAMPLE:

```
BigIron# show ipx traffic
Dropped              Filtered
Port  Forward  Receive  Transmit  Receive  Transmit   Receive  Transmit
  5    46         36        8          2       0         0         0
  7     0          0        6          0       0         0         0
Tot    46         36       14          2       0         0         0
```

Syntax: show ipx traffic

Possible values: N/A

Default value: N/A

show isis config

Displays the global IS-IS configuration commands that are in effect on the device.

NOTE: The running-config does not list the default values. Only commands that change a setting or add configuration information are displayed.

EXAMPLE:

To list the global IS-IS configuration commands in the Layer 3 Switch's running-config, enter the following command at any level of the CLI:


```
NetIron(config-isis-router)# show isis config

Current IS-IS configuration:
router isis
 net  20.00e0.5200.0001.00
end
```

The running-config shown in this example contains the command that enables IS-IS and a command that configures an NET.

To display the interface configuration information in the running-config, enter one of the following commands at any level of the CLI:

- **show running-config**
- **write terminal**

Syntax: show isis config

Possible values: N/A

Default value: N/A

show isis counts

Displays IS-IS error statistics.

EXAMPLE:

```
NetIron(config-isis-router)# show isis counts
 Area Mismatch: 0
 Max Area Mismatch: 0
 System ID Length Mismatch: 0
 Authentication Fail: 0
 Corrupted LSP: 0
 LSP Sequence Number Skipped: 0
 LSP Max Sequence Number Exceeded: 0
 Level-1 Database Overload: 0
 Level-2 Database Overload: 0
 Our LSP Purged: 0
```

Syntax: show isis counts

For information about the fields in this display, see the "Configuring IS-IS" chapter in the *Foundry NetIron Service Provider Configuration and Management Guide*.

Possible values: N/A

Default value: N/A

show isis database

Displays summary or detailed information about the entries in the IS-IS LSP database.

NOTE: The Layer 3 Switch maintains separate LSP databases for Level-1 LSPs and Level-2 LSPs.

EXAMPLE:

To display summary information for all the LSPs in the Layer 3 Switch's LSP databases, enter the following command at any level of the CLI:

```
NetIron(config-isis-router)# show isis database

IS-IS Level-1 Link State Database
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime      ATT/P/OL
00e0.5200.0001.00-00* 0x00000009   0x027b        1082              0/0/1
00e0.52b5.7800.00-00  0x00000007   0x8631        1014              0/0/0
00e0.52b5.7800.01-00  0x00000006   0xcb17        1014              0/0/0

IS-IS Level-2 Link State Database
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime      ATT/P/OL
00e0.5200.0001.00-00* 0x0000000a   0xc1da        1082              0/0/1
00e0.52b5.7800.00-00  0x00000005   0xf307        115               0/0/0
```

The command in this example shows information for the LSPS in the Layer 3 Switch's Level-1 and Level-2 LSP databases. Notice that the display groups the Level-1 and Level-2 LSPs separately.

To display detailed information for all the LSPs in the Layer 3 Switch's LSP databases, enter the following command at any level of the CLI:

```
NetIron(config-isis-router)# show isis database detail

IS-IS Level-1 Link State Database
LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime      ATT/P/OL
00e0.5200.0001.00-00* 0x00000009   0x027b        1092              0/0/1
   Area Address:  20.8101
   NLPID:  cc
   IP address:  128.1.1.2
   Metric:  10    IP-Extended 128.1.1.0/24  UP bit: 0
   Metric:  10    IS 00e0.52b5.7800.01
   Metric:  10    IS-Extended 00e0.52b5.7800.01

LSPID                 LSP Seq Num  LSP Checksum  LSP Holdtime      ATT/P/OL
00e0.52b5.7800.00-00  0x00000007   0x8631        1024              0/0/0
   Area Address:  20.8101
   NLPID:  cc
   IP address:  128.1.1.1
   Metric:  10    IP-Internal 2.2.5.0          255.255.255.0
   Metric:  10    IP-Internal 2.2.6.0          255.255.255.0
   Metric:  10    IP-Internal 2.2.7.0          255.255.255.0
   Metric:  10    IP-Internal 2.2.8.0          255.255.255.0
   Metric:  10    IP-Internal 40.1.3.0         255.255.255.0
   Metric:  10    IP-Internal 128.1.1.0        255.255.255.0
   Metric:  10    IS 00e0.52b5.7800.01
   Metric:  10    IS 00e0.52b5.7800.02
```

Syntax: show isis database [detail | l1 | l2 | level1 | level2]

The **detail** parameter displays detailed information about the LSPs

The **l1** and **level1** parameters display the Level-1 LSPs only. You can use either parameter. They do the same thing.

The **l2** and **level2** parameters display the Level-2 LSPs only. You can use either parameter. They do the same thing.

To display details about Level-1 or Level-2 LSPs only, use a combination of display options, as in the following example:

```
NetIron(config-isis-router)# show isis database level2 detail
```

For information about the fields in these displays, see the "Configuring IS-IS" chapter in the *Foundry NetIron Service Provider Configuration and Management Guide*.

Possible values: N/A

Default value: N/A

show isis hostname

Displays the mappings between IS-IS system IDs and the hostnames of the devices with those IDs.

EXAMPLE:

```
NetIron(config-isis-router)# show isis hostname
Total number of entries in IS-IS Hostname Table: 1
   System ID       Hostname            * = local IS
 * bbbb.cccc.dddd  NetIron
```

Syntax: show isis hostname

The table in this example contains one mapping, for this Layer 3 Switch. The Layer 3 Switch's IS-IS system ID is "bbbb.cccc.dddd" and its hostname is "NetIron". The display contains one entry for each IS that supports name mapping.

NOTE: Name mapping is enabled by default. When name mapping is enabled, the output of the **show isis database**, **show isis neighbor**, and **show isis routes** commands uses the host name instead of the system ID. To disable mapping so that these displays use the system ID instead, see "hostname" on page 12-5.

Possible values: N/A

Default value: N/A

show isis interface

Displays information about the interfaces on which IS-IS is enabled.

EXAMPLE:

```
NetIron(config-isis-router)# show isis interface

Total number of IS-IS Interfaces: 2

Interface : 2/4     Local Circuit Number: 00000001
   Circuit Type : BCAST Circuit Mode : LEVEL-1-2
   Circuit State: UP Passive State: FALSE
   MTU : 1497
   Level-1 Metric: 10, Level-1 Priority: 64
   Level-1 Hello Interval: 10 Level-1 Hello Multiplier: 3
   Level-1 Designated IS: 00e0.52b5.7800.01-00  Level-1 DIS Changes: 4
   Level-2 Metric: 10, Priority: 64
   Level-2 Hello Interval: 10 Level-2 Hello Multiplier: 3
   Level-2 Designated IS: 00e0.52b5.7800.01-00, Level-2 DIS Changes: 5
   Next IS-IS LAN Level-1 Hello in 3 seconds
   Next IS-IS LAN Level-2 Hello in 8 seconds
   Number of active level-1 adjacencies: 1
   Number of active level-2 adjacencies: 1
   Circuit State Changes: 0 Circuit Adjacencies State Changes: 2
   Rejected Adjacencies: 0
   Circuit Authentication Fails: 0 Bad LSP: 0
   Control Messages Sent: 204 Control Messages Received: 1990
   IP Address and Subnet Mask:
     128.1.1.2           255.255.255.0
...
```

Syntax: show isis interface

For information about the fields in this display, see the "Configuring IS-IS" chapter in the *Foundry NetIron Service Provider Configuration and Management Guide*.

Possible values: N/A

Default value: N/A

show isis neighbor

Displays information about the Layer 3 Switch's IS-IS neighbors.

EXAMPLE:

```
NetIron(config-isis-router)# show isis neighbor
Total number of IS-IS Neighbors: 2
System ID      Interface   SNPA           State Holdtime Type Pri StateChgeTime
00e0.52b5.7800 Ether2/4    00e0.52b5.7843 UP    10       ISL2 64  0  :0 :16:8
00e0.52b5.7800 Ether2/4    00e0.52b5.7843 UP    10       ISL1 64  0  :0 :16:8
```

Syntax: show isis neighbor

For information about the fields in this display, see the "Configuring IS-IS" chapter in the *Foundry NetIron Service Provider Configuration and Management Guide*.

Possible values: N/A

Default value: N/A

show isis routes

Displays the routes in the Layer 3 Switch's IS-IS route table.

EXAMPLE:

```
NetIron(config-isis-router)# show isis routes

Total number of IS-IS routes: 26
Destination      Mask             Cost  Type Tag      Flags1    Flags2
50.50.15.0       255.255.255.0    11    L2   00000000 00000640  73010000
   Path: 1 Next Hop IP: 128.1.1.1         Interface: 2/4   Flags :84000003
50.50.18.0       255.255.255.0    11    L2   00000000 00000640  73010000
   Path: 1 Next Hop IP: 128.1.1.1         Interface: 2/4   Flags :84000003
50.50.21.0       255.255.255.0    11    L2   00000000 00000640  73010000
   Path: 1 Next Hop IP: 128.1.1.1         Interface: 2/4   Flags :84000003
```

Syntax: show isis routes

For information about the fields in this display, see the "Configuring IS-IS" chapter in the *Foundry NetIron Service Provider Configuration and Management Guide*.

Possible values: N/A

Default value: N/A

show isis traffic

Displays IS-IS PDU statistics.

EXAMPLE:

```
NetIron(config-isis-router)# show isis traffic
                            Message Received     Message Sent
 Level-1 Hellos             1029                 115
 Level-2 Hellos             1027                 112
 PTP Hellos                 0                    0
 Level-1 LSP                6                    3
 Level-2 LSP                6                    3
 Level-1 CSNP               0                    0
 Level-2 CSNP               0                    0
 Level-1 PSNP               107                  0
 Level-2 PSNP               107                  0
```

Syntax: show isis traffic

For information about the fields in this display, see the "Configuring IS-IS" chapter in the *Foundry NetIron Service Provider Configuration and Management Guide*.

Possible values: N/A

Default value: N/A

show link-aggregation

Displays the 802.3ad link aggregation configuration in effect on a Foundry device.

EXAMPLE:

To display the link aggregation information for a specific port, enter a command such as the following at any level of the CLI:

```
BigIron(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]
1/1         0         0        0   No    L   No   No   No   No   No   No
```

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:

```
BigIron(config-mif-1/1-1/8)# 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  Ina
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  Ina
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  Blo
1/8         1         1        1   No    L   Agg  No   No   No   Def  Exp  Ope
3/1         1         1       32   Yes   L   Agg  No   No   No   No   No   Dwn
3/2         1         1       32   Yes   L   Agg  No   No   No   No   No   Dwn
3/3         1         1       32   Yes   L   Agg  Syn  No   No   Def  Exp  Ope
3/4         1         1       32   Yes   L   Agg  Syn  No   No   Def  Exp  Ope
3/5         1         1       33   Yes   L   Agg  Syn  No   No   Def  Exp  Ope
3/6         1         1       33   Yes   L   Agg  Syn  No   No   Def  Exp  Ope
3/7         1         1       33   Yes   L   Agg  Syn  No   No   Def  Exp  Ope
3/8         1         1       33   Yes   L   Agg  Syn  No   No   Def  Exp  Ope
3/9         1         1       34   Yes   L   Agg  Syn  No   No   Def  Exp  Ope
3/10        1         1       34   Yes   L   Agg  Syn  No   No   Def  Exp  Ope
3/11        1         1       34   Yes   L   Agg  Syn  No   No   Def  Exp  Ope
3/12        1         1       34   Yes   L   Agg  Syn  No   No   Def  Exp  Ope
```

For information about the fields in this display, see the "Configuring Trunk Groups and Dynamic Link Aggregation" chapter in the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Syntax: show link-aggregation [ethernet <portnum>]

Possible values: N/A

Default value: N/A

show link-keepalive

Displays Uni-Directional Link Detection (UDLD) information.

EXAMPLE:

To display UDLD information for all ports, enter the following command:

```
BigIron(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
```

To display detailed UDLD information for a specific port, enter a command such as the following:

```
BigIron(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
```

Syntax: show link-keepalive [ethernet <portnum>]

Possible values: N/A

Default value: N/A

show logging

Displays the SNMP event log.

NOTE: This section describes the command syntax. For configuration information and a list of the Syslog messages, see the "Syslog Messages" appendix in the *Foundry Switch and Router Installation and Basic Configuration Guide*.

EXAMPLE:

To display the Syslog messages in the buffer, enter the following command at any level of the CLI:

```
BigIron> 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 ethernet4, 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
```


EXAMPLE:

This example show some common Syslog messages.

```
BigIron# show logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
    Buffer logging: level ACDMEINW, 7 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):
00d05h44m28s:info:Interface e3/11, state up
00d05h44m28s:info:Bridge topology change, vlan 1, interface 3/11, changed state
to forwarding
00d04h45m49s:info:Interface e3/11, state down
00d04h45m20s:info:Interface e3/11, state up
00d04h45m20s:info:Bridge topology change, vlan 1, interface 3/11, changed state
to forwarding
00d01h45m13s:info:Interface e3/11, state down
00d00h01m00s:info:Interface e3/11, state up
00d00h00m05s:info:Bridge topology change, vlan 1, interface 3/11, changed state
to forwarding
00d00h00m00s:info:Warm start
```

EXAMPLE:

This example shows log entries for authentication failures. If someone enters an invalid community string when attempting to access the SNMP server on the Foundry device, the device generates a trap in the device's syslog buffer. (If you have configured the device to use a third-party Syslog server, the device also sends a log entry to the server.)

Here is an example of a log that contains SNMP authentication traps. In this example, someone attempted to access the Foundry device three times using invalid SNMP community strings. The unsuccessful attempts indicate either an authorized user who is also a poor typist, or an unauthorized user who is attempting to access the device.

```
BigIron(config)# show log
Syslog logging: enabled (0 messages dropped, 0 flushes, 1 overruns)
Buffer logging: level ACDMEINW, 50 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):
00d01h45m13s:info:SNMP Authentication failure, intruder IP: 207.95.6.55
00d00h01m00s:info:SNMP Authentication failure, intruder IP: 207.95.6.55
00d00h00m05s:info:SNMP Authentication failure, intruder IP: 207.95.6.55
```

EXAMPLE:

This example shows a log entry for an IP address conflict between the Foundry device and another device on the network.

In addition to placing an entry in the log, the software sends a log message to the Syslog server, if you have configured one, and sends a message to each open CLI session.

```
BigIron(config)# show log
Syslog logging: enabled (0 messages dropped, 0 flushes, 1 overruns)
Buffer logging: level ACDMEINW, 50 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):
00d01h45m13s:warning:Duplicate IP address 209.157.23.188 detected,sent from MAC
address 00e0.5201.3bc9 coming from port 7/7
```

EXAMPLE:

Here are some examples of log entries for packets denied by Access Control Lists (ACLs).

```
BigIron(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)

00d07h03m30s:warning:list 101 denied tcp 209.157.22.26(0)(Ethernet 4/18
0010.5a1f.77ed) -> 198.99.4.69(http), 1 event(s)

00d06h58m30s:warning:list 101 denied tcp 209.157.22.198(0)(Ethernet 4/18
0010.5a1f.77ed) -> 198.99.4.69(http), 1 event(s)
```

EXAMPLE:

Here are some examples of log entries for BGP4. The first log entry written to the log (the entry at the bottom) occurs when you try to enable BGP4 on a device that does not have enough free memory to run the protocol. The other messages occur when a BGP4 neighbor's state changes. In this case, the state changes occur when the neighbor session starts and when it ends.

The messages in this example show state changes that indicate the neighbor session is coming up (ESTABLISHED) and going down (IDLE).

For an explanation of the BGP4 neighbor states, see the "Configuring BGP4" chapter of the *Foundry Enterprise Configuration and Management Guide*.

```
00d01h31m49s:info:BGP Peer 192.168.12.3 UP (ESTABLISHED)
00d01h31m38s:info:BGP Peer 192.168.12.3 DOWN (IDLE)
00d00h06m01s:info:BGP Peer 192.168.11.2 UP (ESTABLISHED)
00d00h00m00s:info:Warm start
00d00h00m00s:debug:BGP4: Not enough memory available to run BGP4
```


EXAMPLE:

Here are some examples of log messages for CLI access.

```
BigIron(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:

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
```

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 in to 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.

Syntax: show logging

Possible values: N/A

Default value: N/A

show mac vpls

Displays the contents of the device’s VPLS MAC database. The VPLS MAC database stores entries associating MAC addresses with VC LSPs.

EXAMPLE:

```
NetIron# show mac vpls
Total VPLS mac entries in the table: 2 (Local: 1, Remote: 1)
VplsId:MacAddr      L/R Port Vlan/Peer      Age DMA Valid Flags   DMA:CamIdx
1:1000.0000.0001    L   4/2  10             0   00003000-00000000 12:14339
1:2000.0000.0001    R   3/1  2.2.2.2        0   00003000-00000000 12:14337
```

Syntax: show mac vpls

Syntax: show mac vpls <mac-address> <vpls-id>

Syntax: show mac vpls id <vpls-id>

Syntax: show mac vpls id <vpls-id> <starting-entry> <number-of-entries>

Possible values: The <vpls-id> parameter specifies the VPLS ID for which database entries are displayed. The <starting-entry> parameter specifies the point in the database from which the entries are displayed. Entering 0 as the <starting-entry> causes entries to be displayed from the start; entering 200 causes the first 200 entries to be skipped; and so on. The <number-of-entries> parameter specifies the number of database entries to be displayed from the <starting-entry>.

Default value: N/A

show mac-address

Displays the MAC addresses in the MAC address table. This table contains the MAC addresses learned from other devices or added using the **static-mac-address** command. See “static-mac-address” on page 6-130.

This table does not contain the MAC addresses of the Foundry device’s own ports.

EXAMPLE:

To display all the MAC addresses in the MAC address table, enter the following:

```
FastIron(config)# show mac-address
Total entries from all ports = 75
       MAC     Port   Age CamF CIDX0 CIDX1 CIDX2 CIDX3 CIDX4 CIDX5
0000.0300.0000  10  17293 00H     0     0     0     0     0     0
0060.089f.8086   1     12 0bH    23    15     0     6     0     0
0060.9709.914b  16   2130 00H     0     0     0     0     0     0
00a0.249a.0163  16    130 00H     0     0     0     0     0     0
0060.979d.41a5  11    475 00H     0     0     0     0     0     0
00a0.24c5.01d1  11      0 0cH     0     0    20    14     0     0
0060.979d.41df  11    570 00H     0     0     0     0     0     0
0060.9759.4226  16    240 00H     0     0     0     0     0     0
0060.9759.4235  16    130 00H     0     0     0     0     0     0
0800.208f.725b   2    135 00H     0     0     0     0     0     0
0060.9759.4264  16      0 0aH     0    14     0    21     0     0
00a0.24c5.02a1  16     15 09H     5     0     0    33     0     0
0000.c02c.a2bf   7     11 03H    27     5     0     0     0     0
00a0.24c5.02f8   4    135 00H     0     0     0     0     0     0
00a0.24c5.02fc   6      0 06H     0     8    31     0     0     0
0800.207e.c312   2      2 0dH    25     0    24    13     0     0
0800.208f.5331   2    135 00H     0     0     0     0     0     0
00e0.5200.0385  10   5160 00H     0     0     0     0     0     0
00e0.5200.0388  10  35420 00H     0     0     0     0     0     0
0000.f820.a3ad   3  11239 00H     0     0     0     0     0     0
0000.c06f.d3be   3    135 00H     0     0     0     0     0     0
--More--, next page: Space/Return key, quit: Control-c
```

NOTE: The information displayed in columns with headings, **CamF**, and **CIDX0** through **CIDX5**, is not relevant for day-to-day management of the device. The information is used by engineering and technical support staff for debug purposes.

Syntax: show mac-address [ethernet <portnum> | session | statistics | <mac-addr>]

The **ethernet** <portnum> option displays the MAC address(es) on the specified port.

The **session** parameter displays the MAC session table.

The **statistics** option displays statistics. See “show mac-address statistics” on page 43-97.

The <mac-addr> option displays information about the specified MAC address. Specify the MAC address in the following format: abcd.egh.1234. (Separate each four bytes with a period.)

Possible values: see above

Default value: N/A

show mac-address statistics

Displays the total number of MAC addresses currently active on a Foundry device. This command serves as a numerical summary of the detailed summary provided by the command **show mac-address**.

For each port, the number of learned MAC addresses is displayed.

EXAMPLE:

```
BigIron(config)# show mac-address statistics
Total entries  = 41
Port     1     2     3     4     5     6     7     8     9
         0     6    11     1     1     1     2     1     1
Port     10     11      12    13    14    15    16
         0      3       1      3     1     1     8
```

Syntax: show mac-address statistics

Possible values: N/A

Default value: N/A

show media

Shows the type of ports active on a Chassis device.

EXAMPLE:

```
BigIron(config)# show media
1/1:SX 1/2:SX 1/3:SX 1/4:SX
2/1:SX 2/2:SX 2/3:SX 2/4:SX 2/5:SX 2/6:SX 2/7:SX 2/8:SX
3/1:SX 3/2:SX 3/3:SX 3/4:SX 3/5:SX 3/6:SX 3/7:SX 3/8:SX
4/1:SX 4/2:SX 4/3:SX 4/4:SX 4/5:SX 4/6:SX 4/7:SX 4/8:SX

6/1:SX 6/2:SX 6/3:SX 6/4:SX 6/5:SX 6/6:SX 6/7:SX 6/8:SX
7/1:SX 7/2:SX 7/3:SX 7/4:SX 7/5:SX 7/6:SX 7/7:SX 7/8:SX
8/1:SX 8/2:SX 8/3:SX 8/4:SX 8/5:SX 8/6:SX 8/7:SX 8/8:SX
```

Syntax: show media

Possible values: N/A

Default value: N/A

show memory

Displays memory utilization statistics for protocols that use dynamic memory allocation.

EXAMPLE:

```
BigIron# show memory
Total DRAM:                    134160384
  Dynamic memory size:    122538368
    BGP memory usage:          198400
    OSPF memory usage:        443980
    Free memory size:          64362156
```

Syntax: show memory

The command lists the total amount of dynamic memory the system has, the amount currently in use by BGP4 and OSPF, and the amount that is still free for use.

The rows for BGP4 and OSPF information are displayed only when those protocols are enabled. In this example, BGP4 and OSPF are both enabled and are currently using dynamic memory.

Possible values: N/A

Default value: N/A

show memory tcp

Displays the amount of used and free memory for each of the four internal TCP buffers.

EXAMPLE:

```
BigIron# show memory tcp
TCP MEMORY USAGE
    TCB usage: total=65025, free=63750
    TCP QUEUE BUFFER usage: total=28616, free=18032
    TCP SEND BUFFER usage: total=382500, free=121500
    TCP RECEIVE BUFFER usage: total=382500, free=360000
    TCP OUT OF SEQUENCE BUFFER usage: total=19900, free=19900
```

Syntax: show memory tcp

For each internal buffer, the amount of used and free memory is shown in bytes.

Possible values: N/A

Default value: N/A

show metro

Displays information for a metro ring.

EXAMPLE:

```
BigIron(config)# show metro

Metro Ring 2
=============
Ring       State      Ring       Master     Topo       Hello      Prefwing
id                    role       vlan       group      time(ms)   time(ms)
2          enabled    master     2          not conf   100        300

Ring interfaces      Interface role       Forwarding state     Active interface
ethernet 1           primary              disabled             none
ethernet 2           secondary            forwarding           ethernet 2

RHPs sent            RHPs rcvd            TC RHPs rcvd         State changes
3                    0                    0                    4
```

Syntax: show metro [<ring-id>]

For information about the fields in this display, see the "Displaying Ring Information" section in the "Configuring Metro Features" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

EXAMPLE:

```
BigIron(config)# show metro 2 diag

Metro Ring 2 - 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

For information about the fields in this display, see the "Displaying MRP Diagnostic Information" section in the "Configuring Metro Features" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Possible values: N/A

Default value: N/A

show module

Shows the types of modules installed on a Chassis device.

EXAMPLE:

Here is an example of the command's display output on a BigIron 8000 Layer 3 Switch.

```
BigIron# show module
    Module                           Status    Ports Starting MAC
S1: B8GM Fiber Management Module     OK          8   00e0.52f0.5a00
S2: B24E Copper Switch Module        OK         24   00e0.52f0.5a20
S3: B24E Copper Switch Module        OK         24   00e0.52f0.5a40
S4: B24E Copper Switch Module        OK         24   00e0.52f0.5a60
S5: B8G Fiber Switch Module          OK          8   00e0.52f0.5a00
S6: B24E Copper Switch Module        OK         24   00e0.52f0.5aa0
S7: B8G Fiber Switch Module          OK          8   00e0.52f0.5a00
S8: B8G Fiber Switch Module          OK          8   00e0.52f0.5a00
```

EXAMPLE:

Here is an example of the command's display output on a FastIron II Layer 2 Switch.

```
FastIron# show module
    Module                           Status    Ports Starting MAC
S1: FI4GM Fiber Management Module    OK          4   00e0.5283.8a00
S2: FI24E Fiber Switch Module        OK         24   00e0.5283.8a20
S3: FI24E Fiber Switch Module        OK         24   00e0.5283.8a40
S4: FI24E Fiber Switch Module        OK         24   00e0.5283.8a60
```

Possible values: N/A

Default value: N/A

show monitor

Displays the current port mirroring and monitoring configuration.

EXAMPLE:

To display the current mirroring and monitoring configuration, enter the following command at any level of the CLI:

```
BigIron(config)# show monitor
Mirror Interface:        ethernet 4/1
Monitored Interfaces:
        Both          Input     Output
        ------------------------------------------------
         ethernet 4/3
```

Syntax: show monitor

In this example, port 4/1 is the mirror interface, to which the software copies (“mirrors”) the traffic on port 4/3. In this case, both directions of traffic on the monitored port are mirrored to port 4/1.

If only the incoming traffic is mirrored, the monitored interface is listed under Input. If only the outbound traffic is mirrored, the monitored interface is listed under Output.

Possible values: N/A

Default value: N/A

show mpls interface

Displays information about the interfaces on the device that have been enabled for MPLS.

EXAMPLE:

```
NetIron# show mpls interface
e1/1
  Maximum BW: 1000000 kbps, maximum reservable BW: 1000000 kbps
  Admin group: 0x0000003a ( 1 3 4 5)
  Reservable BW [priority] kbps:
    [0] 1000000    [1] 1000000    [2] 1000000    [3] 1000000
    [4] 1000000    [5] 1000000    [6] 1000000    [7] 1000000
  Last sent reservable BW [priority] kbps:
    [0] 0    [1] 0    [2] 0    [3] 0
    [4] 0    [5] 0    [6] 0    [7] 0
```

Syntax: show mpls interface

Possible values: N/A

Default value: N/A

show mpls ldp

Displays the LDP version number, the LSR’s LDP identifier and loopback number, and the LDP hello interval and hold time.

EXAMPLE:

```
NetIron(config)# show mpls ldp
Label Distribution Protocol version 1
  LSR ID: 2.2.2.2, using Loopback 1 (deleting it will stop LDP)
  Hello interval: Link 5 sec, Targeted 15 sec
  Hold time value sent in Hellos: Link 15 sec, Targeted 45 sec
```

Syntax: show mpls ldp

Possible values: N/A

Default value: N/A

show mpls ldp database

Displays the contents of the LSR's LDP Label Information Base. This database contains all the labels it has learned from each of its LSR peers, as well as all of the labels it has sent to its LDP peers.

EXAMPLE:

```
NetIron# show mpls ldp database
Session 13.13.13.13:0 - 11.11.11.11:0
 Downstream label database:
   Label     Prefix              State
   3         11.11.11.11/32      Installed
   1066      13.13.13.13/32      Retained
   1227      12.12.12.12/32      Retained
   1228      14.14.14.14/32      Retained
 Upstream label database:
   Label     Prefix
   3         13.13.13.13/32
   1024      14.14.14.14/32
   1027      12.12.12.12/32
   1028      11.11.11.11/32
```

Syntax: show mpls ldp database

Possible values: N/A

Default value: N/A

show mpls ldp interface

Displays information about the LDP enabled interfaces on the LSR.

EXAMPLE:

```
NetIron# show mpls ldp interface
                    Label-space    Nbr            Hello          Next
Interface           ID             Count          Interval       Hello
p4/1                0              1              5              3
(targeted)          1              0              15             3
```

Syntax: show mpls ldp interface

Possible values: N/A

Default value: N/A

show mpls ldp neighbor

Displays information about the connection between this LSR and its LDP-enabled neighbors.

EXAMPLE:

```
NetIron# show mpls ldp neighbor
Nbr Transport       Interface      Nbr LDP ID          Max Hold  Time Left
1.1.1.1             p4/1           1.1.1.1:0           15        14
5.5.5.5             p3/2           5.5.5.5:0           15        11
4.4.4.4             (targeted)     4.4.4.4:0           15        13
```

Syntax: show mpls ldp neighbor

Possible values: N/A

Default value: N/A

show mpls ldp path

Displays information about active LDP-created LSPs for which this device is an ingress, transit or egress LSR.

EXAMPLE:

```
NetIron(config)# show mpls ldp path
Upstr-session(label)     Downstr-session(label, intf)    Destination route
33.3.3.3:0(3)            (egress)                        11.1.1.1/32
22.2.2.2:0(3)            (egress)                        11.1.1.1/32
33.3.3.3:0(1024)         22.2.2.2:0(3, e2/10)            22.2.2.2/32
22.2.2.2:0(1024)         22.2.2.2:0(3, e2/10)            22.2.2.2/32
(ingress)                22.2.2.2:0(3, e2/10)            22.2.2.2/32
33.3.3.3:0(1026)         33.3.3.3:0(3, e2/20)            33.3.3.3/32
22.2.2.2:0(1026)         33.3.3.3:0(3, e2/20)            33.3.3.3/32
(ingress)                33.3.3.3:0(3, e2/20)            33.3.3.3/3
```

Syntax: show mpls ldp path

Possible values: N/A

Default value: N/A

show mpls ldp session

Displays information about LDP sessions between the Foundry device and LDP peers.

EXAMPLE:

```
NetIron# show mpls ldp session
Peer LDP ID         State              Role      Hold time
22.2.2.2:0          Operational        passive   34

Peer LDP ID             State          Targeted  My Role   Max Hold  Time Left
1.1.1.1:0               Operational    No        Active    36        26
5.5.5.5:0               Operational    No        Passive   36        26
4.4.4.4:0               Operational    Yes       Passive   36        26
```

Syntax: show mpls ldp session [detail]

Possible values: N/A

Default value: N/A

show mpls ldp target-peer

Displays information about the state of the LDP connection between the Foundry device and VLL peers.

EXAMPLE:

```
NetIron# show mpls ldp target-peer
Peer-addr      State
192.168.2.100  Initialized
```

Syntax: show mpls ldp target-peer

Possible values: N/A

Default value: N/A

show mpls ldp tunnel

Displays information about LDP-created LSPs for which this device is the ingress LER.

EXAMPLE:

```
NetIron# show mpls ldp tunnel
                Oper      Tunnel    Outbound
To              State     Intf      Intf
22.2.2.2        UP        tnl0      p3/1
33.3.3.3        UP        tnl1      p3/2
```

Syntax: show mpls ldp tunnel

Possible values: N/A

Default value: N/A

show mpls lsp

Displays status information about signalled LSPs for which the device is the ingress LER.

EXAMPLE:

```
NetIron# show mpls lsp
*: The LSP is taking a Secondary Path
                                Admin Oper  Tunnel   Up/Dn Retry Active
Name            To              State State Intf     Times No.   Path
t1              3.3.3.3         UP    UP*   tnl1     1     5     v2
```

Syntax: show mpls lsp [brief]

NOTE: The **show mpls lsp brief** command displays the same information as the **show mpls lsp** command.

Possible values: N/A

Default value: N/A

show mpls lsp detail

Displays detailed information about the status of the LSPs for which the device is the ingress LER:

EXAMPLE:

```
NetIron# show mpls lsp detail
LSP to-1, to 11.1.1.1
  From: 1.2.3.4, admin: UP, status: DOWN (Path not sent)
  Times primary LSP goes up since enabled: 0
  Metric: 0, number of installed aliases: 0
  Maximum retries: 0, no. of retries: 0
  Pri. path: 1, active: yes
  Path specific attributes:
   Tunnel interface: tnl0, outbound interface: p4/1
   Setup priority: 7, hold priority: 0
   Max rate: 0 kbps, mean rate: 0 kbps, max burst: 0 bytes
   Constraint-based routing enabled: yes
   Tie breaking: random, hop limit: 0
   Explicit path hop counts: 1
    20.1.1.1 (S)
   Recorded routes:
    20.1.1.1
```

Syntax: show mpls lsp detail | <name>

Possible values: The <name> parameter displays detailed information about a specific LSP.

Default value: N/A

show mpls path

Displays information about the paths configured on the device.

EXAMPLE:

```
NetIron# show mpls path
Path Name           Address             Strict/loose
1                   1.2.3.4             Strict
                    1.2.3.5             Strict
                    1.2.3.11            Loose
                    1.2.3.12            Strict
                    1.2.3.13            Loose
2                   (n/a)               (n/a)
3                   1.2.3.13            Loose
                    1.2.3.11            Loose
4                   3.4.5.1             Strict
                    4.5.6.7             Loose
                    3.4.5.2             Strict
```

Syntax: show mpls path [<path-name>]

Possible values: Name of a configured MPLS path

Default value: By default, the command displays all MPLS paths.

show mpls route

Displays the contents of the MPLS routing table.

EXAMPLE:

```
NetIron# show mpls route
Total number of MPLS tunnel routes: 2
Start index: 1
      Destination       NetMask           Gateway           Port   Cost
      20.1.1.2          255.255.255.255   20.1.1.2          tnl0      1
      30.1.1.2          255.255.255.255   20.1.1.2          tnl0      1
```

Syntax: show mpls route

Possible values: N/A

Default value: N/A

show mpls rsvp

Displays the RSVP version number and the configured refresh interval and refresh multiple.

EXAMPLE:

```
NetIron# show mpls rsvp
Resource ReSerVation Protocol, version 1. rfc2205
RSVP protocol        = Enabled
R (refresh interval) = 30 seconds
K (refresh multiple) = 3
```

Syntax: show mpls rsvp

Possible values: N/A

Default value: N/A

show mpls rsvp interface

Displays the status of RSVP on devices where it is enabled.

EXAMPLE:

```
NetIron# show mpls rsvp interface
Interface State
     p2/1 Up
     p2/2 Dn
     p4/1 Dn
     p4/2 Dn
```

Syntax: show mpls rsvp interface [brief]

NOTE: The **show mpls rsvp interface brief** command displays the same information as the **show mpls rsvp interface** command.

Possible values: N/A

Default value: N/A

show mpls rsvp interface detail

Displays detailed information about RSVP-enabled interfaces.

EXAMPLE:

```
NetIron# show mpls rsvp interface detail
 Interface State
      P2/1 Up
   PacketType            Total                  Since last clear
                    Sent      Received        Sent      Received
   Path             8288             0           1             0
   PathErr             0             0           0             0
   PathTear            0             0           0             0
   Resv                0          3372           0             0
   ResvErr          1685             0           0             0
   ResvTear            0             0           0             0
```

Syntax: show mpls rsvp interface detail

Possible values: N/A

Default value: N/A

show mpls rsvp session

Displays RSVP session information.

EXAMPLE:

```
NetIron# show mpls rsvp session
To               From             State Style Labelin Labelout LSPname
Ingress RSVP:    1 session(s)
20.1.1.2         10.1.1.1         Up    FF    -       1025     t1


Transit RSVP:    0 session(s)

Egress RSVP:     1 session(s)
10.1.1.1         20.1.1.2         Up    FF    1024    -        t2
```

Syntax: show mpls rsvp session [brief]

NOTE: The **show mpls rsvp session brief** command displays the same information as the **show mpls rsvp session** command.

Possible values: N/A

Default value: N/A

show mpls rsvp session detail

Displays detailed information about RSVP sessions.

EXAMPLE:

```
NetIron# show mpls rsvp session detail
Ingress RSVP:    1 session(s)
To               From             State Style Labelin Labelout LSPname
20.1.1.2         10.1.1.1         Up    FF    -       1025     t1
  Time left in seconds (PATH refresh: 10  RESV refresh: 6)
  Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 65535
  PATH sentto:  10.1.1.2        (p2/1          )
  RESV rcvfrom: 10.1.1.2        (p2/1          )


Transit RSVP:    0 session(s)

Egress RSVP:     1 session(s)
To               From             State Style Labelin Labelout LSPname
10.1.1.1         20.1.1.2         Up    FF    1024    -        t2
  Time left in seconds (PATH refresh: 12  RESV refresh: 36)
  Tspec: peak 0 kbps rate 0 kbps size 0 bytes m 20 M 4470
  PATH rcvfrom: 10.1.1.2        (p2/1          )
```

Syntax: show mpls rsvp session detail

Possible values: N/A

Default value: N/A

show mpls rsvp statistics

Displays RSVP statistics. The Foundry device constantly gathers RSVP statistics. RSVP statistics are collected from the time RSVP is enabled, as well as from the last time the RSVP statistics counters were cleared with the **clear mpls rsvp statistics** command.

EXAMPLE:

```
NetIron# show mpls rsvp statistics
                      Total                  Since last clear
 PacketType        Sent      Received        Sent      Received
  Path             4         4               4         4
  Resv             4         4               4         4
  PathErr          0         0               0         0
  ResvErr          0         0               0         0
  PathTear         0         0               0         0
  ResvTear         0         0               0         0
  ResvConf         0         0               0         0

 Errors                 Total                Since last clear
  Rcv pkt bad length    0                    0
  Rcv pkt unknown type  0                    0
  Rcv pkt bad version   0                    0
  Rcv pkt bad cksum     0                    0
  Memory alloc fail     0                    0
```

Syntax: show mpls rsvp statistics

Possible values: N/A

Default value: N/A

show mpls statistics

Displays MPLS statistics. Statistics about MPLS packets on the Foundry device are gathered on a per-interface basis, rather than on a per-LSP basis.

EXAMPLE:

```
NetIron# show mpls statistics

PORT p2/1 packet stats:

          Tunnel entry tx =      0 Tunnel entry drop =      0
           Tunnel exit rx =      0  Tunnel exit drop =      0
               XC inbound =      0   XC inbound drop =      0
              XC outbound =      0  XC outbound drop =      0
                   VLL tx =           0
                   VLL rx =           0

PORT p2/2 packet stats:

          Tunnel entry tx =      0 Tunnel entry drop =      0
           Tunnel exit rx =      0  Tunnel exit drop =      0
               XC inbound =      0   XC inbound drop =      0
              XC outbound =      0  XC outbound drop =      0
                   VLL tx =           0
                   VLL rx =           0
```

Syntax: show mpls statistics

Possible values: N/A

Default value: N/A

show mpls static-lsp

Displays status information about static LSPs of which the Foundry device is a component.

EXAMPLE:

```
NetIron# show mpls static-lsp
                        Tunnel In     In     Out    Out
Name            State   Intf   Port   Label  Port   Label  To
1               UP      tnl2   --     --     p2/1   200    2.2.2.2
2               UP      --     p1/1   100    p1/2   500
```

Syntax: show mpls static-lsp [brief]

NOTE: The **show mpls static-lsp brief** command displays the same information as the **show mpls static-lsp** command.

Possible values: N/A

Default value: N/A

show mpls static-lsp detail

Displays detailed information about static LSPs of which the Foundry device is a component.

EXAMPLE:

```
NetIron# show mpls static-lsp detail
Static LSP 1
  Admin/oper: UP
  Placement: INGRESS, To: 2.2.2.2
  Out-segment: port p2/1, label 200
  Metric: 1, number of installed aliases: 0
  Max rate: 1000 kbps, mean rate: 100 kbps, max burst: 0
Static LSP 2
  Admin/oper: UP
  Placement: TRANSIT
  In-segment: port p1/1, label 100
  Out-segment: port p1/2, label 500
  Metric: 1, number of installed aliases: 0
  Max rate: 250 kbps, mean rate: 250 kbps, max burst: 0
```

Syntax: show mpls static-lsp detail | <name>

Possible values: The <name> parameter displays detailed information about a specific static LSP.

Default value: N/A

show mpls summary

Displays a summary of MPLS information, including the number of configured paths, signalled LSPs, and static LSPs for which this device is the ingress LER.

EXAMPLE:

```
NetIron# show mpls summary

Path:
        Paths configured    =     0

Signaled LSPs:
        LSPs configured     =     1
        LSPs enabled        =     1
        LSPs operational    =     0

Static LSPs:
        LSPs configured     =     0
        LSPs enabled/oper.  =     0
```

Syntax: show mpls summary

Possible values: N/A

Default value: N/A

show mpls ted data

Displays the contents of an LSR's Traffic Engineering Database (TED).

EXAMPLE:

```
NetIron# show mpls ted data
AreaId: 0
  NodeID: 2.2.2.2, Type: Router
    Type: M/A, To: 10.1.1.3, Local: 10.1.1.2, Remote: 0.0.0.0
  NodeID: 3.3.3.3, Type: Router
    Type: P2P, To: 6.6.6.6, Local: 40.1.1.1, Remote: 40.1.1.2
    Type: M/A, To: 10.1.1.3, Local: 10.1.1.3, Remote: 0.0.0.0
    Type: M/A, To: 20.1.1.2, Local: 20.1.1.1, Remote: 0.0.0.0
  NodeID: 10.1.1.3, Type: Network
    Type: M/A, To: 1.1.1.1, Local: 0.0.0.0, Remote: 0.0.0.0
    Type: M/A, To: 2.2.2.2, Local: 0.0.0.0, Remote: 0.0.0.0
    Type: M/A, To: 3.3.3.3, Local: 0.0.0.0, Remote: 0.0.0.0
  NodeID: 30.1.1.2, Type: Network
    Type: M/A, To: 1.1.1.1, Local: 0.0.0.0, Remote: 0.0.0.0
    Type: M/A, To: 6.6.6.6, Local: 0.0.0.0, Remote: 0.0.0.0
```

Syntax: show mpls ted data

Possible values: N/A

Default value: N/A

show mpls ted data detail

Displays detailed information about each node in the TED.

EXAMPLE:

```
NetIron# show mpls ted data detail
AreaId: 0
  NodeID: 2.2.2.2, Type: Router
    Type: M/A, To: 10.1.1.3, Local: 10.1.1.2, Remote: 0.0.0.0
      Color: 0x00000007
      Metric: 1
      Max BW: 155000 kbps
      Reservable BW: 155000 kbps
      Available BW [priority] kbps:
        [0] 155000       [1] 155000       [2] 155000       [3] 155000
        [4] 155000       [5] 155000       [6] 155000       [7] 155000
  NodeID: 1.1.1.1, Type: Router
    Type: M/A, To: 10.1.1.3, Local: 10.1.1.1, Remote: 0.0.0.0
      Color: 0x00000007
      Metric: 1
      Max BW: 155000 kbps
      Reservable BW: 155000 kbps
      Available BW [priority] kbps:
        [0] 155000       [1] 155000       [2] 155000       [3] 155000
        [4] 155000       [5] 155000       [6] 155000       [7] 155000
    Type: M/A, To: 30.1.1.2, Local: 30.1.1.1, Remote: 0.0.0.0
      Color: 0x00000007
      Metric: 1
      Max BW: 155000 kbps
      Reservable BW: 155000 kbps
      Available BW [priority] kbps:
        [0] 155000       [1] 155000       [2] 155000       [3] 155000
        [4] 155000       [5] 155000       [6] 155000       [7] 155000
```


Syntax: show mpls ted data detail

Possible values: N/A

Default value: N/A

show mpls vll brief

Displays information about MPLS VLLs.

EXAMPLE:

```
NetIron# show mpls vll brief
Name              VC-ID   Vll-peer       End-point             State Tunnel-LSP
vll-1             1       2.2.2.2        untag e 2/1           UP    lsp_1
vll-2             2       3.3.3.3        tag vlan 200  e 2/2   DOWN  --
vll-3             3       --             undefined             DOWN  --
foundry-sj-to-sf  40000   192.168.2.100  untagged e 2/1        UP    lsp_1
```

Syntax: show mpls vll brief

Possible values: N/A

Default value: N/A

show mpls vll detail

Displays detailed information about MPLS VLLs.

EXAMPLE:

```
NetIron# show mpls vll detail
VLL foundry-sj-to-sf VC-ID 40000
  State: UP
  Vll-peer:         2.2.2.2         End-point:        untagged  e 2/1
  Local label:      --              Remote label:     --
  Local group-id:   0               Remote group-id:  1
  COS:              --              Tunnel LSP:       lsp_1 (tnl2)
VLL vll-2 VC-ID 2
  State: DOWN - no tunnel LSP to vll-peer
  Vll-peer:         3.3.3.3         End-point:        tagged  vlan 200   e 2/2
  Local label:      --              Remote label:     --
  Local group-id:   0               Remote group-id:  --
  COS:              5               Tunnel LSP:       --
VLL vll-3 VC-ID 3
  State: DOWN - configuration incomplete
  Vll-peer:         --              End-point:        undefined
  Local label:      --              Remote label:     --
  Local group-id:   0               Remote group-id:  --
  COS:              7               Tunnel LSP:       --
VLL foundry-sj-to-sf VC-ID 40000
  State: UP
  Vll-peer:         192.168.2.100   End-point:        untagged e 2/1
  Local label:      --              Remote label:     --
  Local group-id:   0               Remote group-id:  --
  COS:              5               Tunnel LSP:       lsp_1 (tnl2)
```

Syntax: show mpls vll detail | <vll-name>

Possible values: Configured VLL name

Default value: N/A

show mpls vpls

Displays information about VPLS instances configured on the device.

EXAMPLE:

```
NetIron# show mpls vpls
                      Num     Num     Ports   Num     Peers   Num
Name          Id      Vlans   Ports   Up      Peers   Up      VC-label
test          100     2       3       0       2       0       32
```

Syntax: show mpls vpls

Possible values: N/A

Default value: N/A

show mpls vpls detail

Displays detailed information about VPLS instances configured on the device.

EXAMPLE:

```
NetIron# show mpls vpls detail
VPLS test, Id 100, Max mac entries: 65536
 Total vlans: 2, Tagged ports: 1 (0 Up), Untagged ports 2 (0 Up)
  Vlan 100
   Tagged: ethe 1/5
   Untagged: ethe 1/6
  Vlan 200
   Tagged: ethe 1/5
   Untagged: ethe 1/7
 VC labels allocated: 32 (983040-983071)
 Total VPLS peers: 2 (1 Operational)

 Peer address: 2.2.2.2, State: Operational, Uptime: 2 sec
  Tnnl: tnl0(1024), LDP session: Up, Local VC lbl: 983040, Remote VC lbl: 983040
 Peer address: 3.3.3.3, State: Wait for functional local ports
  Tnnl: N/A, LDP session: Down, Local VC lbl: 983041, Remote VC lbl: N/A
```

Syntax: show mpls vpls detail

Possible values: N/A

Default value: N/A

show mpls vpls down

Displays information about VPLS instances that are not operational.

EXAMPLE:

```
NetIron# show mpls vpls down
The following VPLS'es are not completely operational:
                      Num     Num     Ports   Num     Peers   Num
Name          Id      Vlans   Ports   Up      Peers   Up      VC-label
V1            1       1       1       0       2       0       32
```

Syntax: show mpls vpls down

Possible values: N/A

Default value: N/A

show mpls vpls summary

Displays a summary of VPLS information, including the number of VPLS instances, number of VPLS peers, label range size, and maximum size of the VPLS MAC database.

EXAMPLE:

```
NetIron# show mpls vpls summary
Virtual Private LAN Service summary:
  Total VPLS configured: 1, maximum number of VPLS allowed: 2048
  Total VPLS peers configured: 2, total peers operational: 0
  VC label range size: 32
  Maximum VPLS mac entries allowed: 65536
```

Syntax: show mpls vpls summary

Possible values: N/A

Default value: N/A

show pcmcia

Displays information for a 3Com Megahertz 10/100 LAN PC Card (model 3CCFE574BT) inserted in a PCMCIA slot.

EXAMPLE:

```
BigIron(config)# show pcmcia all
3Com Megahertz 574B lan card present at slot 1
   Hardware is 100mbitEthernet,    Mac address is 0001.03aa.2902
   Speed is 10mb, Duplex is hdx, Flow_control is off
   Member of L2 Vlan ID 1, port is untagged, not a member of any trunk.

PCMCIA card is not present at slot 2
```

Syntax: show pcmcia all | slot1 | slot2

Possible values: N/A

Default value: N/A

show port security

Displays the port security settings for an individual port or for all the ports on a specified module.

EXAMPLE:

```
BigIron# show port security e 7/11
Port  Security Violation Shutdown-Time Age-Time  Max-MAC
----- -------- --------- ------------ --------- -------
 7/11 disabled  shutdown            10        10       1
```

Syntax: show port security <module> | <portnum>

Possible values: N/A

Default value: N/A

show-portname

Displays the name of the interface instead of its number in the Syslog.

EXAMPLE:

```
BigIron># 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 Lab2, state up
Dec 15 18:45:15:I:Warm start
```

This command is applied globally to all interfaces on Layer 2 Switches and Layer 3 Switches.

Syntax: [no] Ip show-portname

When you display the messages in the Syslog, you see the interface name under the Dynamic Log Buffer section. The actual interface number is appended to the interface name. For example, if the interface name is "lab" and its port number is "2", you see "lab2" displayed as in the example below:

By default, this feature is disabled and the interfaces slot/port number appears in the Syslog.

show port security autosave

Displays the secure MAC addresses that have been saved to the configuration by the port security autosave feature.

EXAMPLE:

```
BigIron# show port security autosave
```

Syntax: show port security autosave

Possible values: N/A

Default value: N/A

show port security mac

Lists the secure MAC addresses configured on the device.

EXAMPLE:

```
BigIron(config)# show port security mac
Port  Num-Addr Secure-Src-Addr Resource Age-Left  Shutdown/Time-Left
----- -------- --------------- -------- --------- ------------------
 7/11        1  0050.da18.747c    Local        10       no
```

Syntax: show port security mac

Possible values: N/A

Default value: N/A

show port security statistics

Displays port security statistics for an interface or for a module.

EXAMPLE:

To display port security statistics for interface 7/11:

```
BigIron# show port security statistics e 7/11
Port  Total-Addrs Maximum-Addrs Violation Shutdown/Time-Left
----- ----------- ------------ --------- ------------------
 7/11           1            1         0       no
```

To display port security statistics for a module:

```
BigIron# show port security statistics 7
Module 7:
  Total ports: 0
  Total MAC address(es): 0
  Total violations: 0
  Total shutdown ports 0
```

Syntax: show port security statistics <portnum> | <module>

Possible values: N/A

Default value: N/A

show pos

Displays general information about the POS module.

EXAMPLE:

```
BigIron# show pos
```

This command displays the state of the POS module, the software version running on the module, the contents of the primary and secondary flash on the module, the system uptime, and the status of the CPUs on the module.

Syntax: show pos

Possible values: N/A

Default value: N/A

show priority-mapping

On Chassis devices and the TurboIron/8, displays the queues to which the 802.1p priorities are assigned.

EXAMPLE:

To display the queue assignments for all the priorities, enter the following command at any level of the CLI:

```
BigIron(config)# show priority-mapping all
802.1p priority 0 mapped to qos profile qosp0
802.1p priority 1 mapped to qos profile qosp0
802.1p priority 2 mapped to qos profile qosp1
802.1p priority 3 mapped to qos profile qosp1
802.1p priority 4 mapped to qos profile qosp2
802.1p priority 5 mapped to qos profile qosp2
802.1p priority 6 mapped to qos profile qosp3
802.1p priority 7 mapped to qos profile qosp3
```

In this example, the priorities still have their default queue assignments.

Syntax: show priority-mapping all | <num>

Possible values: N/A

Default value: N/A

show process cpu

Displays CPU utilization statistics for each routing protocol.

NOTE: This command does not apply to Layer 2 Switches.

You can display the percentage of the CPU that was devoted to processing for each protocol during the following time intervals:

- The previous five seconds
- The previous minute
- The previous five minutes
- The previous fifteen minutes
- A number of seconds you specify (from 1 – 900)

The command also lists the total number of milliseconds the CPU has spent on each protocol since the software was reloaded.

Beginning with software release 07.6.02, CPU utilization includes statistics for ACL, NAT, 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:

```
BigIron# 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
NAT               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              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:

```
BigIron# show process cpu
The system has only been up for 6 seconds.
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
NAT                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               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:

```
BigIron# 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
NAT                0       0.00
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.

Possible values: N/A

Default value: N/A

show ptrace

Displays which packet trace functions have been activated with the **ptrace** command.

EXAMPLE:

```
BigIron(config)# sh ptrace

IP: ptrace was turned ON
TCP: ptrace was turned ON
TELNET: ptrace was turned ON
```

Syntax: show ptrace

Possible values: N/A

Default value: N/A

show qos

Displays configuration information for ToS-based QoS.

NOTE: This command is supported only on the NetIron stackable Layer 3 Switch, and is available only in software release 07.1.16 or higher. The command is not supported in software release 07.2.x.

EXAMPLE:

To display configuration information, enter the following command at any level of the CLI:

```
NetIron(config-if-1)# show qos
Interface QoS , Marking and Trust Level:
  i/f   |    QoS    |   Mark    |  Trust-Level
--------+-----------+-----------+---------------
1       |    Yes    |    COS    |   IP Prec
2       |    No     |    No     |   Layer 2 CoS
3       |    No     |    No     |   Layer 2 CoS
4       |    No     |    No     |   Layer 2 CoS
5       |    Yes    |    No     |   DSCP
6       |    No     |    No     |   Layer 2 CoS
7       |    No     |    No     |   Layer 2 CoS
8       |    No     |    No     |   Layer 2 CoS
9       |    No     |    No     |   Layer 2 CoS
10      |    No     |    No     |   Layer 2 CoS
11      |    No     |    No     |   Layer 2 CoS
12      |    No     |    No     |   Layer 2 CoS
13      |    No     |    No     |   Layer 2 CoS
14      |    No     |    No     |   Layer 2 CoS
15      |    No     |    No     |   Layer 2 CoS
16      |    No     |    No     |   Layer 2 CoS
17      |    No     |    No     |   Layer 2 CoS
18      |    No     |    No     |   Layer 2 CoS
ve1     |    Yes    | COS, DSCP |   IP Prec
ve10    |    No     |    No     |   Layer 2 CoS

IP Precedence-DSCP map:

  ip-prec:     0     1     2     3     4     5     6     7
  ------------------------------------------------------------
     dscp:    10    11    12    13    14    15    16    17


DSCP-Priority map: (dscp = d1d2)

     d2|  0   1   2   3   4   5   6   7   8   9
  d1   |
  -----+----------------------------------------
   0   |  0   0   0   0   0   0   0   0   1   1
   1   |  6   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
```

For more information about the fields in this display, see the “IronClad Quality of Service” chapter in the *Foundry Enterprise Configuration and Management Guide*.

Syntax: show qos

Possible values: N/A

Default value: N/A

show qos-profiles

On Chassis devices and the TurboIron/8, displays the QoS settings.

EXAMPLE:

To display the QoS settings for all the queues, enter the following command from any level of the CLI:

```
BigIron(config)# show qos-profiles all
bandwidth scheduling mechanism: weighted priority
Profile qosp3     : PREMIUM     bandwidth requested  75% calculated  75%
Profile qosp2     : HIGH        bandwidth requested  10% calculated  13%
Profile qosp1     : NORMAL      bandwidth requested  10% calculated   8%
Profile qosp0     : BEST-EFFORT bandwidth requested   5% calculated   4%
```

Syntax: show qos-profiles all | <name>

Possible values: N/A

Default value: N/A

show qos-tos

Displays configuration information for JetCore IP ToS-based QoS.

EXAMPLE:

```
BigIron(config)# show qos-tos
Interface QoS , Marking and Trust Level:

i/f    | QoS       | Mark      | Trust-Level
-------+-----------+-----------+---------------
1      | No        | No        | L2 CoS
2      | No        | No        | L2 CoS
3      | No        | No        | L2 CoS
4      | No        | No        | L2 CoS
5      | No        | No        | L2 CoS
6      | No        | No        | L2 CoS
7      | No        | No        | L2 CoS
8      | No        | No        | L2 CoS
9      | No        | No        | L2 CoS
10     | No        | No        | L2 CoS
... <lines omitted for brevity>

49     | No        | No        | L2 CoS
50     | No        | No        | L2 CoS
ve5    | No        | No        | L2 CoS
ve8    | No        | No        | L2 CoS
ve18   | No        | No        | L2 CoS
ve22   | No        | No        | L2 CoS

COS-DSCP map:

     COS: 0 1 2 3 4 5 6 7
  ------------------------------------------------------------
     dscp: 0 8 16 24 32 40 48 56
```



```
IP Precedence-DSCP map:

  ip-prec: 0 1 2 3 4 5 6 7
  ---------------------------------------------------------
     dscp: 0 8 16 24 32 40 48 56

DSCP-Priority map: (dscp = d1d2)

     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

DSCP-DSCP map: (dscp = d1d2)

     d2| 0 1 2 3 4 5 6 7 8 9
  d1   |
  -----+----------------------------------------
     0 | 0 1 2 3 4 5 6 7 8 9
     1 | 10 11 12 13 14 15 16 17 18 19
     2 | 20 21 22 23 24 25 26 27 28 29
     3 | 30 31 32 33 34 35 36 37 38 39
     4 | 40 41 42 43 44 45 46 47 48 49
     5 | 50 51 52 53 54 55 56 57 58 59
     6 | 60 61 62 63
```

Syntax: show qos-tos

Possible values: N/A

Default value: N/A

show rate-limit adjusted-rate

Displays the adjusted rates for a specific range of Average Rates for JetCore hardware-based rate limiting. The device adjusts the rate you enter to a rate that is valid on the device.

EXAMPLE:

To display the adjusted rates for a specific range of Average Rates for inbound rate limiting, enter a command such as the following:

```
BigIron# show rate-limit adjusted-rate inbound 2000000 3000000
On 10/100 ports:
Time interval:  52 * 0.0192 ms
Rate 2000000 to 2179487 (bits/sec) will be mapped to 2051328 (bits/sec)
Rate 2179488 to 2435897 (bits/sec) will be mapped to 2307840 (bits/sec)
Rate 2435898 to 2692307 (bits/sec) will be mapped to 2564352 (bits/sec)
Rate 2692308 to 2948717 (bits/sec) will be mapped to 2820608 (bits/sec)
Rate 2948718 to 3000000 (bits/sec) will be mapped to 3077120 (bits/sec)

On Gig ports:
Time interval:  13 * 0.0192 ms
Rate 2000000 to 2564102 (bits/sec) will be mapped to 2051328 (bits/sec)
Rate 2564103 to 3000000 (bits/sec) will be mapped to 3077120 (bits/sec)
```

This example shows the adjusted rates for Average Rates between 2000000 and 3000000 bps. The rates for 10/100 Ethernet ports and Gigabit Ethernet ports are different and are listed separately.

Syntax: show rate-limit adjusted-rate inbound <start-rate> <end-rate>

The **inbound** parameter specifies that you want to display rates for inbound rate limiting. The adjusted rates for inbound rate limiting and outbound rate limiting are not the same. To display rates for outbound rate limiting, use the command in the next example.

The <start-rate> <end-rate> parameter specifies the range of Average Rates for which you want to list the adjusted rates. You can specify a range of up to 10000000 (10 million) bps. For example, you can specify 10000000 to 19999999, but not 10000000 to 20000000.

EXAMPLE:

To display the adjusted rates for a specific range of Average Rates for outbound rate limiting, enter a command such as the following:

```
BigIron# show rate-limit adjusted-rate outbound gig-port 30000000 40000000

Time interval:  32 * 0.0192 ms
Rate 30000000 to 30416665 (bits/sec) will be mapped to 30000128 (bits/sec)
Rate 30416666 to 31249997 (bits/sec) will be mapped to 30833664 (bits/sec)
Rate 31249998 to 32083330 (bits/sec) will be mapped to 31666688 (bits/sec)
Rate 32083331 to 32916665 (bits/sec) will be mapped to 32500224 (bits/sec)
Rate 32916666 to 33749997 (bits/sec) will be mapped to 33333760 (bits/sec)
Rate 33749998 to 34583330 (bits/sec) will be mapped to 34166784 (bits/sec)
Rate 34583331 to 35416666 (bits/sec) will be mapped to 35000320 (bits/sec)
Rate 35416667 to 36249997 (bits/sec) will be mapped to 35833344 (bits/sec)
Rate 36249998 to 37083330 (bits/sec) will be mapped to 36666880 (bits/sec)
Rate 37083331 to 37916666 (bits/sec) will be mapped to 37500416 (bits/sec)
Rate 37916667 to 38749997 (bits/sec) will be mapped to 38333440 (bits/sec)
Rate 38749998 to 39583330 (bits/sec) will be mapped to 39166976 (bits/sec)
Rate 39583331 to 40000000 (bits/sec) will be mapped to 40000512 (bits/sec)
```

This command shows the adjusted rates between 30000000 and 40000000 bps for outbound rate limiting on a Gigabit Ethernet port.

Syntax: show rate-limit adjusted-rate outbound gig-port | non-gig-port <start-rate> <end-rate>

The **outbound** parameter specifies that you want to display rates for outbound rate limiting. The adjusted rates for inbound rate limiting and outbound rate limiting are not the same. To display rates for inbound rate limiting, use the command in the previous example.

The **gig-port | non-gig-port** parameter specifies the port type. The valid rates differ depending on the port type.

The <start-rate> <end-rate> parameter specifies the range of Average Rates for which you want to list the adjusted rates. You can specify a range of up to 10000000 (10 million) bps. For example, you can specify 10000000 to 19999999, but not 10000000 to 20000000.

Possible values: See above

Default value: N/A

show rate-limit fixed

Displays configuration information and statistics for Fixed Rate Limiting.

EXAMPLE:

```
BigIron(config)# show rate-limit fixed

Total rate-limited interface count: 6.
    Port     Input rate   RX Enforced    Output rate   TX Enforced
     1/1         500000             3
     2/1                                    1234567           100
     2/2                                    2222222             3
     2/3                                    1234567            15
     2/4                                    1238888            12
     2/5                                    1238888             7
```

Syntax: show rate-limit fixed

Possible values: N/A

Default value: N/A

show rate-limit hardware-rate-limit-status

Displays rate limiting configuration information for a JetCore Chassis device or the FastIron 4802.

EXAMPLE:

To display all the policies on the device, enter the following command at any level of the CLI:

```
BigIron(config-if-1/1)# show rate-limit hardware-rate-limit-status
*****************************************
*        Inbound JetCore Rate Limiting      *
*****************************************

Module: 1
 IPC number: 1
  Rate Limit Mode:  Port Based
  Time Interval:    13*0.0192 (ms)
  Credit Size:      32
  Gig Enabled:      Yes

  Port: 1/1, Rate: 3077120(bits/sec), Priority Queue: all, Dir: inbound, ACL: none
  Port: 1/2, Rate: 6153984(bits/sec), Priority Queue: all, Dir: inbound, ACL: none

 IPC number: 2
  Rate Limit Mode:  Port Based
  Time Interval:    13*0.0192 (ms)
  Credit Size:      32
  Gig Enabled:      Yes

  Port: 1/6, Rate: 6153984(bits/sec), Priority Queue: all, Dir: inbound, ACL: none
  Port: 1/2, Rate: 3077120(bits/sec), Priority Queue: all, Dir: inbound, ACL: none

Module: 2
 IPC number: 1
  Rate Limit Mode:  Port Based
  Time Interval:    13*0.0192 (ms)
  Credit Size:      32
  Gig Enabled:      Yes

  Port: 2/2, Rate: 3077120(bits/sec), Priority Queue: all, Dir: inbound, ACL: none
  Port: 2/3, Rate: 3077120(bits/sec), Priority Queue: all, Dir: inbound, ACL: none

 IPC number: 2
  Rate Limit Mode:  Port Based
  Time Interval:    13*0.0192 (ms)
  Credit Size:      32
  Gig Enabled:      Yes

  Port: 2/7, Rate: 6153984(bits/sec), Priority Queue: all, Dir: inbound, ACL: none
  Port: 2/8, Rate: 6153984(bits/sec), Priority Queue: all, Dir: inbound, ACL: none
```



```
*****************************************
*         Outbound JetCore Rate Limiting      *
*****************************************

Module: 1
 IPC number: 1
  Rate Limit Mode:  Port Based
  Time Interval:    32*0.0192 (ms)
  Credit Size:      64
  Gig Enabled:      Yes

  Port: 1/3, Rate: 30000128(bits/sec), Priority Queue: all, Dir: outbound, ACL:
none

 IPC number: 2
  Rate Limit Mode:  Port Based
  Time Interval:    32*0.0192 (ms)
  Credit Size:      64
  Gig Enabled:      Yes

  Port: 1/8, Rate: 60000256(bits/sec), Priority Queue: all, Dir: outbound, ACL:
none

Module: 2
 IPC number: 1
  Rate Limit Mode:  Port Based
  Time Interval:    32*0.0192 (ms)
  Credit Size:      64
  Gig Enabled:      Yes

  Port: 2/2, Rate: 30000128(bits/sec), Priority Queue: all, Dir: outbound, ACL:
none
  Port: 2/3, Rate: 30000128(bits/sec), Priority Queue: all, Dir: outbound, ACL:
none

 IPC number: 2
  Rate Limit Mode:  Port Based
  Time Interval:    32*0.0192 (ms)
  Credit Size:      64
  Gig Enabled:      Yes

  Port: 2/5, Rate: 30000128(bits/sec), Priority Queue: all, Dir: outbound, ACL:
none
  Port: 2/8, Rate: 60000256(bits/sec), Priority Queue: all, Dir: outbound, ACL:
none
```

For information about the fields in the display, see the "JetCore Adaptive Rate Limiting" chapter in the *Foundry Enterprise Configuration and Management Guide*.

Possible values: N/A

Default value: N/A

show relative-utilization

Displays an uplink utilization list, which allows you 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.

EXAMPLE:

To display an uplink utilization list:

```
BigIron(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>

Possible values: The <num> parameter specifies the list number.

Default value: N/A

show reload

Displays the time and date for scheduled system reloads.

EXAMPLE:

```
BigIron# show reload
```

Syntax: show reload

Possible values: N/A

Default value: N/A

show rmon alarm

Displays any reported RMON alarms for the system.

EXAMPLE:

```
BigIron# show rmon alarm

Alarm table is empty
```

Syntax: show rmon alarm [<alarm-table-entry>]

Possible values: N/A

Default value: N/A

show rmon event

Displays any reported RMON events for the system.

EXAMPLE:

```
BigIron# show rmon event

Event table is empty
```

Syntax: show rmon event [<event-table-entry>]

Possible values: N/A

Default value: N/A

show rmon history

Displays the RMON history for the system.

EXAMPLE:

```
BigIron# show rmon history
History 1 is active, owned by monitor
 Monitors interface 1/1 (ifIndex 1) every 30 seconds
 25 buckets were granted to store statistics

History 2 is active, owned by monitor
 Monitors interface 1/1 (ifIndex 1) every 1800 seconds
 25 buckets were granted to store statistics

History 3 is active, owned by monitor
 Monitors interface 5/20 (ifIndex 148) every 30 seconds
 25 buckets were granted to store statistics

History 4 is active, owned by monitor
 Monitors interface 5/20 (ifIndex 148) every 1800 seconds
 25 buckets were granted to store statistics
```

Syntax: show rmon history [<control-table-entry>]

Possible values: N/A

Default value: N/A

show rmon statistics

Displays detailed statistics for each port.

EXAMPLE:

```
BigIron# sh rmon st
```

Syntax: show rmon statistics [ethernet <portnum>] | [<num>]

The **ethernet** <portnum> parameter displays the RMON port statistics for the specified port.

The <num> parameter displays the specified entry. Entries are numbered beginning with 1.

Possible values: see above

Default value: N/A

show route-map

Displays the device's active route map configuration. Use this command when you want to view the route map configuration without displaying the entire running-config.

EXAMPLE:

To display the device's active route map configuration, enter the following command at any level of the CLI:

```
BigIron# 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:

```
BigIron# 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>]

Possible values: see above

Default value: N/A

show running-config

Displays the running configuration of the Foundry device on the terminal screen. In software releases 07.6.03 and later, this command also displays the size of the running-config file.

NOTE: This command is equivalent to the **write terminal** command.

EXAMPLE:

```
BigIron# show running-config
```

Syntax: show running-config
[interface atm | ethernet | loopback | pos | ve <portnum>... [to <portnum>] |
[vlan]

The **interface atm | ethernet | loopback | pos | ve** <portnum>... [**to** <portnum>] parameter specifies one or more interfaces. You can specify a list, a range, or both.

To specify a list of interfaces, entering each interface's type and number as follows:

ethernet 1/1 atm 3/1 pos 4/2

Enter each interface's type, a space, and the port number, then enter another space before entering the next interface's type.

To enter a range of interfaces, enter the starting interface number (the lower one), **to**, and the ending interface number. Here is an example:

ethernet 1/1 to 2/3

You can enter a list and a range on the same command line. Here are some examples:

ethernet 1/1 to 1/4 atm 3/1 pos 4/2

pos 2/1 to 2/2 ethernet 4/1 to 4/4 atm 5/1

The **vlan** parameter displays configuration information for VLANs.

NOTE: If you have enabled the display of passwords with the **enable password-display** CONFIG command, SNMP community strings and passwords are displayed when you enter the **show running-config** command in Privileged EXEC mode, but not in User EXEC mode.

Possible values: N/A

Default value: N/A

show server

Displays configuration information and statistics for a web server address you added using the server real-name command.

This command applies only to Layer 3 Switches you have configured to assist ServerIrons, third-party Server Load Balancers, or directly connected web servers with globally-distributed Server Load Balancing (SLB). See the "Route Health Injection" chapter of the *Foundry Enterprise Configuration and Management Guide*.

EXAMPLE:

```
BigIron# show server real tinman

Real Servers Info

Server State - 1:enabled, 2:failed, 3:test, 4:suspect, 5:grace_dn, 6:active
Name:tinman             IP:  209.157.23.60:4    State:6
```

Syntax: show server real <name> | keepalive-port

See the "Route Health Injection" chapter of the *Foundry Enterprise Configuration and Management Guide* for an explanation of the fields in this display.

Possible values: N/A

Default value: N/A

show sflow

Displays sFlow configuration information and statistics.

EXAMPLE:

```
BigIron(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

Possible values: N/A

Default value: N/A

show snmp engineid

Displays the engine ID of a management module. (For SNMP version 3.)

EXAMPLE:

```
BigIron(config)# show snmp engineid
Local SNMP Engine ID: 800007c70300e05290ab60
Engine Boots: 3
Engine time: 5
```

Syntax: show snmp engineid

The engine ID identifies the source or destination of the packet.

The engine boots represents the number of times that the SNMP engine reinitialized itself with the same engine ID. If the engineID is modified, the boot count is reset to 0.

The engine time represents the current time with the SNMP agent.

Possible values: N/A

Default values: N/A

show snmp group

Displays the SNMP group information.

EXAMPLE:

```
BigIron(config)# show snmp group
groupname = exceptifgrp
security model = v3
security level = authNoPriv
ACL id = 2
readview = exceptif
writeview = <none>
```

Syntax: show snmp group

Possible values: The value for security level can be one of the following:

Security Level	Authentication
<none>	If the security model shows v1 or v2, then security level is blank. User names are not used to authenticate users; community strings are used instead.
noauthNoPriv	Displays if the security model shows v3 and user authentication is by user name only.
authNoPriv	Displays if the security model shows v3 and user authentication is by user name and the MD5 or SHA algorithm.

Default values: N/A

show snmp server

Lists system administrative information—contact name, system location, community strings, and traps enabled for a Foundry device.

EXAMPLE:

```
BigIron# show snmp server
Contact: Marshall
Location: Copy Center
```



```
Community(ro): public
Community(rw): private
Traps
                    Cold start: Enable
                       Link up: Enable
                     Link down: Enable
                Authentication: Enable
     Locked address violation: Enable
         Power supply failure: Enable
                  Fan failure: Enable
Redundant module state change: Enable
          Temperature warning: Enable
                 STP new root: Enable
          STP topology change: Enable
                         ospf: Enable
                         fsrp: Enable
                         vrrp: Enable

 Total Trap-Receiver Entries: 4
Trap-Receiver IP Address       Community
     1        207.95.6.211
     2        207.95.5.21
```

Syntax: show snmp server

Possible values: N/A

Default value: N/A

show snmp user

Displays SNMP user account information. (For SNMP version 3.)

```
BigIron(config)# show snmp user
```

```
username = bob
acl id = 2
group = admin
security model = v3
group acl id = 0
authtype = md5
authkey = 3aca18d90b8d172760e2dd2e8f59b7fe
privtype = des,  privkey = 1088359afb3701730173a6332d406eec
engine ID= 800007c70300e052ab0000
```

Syntax: show snmp user

Possible values: N/A

Default values: N/A

show sntp associations

Displays information about SNTP associations.

EXAMPLE:

```
BigIron# 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
```


The following table describes the information displayed by 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

Syntax: show sntp associations

Possible values: N/A

Default value: N/A

show sntp status

Displays information about SNTP status.

EXAMPLE:

```
BigIron# 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
```

The following table describes the information displayed by 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

This Field...	Indicates...
root delay	Total delay along the path to the root clock
root dispersion	Dispersion of the root path
peer dispersion	Dispersion of the synchronized peer

Syntax: show sntp status

Possible values: N/A

Default value: N/A

show span

Displays spanning tree statistics such as root cost, root port, and priority. For descriptions of the information shown by this display, see the "Configuring Spanning Tree Protocol (STP) and IronSpan" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

EXAMPLE:

```
SW-FI4802-PREM# 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> [atm <portnum> | ethernet <portnum> | pos <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.

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 “show span detail”.

Possible values: N/A

Default value: N/A

show span detail

Displays detailed STP information for individual ports. For descriptions of the information shown by this display, see the "Configuring Spanning Tree Protocol (STP) and IronSpan" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

EXAMPLE:

```
BigIron# 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>
```

EXAMPLE:

```
BigIron(config)# show span detail vlan 1 ethernet 1/7
Port 1/7 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> [atm <portnum> | ethernet <portnum> | pos <portnum>] | <num>]

The **vlan** <vlan-id> parameter specifies a VLAN.

The **atm** <portnum> | **ethernet** <portnum> | **pos** <portnum> parameter specifies an individual port within the VLAN (if specified).

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.

Possible values: N/A

Default value: N/A

show span pvst-mode

Displays Per VLAN Spanning Tree (PVST) information for ports on a Foundry device.

EXAMPLE:

Here is the display if you are using software release 07.6.01 or later.

```
BigIron(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
```

Here is the display if you are using a software release from 07.1.00 – 07.6.00x.

```
BigIron(config)# show span pvst-mode

VLAN   Port   PVST   PVST
 ID    Num.   Cfg.   On(by cfg. or detect)
 200   10      0      1
 200   11      1      1
```

This example shows that for VLAN 200, PVST support is statically enabled on port 11. PVST is not statically enabled on Port 10, but because port 10 received an incoming PVST BPDU on its interface, the port converted to using PVST mode.

Syntax: show span pvst-mode

Possible values: N/A

Default value: N/A

show span vlan

Displays global and port STP information for a given VLAN on a Foundry device.

EXAMPLE:

```
BigIron# show span vlan 2
Global Bridge 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
 2   800000e0520002f5  0      Root  8000   20       2     2     15     0   0
00e0520002f5
Port STP Parameters:
VLAN  Port  Prio  Path State     Fwd          Design   Design    Design
 ID   Num   rity  Cost          Trans         Cost     Root     Bridge
           Hex
    2        1      0080    0                DISABLED         0
0000000000000000    0000000000000000
   2         2      0080    0                DISABLED         0
0000000000000000    0000000000000000
   2         3      0080    0                DISABLED         0
0000000000000000    0000000000000000
   2         4      0080    0                DISABLED         0
0000000000000000    0000000000000000
   2         5      0080    0                DISABLED         0
0000000000000000    0000000000000000
```

Syntax: show span vlan <vlan-id>

Possible values: N/A

Default value: N/A

show statistics

Displays port statistics for a Foundry device (transmit, receive, collisions, errors).

EXAMPLE:

```
BigIron# show statistics
        Buffer Manager           Queue
  [Pkt Receive Pkt Transmit]
           0             0


Ethernet                      Packets               Collisions        Errors
Port              [Receive            Transmit]    [Recv  Txmit]   [InErr  OutErr]
3/1                      1                    2       0       0       0       0
3/2                      0                    0       0       0       0       0
3/3                      0                    0       0       0       0       0
3/4                      0                    0       0       0       0       0
```

Syntax: show statistics [atm <portnum> [to <portnum>]] | [ethernet <portnum> [to <portnum>]] | [pos <portnum> [to <portnum>]] | [slot <slot-num>]

The **atm** <portnum> parameter displays statistics for a specific ATM port.

The **ethernet** <portnum> parameter displays statistics for a specific Ethernet port.

The **pos** <portnum> parameter displays statistics for a specific POS port.

The **slot** <slot-num> parameter displays statistics for a specific chassis slot.

NOTE: The **slot** <slot-num> parameter applies only to Chassis devices.

NOTE: The **show statistics brief** command displays the same information as the **show statistics** command entered without any optional parameters.

This display shows the following information for each port.

Table 43.5: CLI Display of Port Statistics

This Field...	Displays...
Packet counters	
Receive	The number of packets received on this interface.
Transmit	The number of packets transmitted on this interface.
Collision counters **Note**: These statistics do not apply to POS modules.	
Recv	The number of collisions that have occurred when receiving packets.
Txmit	The number of collisions that have occurred when sending packets.
Packet Errors These fields show statistics for various types of packet errors. The device drops packets that contain one of these errors.	
InErr	The number of inbound packets that had errors.
OutErr	The number of outbound packets that had errors.

Here is an example of the detailed statistics display for an individual port.

```
BigIron# show statistics ethernet 2/1
Port  Link State     Dupl Speed Trunk Tag Priori MAC            Name
2/1   Up   Forward   Half 10M   None  No  level0 00e0.52a9.2b00

Port 2/1 Counters:
                InOctets          45994           OutOctets          31528
                  InPkts            562             OutPkts             48
          InBroadcastPkts           462       OutBroadcastPkts            1
          InMulticastPkts            38       OutMulticastPkts            0
               InDiscards             0            OutDiscards            0
                 InErrors             0              OutErrors            0
            InCollisions              0         OutCollisions             0
                                                OutLateCollisions         0
              Alignment               0                    FCS            0
              GiantPkts               0              ShortPkts            0
            InBitsPerSec           1160         OutBitsPerSec          832
            InPktsPerSec              1         OutPktsPerSec            0
           InUtilization          0.01%        OutUtilization         0.00%
```


NOTE: In software release 07.2.03 and later, runts are still counted in the OutCollisions field. However, a runt is not counted in the InErrors field for 10 Mpbs ports running in half-duplex mode.

- For 10 Mbps ports in half-duplex mode, the software does not increment the InErrors field when a runt packet is detected. Instead, the runt is counted in the OutCollisions field.
- For 100 Mpbs ports, and for 10 Mpbs ports in full-duplex mode, the software does increment the InErrors field when a runt packet is detected.
    - Statistics for runt packets are not applicable to Gigabit Ethernet ports or POS ports, so the field is not used for these port types.

NOTE: The InCollisions field is not used for any ports and should always contain the value 0.

NOTE: For 10 Mbps ports, the OutLateCollisions field is incremented each time the port detects an Ethernet collision that occurs 51.2 microseconds or later, after the data was transmitted onto the network segment the port is connected to.

For 100 Mbps ports, the OutLateCollisions field is incremented each time the port detects an Ethernet collision that occurs 5.12 microseconds or later (one-tenth the interval for 10 Mpbs ports), after the data was transmitted onto the network segment the port is connected to.

A collision that occurs before the late collision time (51.2 microseconds for 10 Mbps and 5.12 microseconds for 10 Mbps) is considered to be a normal collision and is counted in the OutCollisions field instead.

NOTE: Ethernet devices must allow a minimum idle period between transmission of frames known as interframe gap (IFG) or interpacket gap (IPG). The gap provides a brief recovery time between frames to allow devices to prepare to receive of the next frame. The minimum IFG is 96 bit times, which is 9.6 microseconds for 10 Mbps Ethernet, 960 nanoseconds for 100 Mbps Ethernet, and 96 nanoseconds for 1 Gbps Ethernet. In addition to accounting for the bit rate on the port, port utilization should also account for the IFG, which normally is filtered by the packet synchronization circuitry.

Refer to the etherHistoryUtilization objects in the *RFC 1757: Remote Network Monitoring Management Information Base* for details.

Possible values: see above

Default value: statistics for all ports are displayed

show statistics dos-attack

Displays information about ICMP and TCP SYN packets dropped because burst thresholds were exceeded.

EXAMPLE:

```
BigIron# show statistics dos-attack
---------------------------- Local Attack Statistics --------------------------
ICMP Drop Count    ICMP Block Count    SYN Drop Count    SYN Block Count
---------------    ----------------    --------------    ---------------
              0                   0                 0                   0
--------------------------- Transit Attack Statistics -------------------------
Port   ICMP Drop Count    ICMP Block Count    SYN Drop Count    SYN Block Count
-----  ---------------    ----------------    --------------    ---------------

3/11                 0                   0                 0                   0
```


Syntax: show statistics dos-attack

Possible values: N/A

Default value: N/A

show super-span

Displays configuration information and statistics for SuperSpan.

EXAMPLE:

```
BigIron(config)# show super-span
CID 1 Boundary Ports:
     Port   C-BPDU   C-BPDU   T-BPDU   T-BPDU
            Rxed     Txed     Rxed     Txed
     1/1    1        0        0        0
     1/2    0        0        0        0
     Total  1        0        0        0

CID 2 Boundary Ports:
     Port   C-BPDU   C-BPDU   T-BPDU   T-BPDU
            Rxed     Txed     Rxed     Txed
     2/1    0        0        3        0
     2/2    0        0        0        0
     Total  0        0        3        0
```

In this example, the device has two SuperSpan customer IDs.

Syntax: show superspan [cid <num>]

The **cid** <num> parameter specifies a SuperSpan customer ID. If you do not specify a customer ID, information for all the customer IDs configured on the device is shown.

For information about the fields in this display, see the "Displaying SuperSpan Information" section in the "Configuring Spanning Tree Protocol (STP) and IronSpan" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Possible values: N/A

Default value: N/A

show tech-support

Shows technical details to you for assistance in troubleshooting issues when working with technical support. The information shown is a sub-set of all the available information.

NOTE: This command is not supported at the User EXEC level of the CLI.

EXAMPLE:

```
BigIron# show tech-support
show tech
  SW: Version 07.5.00b63T53 Copyright (c) 1996-2001 Foundry Networks, Inc.
      Compiled on Sep 21 2001 at 04:37:22 labeled as B2R07500b63
  HW: BigIron 8000 Router, SYSIF version 21
==========================================================================
SL 1: B8GMR Fiber Management Module, SYSIF 2, M2, ACTIVE
      Serial #: 12345678
 2048 KB BRAM, SMC version 1, ICBM version 21
  512 KB PRAM(512K+0K) and 2048*8 CAM entries for DMA  0, version 0209
  512 KB PRAM(512K+0K) and shared CAM entries for DMA  1, version 0209
  512 KB PRAM(512K+0K) and 2048*8 CAM entries for DMA  2, version 0209
  512 KB PRAM(512K+0K) and shared CAM entries for DMA  3, version 0209
==========================================================================
SL 3: B24E Copper Switch Module
      Serial #: Non Exist.
 2048 KB BRAM, SMC version 2, ICBM version 21
  256 KB PRAM(256K+0K) and 2048*8 CAM entries for DMA  8, version 0808
  256 KB PRAM(256K+0K) and shared CAM entries for DMA  9, version 0808
  256 KB PRAM(256K+0K) and shared CAM entries for DMA 10, version 0808
==========================================================================
Active management module:
  240 MHz Power PC processor 603 (version 7/1201) 63 MHz bus
  512 KB boot flash memory
 8192 KB code flash memory
  256 KB SRAM
  128 MB DRAM
The system uptime is 6 seconds
SW-telnet@BigIron#Port Link State     Dupl Speed Trunk Tag Priori MAC
1/1  Up   Forward   Full 1G    None  No  level0 00e0.5280.1400 1/1
1/2  Down None      None None  None  No  level0 00e0.5280.1401 1/2
[. . . . . .]
 The system had been up for 422 minutes
General Registers:
04208278 0425f358 0421c200 00009030 00000000 00000000 000000ff 044b94e8
[. . . . . .]
```

Syntax: show tech-support

Possible values: N/A

Default value: N/A

show telnet

Shows the IP address of the station with the active Telnet session. Up to five read-only access Telnet sessions are supported on the Foundry device at one time. Write access through Telnet is limited to one session.

EXAMPLE:

```
BigIron# show telnet

Console connections:
        established, active
        14 seconds in idle
Telnet connections:
 1      established, client ip address 192.168.1.234
        7 seconds in idle
 2      established, client ip address 192.168.1.234
        3 seconds in idle
 3      closed
 4      closed
 5      closed
SSH connections:
 1      closed
 2      closed
 3      closed
 4      closed
 5      closed
```

Syntax: show telnet

Possible values: N/A

Default value: N/A

show terminal

Specifies the number of lines that will be displayed on the screen as specified by the **terminal length**, **page display**, and **skip-page-display** commands. It also shows if the **enable skip-page-display** command has been configured. The **enable skip-page-display** command allows you to use the skip-page-display to disable the configured page-display settings.

EXAMPLE:

```
BigIron(config)# show terminal
Length: 24 lines
Page display mode (session): enabled
Page display mode (global): enabled
```

Syntax: show terminal

Possible values: N/A

Default value: N/A

show topology-group

Displays topology group information.

EXAMPLE:

```
BigIron(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>]

For information about the fields in this display, see the "Displaying Topology Group Information" section in the "Configuring Metro Features" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Possible values: N/A

Default value: N/A

show trunk

Displays trunk groups and their port membership for Foundry devices.

EXAMPLE:

```
BigIron(config)# show trunk ethernet 1/1 to 1/8

Configured trunks:

Trunk ID: 1
Type: Switch
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
Type: Switch
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
 IPX            0        2         1        0        0      0       0        1
 Apple Talk     1        2         0        4        0      0       0        3
 Multicast      4        2         5        2        0      0       2        3
```

Syntax: show trunk [ethernet | pos <portnum> to <portnum>]

Use **<portnum> to <portnum>** to display trunk groups and LACP status information for a specific port range.

For information about this display, see the "Displaying Trunk Group Configuration Information" section in the "Configuring Trunk Groups and Dynamic Link Aggregation" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Possible values: N/A

Default value: N/A

show users

Lists the local access user accounts configured on the device.

EXAMPLE:

```
BigIron# sh u
Username            Password                       Encrypt   Privilege
=================================================================
JB                  $1$$arc/3B93fBiatch/DmGWt1     enabled   0
```

Syntax: show users

Possible values: N/A

Default value: N/A

show version

Lists software, hardware and firmware details for a Foundry device. Much of the information displayed by this command can be used by Foundry technical support to help identify your system if you need help to resolve an issue. The following information might be particularly useful and is highlighted in bold type in the example:

- Software version – The version number of the software. This is the number referred to in release notes and other product documentation.
- Software label – The name of the software image file. This is the name of the file you install into the device's flash memory. Note that the same software version usually has different software labels depending on the product and in come cases on the contents of the software.
- DRAM – the amount of memory on the device. This memory amount can be important if you want to use memory-intensive features such as Border Gateway Protocol version 4 (BGP4).

EXAMPLE:

This example shows the command output on a BigIron 8000 running software version 07.5.00.

```
BigIron# show version
  SW: Version 07.5.00b63T53 Copyright (c) 1996-2001 Foundry Networks, Inc.
      Compiled on Sep 21 2001 at 04:37:22 labeled as B2R07500b63
  HW: BigIron 8000 Router, SYSIF version 21
==========================================================================
SL 1: B8GMR Fiber Management Module, SYSIF 2, M2, ACTIVE
      Serial #: 12345678
 2048 KB BRAM, SMC version 1, ICBM version 21
  512 KB PRAM(512K+0K) and 2048*8 CAM entries for DMA  0, version 0209
  512 KB PRAM(512K+0K) and shared CAM entries for DMA  1, version 0209
  512 KB PRAM(512K+0K) and 2048*8 CAM entries for DMA  2, version 0209
  512 KB PRAM(512K+0K) and shared CAM entries for DMA  3, version 0209
==========================================================================
SL 3: B24E Copper Switch Module
      Serial #: Non Exist.
 2048 KB BRAM, SMC version 2, ICBM version 21
  256 KB PRAM(256K+0K) and 2048*8 CAM entries for DMA  8, version 0808
  256 KB PRAM(256K+0K) and shared CAM entries for DMA  9, version 0808
  256 KB PRAM(256K+0K) and shared CAM entries for DMA 10, version 0808
==========================================================================
Active management module:
  240 MHz Power PC processor 603 (version 7/1201) 63 MHz bus
  512 KB boot flash memory
 8192 KB code flash memory
  256 KB SRAM
  128 MB DRAM
The system uptime is 6 seconds
The system : started=warm start   reloaded=by "reload"
```

Syntax: show version

Possible values: N/A

Default value: N/A

show vlan-group

Displays configuration information for VLAN groups.

EXAMPLE:

```
BigIron# 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
!
```

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.

Possible values: N/A

Default value: N/A

show vlans

Displays the VLANs configured on the system, their member ports, assigned priority, and STP status.

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.

EXAMPLE:

```
BigIron(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 <portnum> | pos <portnum>]

The <vlan-id> parameter specifies a VLAN for which you want to display the configuration information.

The **ethernet** <portnum> | **pos** <portnum> parameter specifies a port. If you use this parameter, the command lists all the VLAN memberships for the port.

Possible values: N/A

Default value: N/A

show vm-map

Displays the Velocity Switching Processor (VSP) CPU allocations for the forwarding modules in a Chassis device.

NOTE: This command applies only to the Velocity Management Module.

EXAMPLE:

To display the slot allocations for the VSP CPUs, enter the following command at any CLI level:

```
BigIron(config)# show vm-map
slot 2 (weight 24 x 100M) is processed by vsp 1/2 (weight 24)
slot 3 (weight 8 x 1000M) is processed by vsp 1/1 (weight 80)
slot 4 (weight 24 x 100M) is processed by vsp 1/3 (weight 24)
```

Syntax: show vm-map

This example shows the slot allocations for a four-slot chassis. The output displays rows only for the slots that contain forwarding modules. No information is displayed for empty slots.

Each row shows the following information:

- The chassis slot ("slot 2" in the first row of the example above)
- The weight of the module in the slot ("weight 24 x 100M" in the first row of the example above)
- The chassis slot that contains the VM1 and the VSP to which the forwarding module described by this row is allocated ("is processed by vsp 1/2"). The "1" in this example indicates the VM1 is in chassis slot 1. The "2" in this example indicates that VSP 2 is handling the forwarding module in slot 2.
- The total weight assigned to the VSP ("weight 24" in the first row of this example).

EXAMPLE:

When per-DMA VSP load sharing is enabled on the device, the show **vm-map** command displays static VSP assignments. For example:

```
BigIron# show vm-map
slot 1 (weight 80 x 100M):
  e   1/5-1/6   is processed by VSP processor 4/2
  e   1/7-1/8   is processed by VSP processor 4/3
slot 2 (weight 24 x 100M) is processed by VSP processor 4/1

Static configuration:
slot 1 (weight 80 x 100M):
  e   1/1-1/2   is processed by VSP processor 4/3
  e   1/3-1/4   is processed by VSP processor 4/1
```

In the example above, per-DMA VSP load sharing has been enabled on the device. The module in slot 1 supports per-DMA VSP load sharing, but the module in slot 2 does not. The VM1 is located in slot 4.

On the module in slot 1, the DMAs controlling ports 1 – 2 and 3 – 4 have been statically assigned to VSPs. The DMAs controlling the other ports on the module have been dynamically assigned to VSPs based on the weight of the DMAs.

All of the ports on the module in slot 2 are assigned to VSP 4/1. Since the module does not support per-DMA VSP load sharing, all of its ports are assigned to a single VSP.

Possible values: N/A

Default value: N/A

show vm-state

Displays general information for a Velocity Management Module.

EXAMPLE:

```
MON-BigIron(config)# show vm-state
================================================
VM MODULE (2) App CPU    0 MB SHM, 3 Application Processors
         CPU 0 in state of VM_STATE_RUNNING
         CPU 1 in state of VM_STATE_RUNNING
         CPU 2 in state of VM_STATE_RUNNING
---------------
Module 2 App CPU 1, SW: Version 07.5.00b2SPT72
Compiled on Jun 04 2001 at 17:14:08 labeled as VSP07500b2SP
DRAM 268M, BRAM 262K, FPGA Version 0050
Code Flash 4M: Primary (1231492 bytes, 07.5.00b2SPT72),
                Secondary (1004047 bytes, 07.2.11T71)
Boot Flash 131K, Boot Version 07.01.00
The system uptime is 0 day 0 hour 44 minute 15 second
General Status: 0 ipc msg rec, 2 ipc msg sent
---------------
Module 2 App CPU 2, SW: Version 07.5.00b2SPT72

Compiled on Jun 04 2001 at 17:14:08 labeled as VSP07500b2SP
DRAM 268M, BRAM 262K, FPGA Version 0050
Code Flash 4M: Primary (1231492 bytes, 07.5.00b2SPT72),
                Secondary (1004047 bytes, 07.2.11T71)
Boot Flash 131K, Boot Version 07.01.00
The system uptime is 0 day 0 hour 44 minute 15 second
General Status: 0 ipc msg rec, 2 ipc msg sent
---------------
Module 2 App CPU 3, SW: Version 07.5.00b2SPT72
Compiled on Jun 04 2001 at 17:14:08 labeled as VSP07500b2SP
DRAM 268M, BRAM 262K, FPGA Version 0050
Code Flash 4M: Primary (1231492 bytes, 07.5.00b2SPT72),
                Secondary (1004047 bytes, 07.2.11T71)
Boot Flash 131K, Boot Version 07.01.00
The system uptime is 0 day 0 hour 44 minute 15 second
General Status: 0 ipc msg rec, 2 ipc msg sent
```

Syntax: show vm-state

Possible values: N/A

Default value: N/A

show vsrp

Displays information for a VSRP VRID.

EXAMPLE:

```
BigIron(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>]

For information about the fields in this display, see the "Displaying VRID Information" section in the "Configuring Metro Features" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Possible values: N/A

Default value: N/A

show vsrp aware

Displays the active VRID interfaces on a VSRP-aware device.

EXAMPLE:

```
BigIron(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

For information about the fields in this display, see the "Displaying the Active Interfaces for a VRID" section in the "Configuring Metro Features" chapter of the *Foundry Switch and Router Installation and Basic Configuration Guide*.

Possible values: N/A

Default value: N/A

show web-connection

Displays the access levels and IP addresses of the devices that currently have Web management interface sessions with the device.

To clear all sessions displayed by this command, see “clear web-connection” on page 5-18.

EXAMPLE:

```
BigIron(config)# show web-connection
User                            IP address
set                             10.10.11.150
```


Syntax: show web-connection

Possible values: N/A

Default value: N/A

show who

Lists the active console and Telnet CLI sessions.

EXAMPLE:

```
BigIron# show who
Console connections:
 established
Telnet connections:
 1 established, client ip address 209.157.22.63
 2 closed
 3 closed
 4 closed
 5 closed
```

Syntax: show who

Possible values: N/A

Default value: N/A

Appendix A
Commands That Require a Reload

Most CLI commands take effect as soon as you enter them. However, a small number of commands require a software reload to take effect. Table A.1 lists the commands.

To place a configuration change made by one of these commands into effect, you must save the change to the startup-config file, then reload the software. If you reload the software without saving the change to the startup-config file, the device does not make the change.

To reload the software, you must perform a cold start. To perform a cold start, do one of the following:

- Enter the **reload** command at the Privileged EXEC level of the CLI.
- Cycle the power by powering down the device, then powering it on again.

NOTE: The **boot system** command does not perform a cold start. It performs a warm start.

Table A.1: Commands That Require a Software Reload

cam-partition	6-19
default-mtu	6-23
ip dont-use-acl	6-45
ip high-perf	6-51
ip multicast-perf	6-59
jumbo	6-80
maximum-number-of-lsa	11-11
Note: This command is not supported in software release 07.1.00 and later. In these releases, OSPF dynamically allocates memory.	
max-routes	11-12
Note: This command is not supported in software release 07.1.00 and later. In these releases, OSPF dynamically allocates memory.	
mcast-hw-replic-disable	6-87
mcast-hw-replic-oar	6-87

Table A.1: Commands That Require a Software Reload (Continued)

mtu	7-59
multicast-flooding	33-7
port-priority	6-92
router dvmrp	6-107
router ipx	6-108
router msdp	6-109
router pim	6-109
system-max	6-131
trunk	6-137
Note: In software release 07.5.00 and later, you do not need to reload the software to place a trunk group configuration change into effect. Instead, you can place the change into effect by entering the **trunk deploy** command at the global CONFIG level of the CLI.	
virtual-interface-mac	6-139