ProCurve Networking
HP Innovation

Management and Configuration Guide

ProCurve Series 8100fl Switches

www.procurve.com

invent

ProCurve Series 8100fl Switches

September 2005
Software Release CY.01.*xx.xxxx* or Greater

Management and Configuration Guide

Publication Number

5990-8867 (Rev B)
September 2005

Applicable Products

ProCurve Switch 8108fl	(J8727A)
ProCurve Switch 8116fl	(J8728A)

Trademark Credits

Ethernet is a registered trademark of Xerox Corporation. Cisco® is a trademark of Cisco Systems, Inc.

Software Credits and Notices

This product includes software developed by Trimble Navigation, Ltd.

Disclaimer

The information contained in this document is subject to change without notice.

HEWLETT-PACKARD COMPANY MAKES NO WARRANTY OF ANY KIND WITH REGARD TO THIS MATERIAL, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. Hewlett-Packard shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

Hewlett-Packard assumes no responsibility for the use or reliability of its software on equipment that is not furnished by Hewlett-Packard.

Warranty

See the Customer Support/Warranty booklet included with the product.

A copy of the specific warranty terms applicable to your Hewlett-Packard products and replacement parts can be obtained from your HP Sales and Service Office or authorized dealer.

Hewlett-Packard Company
8000 Foothills Boulevard, m/s 5551
Roseville, California 95747-5551
http://www.procurve.com

Contents

1 Getting Started

2 Using the Command Line Interface (CLI)

3 File and System Management

4 Configuring Basic Features

5 Security Configuration

6 RIP Configuration

7 OSPF Configuration

8 VRRP Configuration

9 Link Aggregation Configuration

10 Access Control Lists (ACLs)

11 QoS Configuration

12 Bridging Configuration

13 Configuring Routing Policies

14 IP Routing Configuration

— This page is intentionally unused. —

1

Getting Started

Contents

Overview

This *Management and Configuration Guide* is intended for use with the following switches:

ProCurve Switch 8108fl

ProCurve Switch 8116fl

Note

Each device uses the same command line functions. Together, these two devices are referred to in this guide as the *ProCurve Switch 8100fl*.

This guide describes how to use the command line interface (CLI) to configure, manage, monitor, and troubleshoot switch operation. The *Product Documentation CD-ROM* shipped with the switch includes a copy of this guide. You can also download a copy from the ProCurve web site. (See "Getting Documentation From the Web" on page 1-4, below.) For information on other product documentation for the ProCurve Switch 8100fl, refer to "Related Publications" on page 1-4.

Conventions

This guide uses the following conventions for displaying command syntax.

Convention	Description
boldface font	Identifies commands that you enter as shown.
italic font	Identifies elements for which you enter values.
`screen` font	Indicates text that appears on your computer screen.
[ ]	Identifies optional elements. Square brackets are also used to indicate default system prompts on screen.
{ x | y | z }	Indicates required elements of which you select one. Vertical bars (|) are used to separate alternative, mutually exclusive elements.
[x | y | z]	Indicates optional elements of which you select one.
string	Indicates that the entry is a literal set of characters.
[**ctrl**][**Enter**]	Represents a keystroke (or keystrokes) to type on your keyboard.
< >	Indicates nonprinting characters for which you enter values.

Command Prompts

The default configuration for your switch displays one of the following CLI prompts:

```
ProCurve 8108fl#
ProCurve 8116fl#
```

To simplify recognition, this guide uses the hostname `Switch 8100fl` to represent command prompts for both models. For example:

```
Switch 8100fl#
```

Note You can use the **hostname** command to change the text in the CLI prompt.

Screen Simulations

Single lines of screen text and command output are represented like this:

```
Switch 8100fl#show running-config
```

Screens containing more than one line of text and command output are shown in table format like this:

```
Switch 8100fl#show version

ProCurve Networking Switch 8100fl Series System Software
Version CY.01.01.0116
Copyright (c) 1998-2005 by ProCurve Networking.
Compiled on Mon Apr 11 05:01:32 PDT 2005
Bootloader Version CY.01.01.0113
Switch uptime is 1 week, 3 days, 22 hours, 11 minutes, 4
seconds
System restarted by cold reset
System image file is ms-CY.01.01.0116.ver

Switch 8100fl#
```

Related Publications

Read Me First. The *Read Me First* shipped with your switch provides software update information, product notes, and other information. A printed copy is shipped with your switch.

Installation and Getting Started Guide. Use the *Installation and Getting Started Guide* shipped with your switch to prepare for and perform the physical installation. This guide steps you through connecting the switch to your network and assigning IP addressing, as well as describing the LED indications for correct operation and trouble analysis. A PDF version of this guide is also provided on the *Product Documentation CD-ROM* shipped with the switch.

Release Notes. Release notes are posted on the ProCurve web site and provide information on new software updates:

New features and how to configure and use them

Software management, including downloading software to the switch

Software fixes addressed in current and previous releases

Getting Documentation From the Web

To download the latest version of documentation for your switch:

1. Go to the Procurve Networking web site at **http://www.procurve.com**.
2. Click on **technical support**.
3. Click on **product manuals**.
4. Click on the **ProCurve Switch 8100fl** link to view or download the most recent manuals and release notes for this product.

Sources for More Information

For more information on a specific command in the CLI, type the command name followed by "**?**" or use the [**Tab**] key (see "Getting Help with CLI Commands" on page 2-10 for details).

For more information on ProCurve products and technology, visit the ProCurve Networking web site at:

http://www.procurve.com

Need Only a Quick Start?

IP Addressing. If you just want to give the switch an IP address so that it can communicate on your network, ProCurve recommends that you use the CLI to quickly configure IP addressing and enable Telnet access to the switch: see "Setting the Management Module IP Address" on page 4-3 for details.

Note

For an introduction and overview on using the CLI, refer to Chapter 2, "Using the Command Line Interface (CLI)". For instructions on setting up basic features, see Chapter 4, "Configuring Basic Features".

To Set Up and Install the Switch in Your Network

For instructions on how to physically install the switch and its components in your network, refer to the *Installation and Getting Started Guide* that shipped with your switch. This provides information on the following:

- Notes, cautions, and warnings related to installing the switch and its related modules
- Instructions for mounting the switch and physically installing its modules, fans, and power
- Procedures for setting up basic system information and passwords.
- Descriptions for interpreting LED behavior on the switch.

— This page is intentionally unused. —

2

Using the Command Line Interface (CLI)

Contents

Accessing the CLI

The CLI can be accessed through both serial and Telnet connections (including Secure Shell). For initial log on, you must use a serial connection. Once an IP address is assigned to the management interface (see "Setting the Management Module IP Address" on page 4-3), you can access the CLI through a Telnet connection. For more information on using Telnet and SSH sessions, refer to Chapter 5, "Security Configuration".

When accessing the CLI through Telnet, you will prompted for a password if one has been set via local, RADIUS, or TACACS configuration. By default, the password required is the password you enter for general access at initial setup (see "Configuring Passwords" on page 5-2). You also have the option of assigning a separate password for Privileged Exec mode access with the **enable secret** command.

Note

Up to 10 Telnet sessions can run simultaneously on the switch. All sessions can be in Configuration mode at the same time, so you should consider limiting access to the switch to authorized users.

To access the CLI:

1. Once you connect to the device, you will see the following prompt:

   ```
   Switch 8100fl>
   ```

   At this prompt (>), you are at the user Exec mode of the CLI command structure. You can view system status at this level, but you do not have permission to change system configurations. To make configuration and system changes, you must be in (and have authorization to enter) the Privileged Exec mode.

Note

For more information on the CLI Access modes and permissions, see Table 2-1 on page 2-4.

2. To access the Privileged Exec mode from the Exec mode, enter:

   ```
   Switch 8100fl>enable
   ```

You will be prompted for a password if one has been assigned. Otherwise, the prompt will change to the Privileged Exec mode (#):

```
Switch 8100fl#
```

From Privileged Exec mode, you can manage system-level functions and enter Configuration mode to make configuration changes.

3. To access Configuration mode, from Privileged Exec mode enter:

```
Switch 8100fl#configure
```

The prompt will change to Configuration mode:

```
Switch 8100fl(config)#
```

From Configuration mode, you can reach all other configuration levels (for ports on interface modules, for specific protocols, and so on) from this mode.

Notes

The CLI supports partial matching, so you do not need to enter the entire name of a command or option.

CLI commands are not case sensitive.

To help identify the current command level or mode, the CLI prompt changes at each level of the Configuration command structure.

Changes made in Exec and Privileged Exec mode (such as setting the system date and time) do not require saving the scratchpad to the active or startup configuration. For more information on managing configuration files and the use of the scratchpad, refer to "Maintaining Configuration Files" on page 3-2.

Changes made in Configuration mode require that you save them before they become part of the running configuration.

CLI Access Modes

The CLI has four different access modes, each of which provides the ability to perform the specific operations shown in Table 2-1.

Table 2-1. CLI Access Modes

Access Mode	Command Prompt	Description
Exec	Switch 8100fl>	Provides limited access to the system. Allows you to display status, perform diagnostic operations, and power slots on and off. You can also perform basic system-level tasks such as traceroute, launch ping requests, control terminal configuration, and logout. The Exec mode command prompt consists of the system name, followed by the angle brackets (>). For procedures on how to change the system name using the **hostname** command, refer to "Setting the Log in Banners" on page 4-5 .
Privileged Exec	Switch 8100fl#	Allows you to manage the system. Privileged Exec mode provides more facilities than Exec mode. For example, you can display critical features such as router configuration, access control lists and SNMP statistics. The Exec mode command prompt consists of the system name, followed by the pound sign (#). To enter this mode, enter the **enable** command from the Exec mode, then supply a password when prompted (if password protection has been configured). To exit Privileged Exec mode and return to Exec mode, type **disable** and press [**Enter**].
Configuration	Switch 8100fl (config)#	Allows you to configure all features and functions on the switch. These include switch configuration, access control lists, routing protocols, spanning tree configuration, and so on. To enter Configuration mode, first enter Privileged Exec mode (**enable** command or **en**), and then enter the **configure** or **config** command.
Boot	PMOM>	Certain tasks can be performed only from Boot mode. Enter the **reboot** command to reset the switch. If the switch still fails to boot, contact ProCurve Customer Support. To enter the Boot mode, boot the switch, and then interrupt the normal boot sequence by pressing the [**Esc**] key. (Use the spacebar to skip the countdown sequence). For information on how to upgrade the boot PMOM software and boot using the upgraded image, see "Upgrading Software" on page 3-14.

Notes

The command prompt shows the hostname in front of the mode character(s). The default name is "ProCurve 8108fl" or 'ProCurve 8116fl" according to model. To change the name, see "Setting the Host Name" on page 4-4.

When you are in Configuration mode, use the **exit** command or press [**Ctrl**][**z**] to exit to the previous mode. Typing **exit** in Privileged Exec mode will quit the session entirely (see "Terminating Sessions and Exiting Modes" on page 2-11).

Using the CLI

The CLI supports partial matching (also known as 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 be unique, the CLI understands what you are typing. If you enter enough characters of a command keyword to uniquely identify it and press the [**Tab**] key, the CLI will complete the command. For example, if you enter the following in Privileged Exec mode and then press the [**Tab**] key as indicated:

```
Switch 8100fl#show ru[Tab]
```

The CLI completes the command as follows:

```
Switch 8100fl#show running-config
```

If you do not enter enough characters or you enter the wrong characters, the CLI cannot complete the command. When you mis-enter command syntax, or enter syntax that the CLI does not recognize, the CLI will flag the syntax error with a ^ marker indicating the word where the error has occurred.

For example:

```
Switch 8100fl##show rum
                     ^
% Invalid input detected at '^' marker.
```

Use the CLI editing commands (see Table 2-2 on page 2-6) to correct the error.

CLI Editing Commands

The switch provides line editing capabilities to move forward or backward on a line, delete or transpose characters, and delete portions of a line. To use the line editing commands, you need a VT-100 terminal or terminal emulator. For more information on connecting a console and configuring a terminal, refer to the *Installation and Getting Started Guide* for your switch.

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 as detailed in the following table.

Table 2-2. CLI Line Editing Commands

Command	Resulting Action
[**Ctrl**] [**A**]	Move to beginning of line
[**Ctrl**] [**B**]	Move back one character
[**Ctrl**] [**C**]	Abort current line
[**Ctrl**] [**D**]	Delete character under cursor
[**Ctrl**] [**E**]	Move to end of line
[**Ctrl**] [**F**]	Move forward one character
[**Ctrl**] [**G**]	None
[**Ctrl**] [**H**]	Delete character just prior to the cursor
[**Ctrl**] [**I**]	Insert one space (tab substitution)
[**Ctrl**] [**J**]	Carriage return (executes command)
[**Ctrl**] [**K**]	Delete characters from cursor to end of line
[**Ctrl**] [**L**]	Refresh current line
[**Ctrl**] [**M**]	Carriage return (executes command)
[**Ctrl**] [**N**]	Next command from history buffer
[**Ctrl**] [**O**]	None
[**Ctrl**] [**P**]	Previous command from history buffer
[**Ctrl**] [**Q**]	Resume processing command
[**Ctrl**] [**R**]	Refresh current line
[**Ctrl**] [**S**]	Stop processing command
[**Ctrl**] [**T**]	Transpose character under cursor with the character just prior to the cursor
[**Ctrl**] [**U**]	Delete line from the beginning of line to cursor
[**Ctrl**] [**V**]	Follow by Ctrl-character to enter the Ctrl character.
[**Ctrl**] [**W**]	Delete one word backwards
[**Ctrl**] [**X**]	Move forward one word

Table 2-2. CLI Line Editing Commands (Continued)

Command	Resulting Action
[**Ctrl**] [**Y**]	Paste back what was deleted by the previous Ctrl-k or Ctrl-w command. Text is pasted back at the cursor location
[**Ctrl**] [**Z**]	If inside a subsystem, it exits back to the top level. If in Privileged Exec mode, it exits back to Exec mode. If in Configuration mode, it exits back to Privileged Exec mode.
[**Esc**] [**D**]	Delete characters from cursor's current location to the first blank space.
[**Esc**] [**F**]	Move forward one word
[**Esc**] [**backspace**]	Delete backwards from cursor to the previous blank space (essentially a delete-word-backward command)
[**Tab**]	Attempts to complete command keyword.
"<string>"	Opaque strings may be specified using double quotes. This prevents interpretation of otherwise special CLI characters.

You can also use the left and right arrow keys to move the cursor to the left and right respectively. Use the up-arrow key to scroll backwards through the previous commands entered in the current mode. Use the down-arrow key to return to the most recently entered command in the current mode.

Note

The correct use of [**Ctrl**][**Z**] is to exit a mode only. Do not use [**Ctrl**][**Z**] while a command is being processed (for example, during a **show** command). If you wish to abort a command that is in process, use [**Ctrl**][**C**], or use [**Ctrl**][**S**] to stop processing the current command.

If [**Ctrl**][**S**] is used to suspend output, use [**Ctrl**][**Q**] to resume processing the command.

Scrolling Down a Line or a Screen

When viewing some commands, the output might be longer than your screen can display. In such cases, a **--more--** prompt appears at the bottom of the screen. To display the next line, press the [**Enter**] key. To scroll down one full screen, press the [**Spacebar**]. To return to the CLI prompt, press the [**q**] key.

CLI Parameter Types

The following table describes all the parameter types you can use with the CLI.

Table 2-3. CLI Parameter Types

Data Type	Description	Example
conditional	A numerical conditional expression. Special symbols are used to describe a numerical condition: > (greater than), < (less than) and != (not equal to).	<1024 or >2048 or !=4096
hexadecimal	A hexadecimal number	a7 or 0xa7
hostname	Hostname of an IP host	whistler or john-pc
hostname/IP	Hostname or IP address of a host	munich or 10.43.1.4
keyword	A keyword described in the list of acceptable keywords in the online help	aggregate or individual
interface name or IP address	Name of an interface or its IP address	int1 or 10.1.4.33
IP address	An IP address of the form < x.x.x.x. >	10.1.2.3
IP address/mask	A pair of IP address and mask values. Depending on the command, the mask may be a network mask or filtering mask. The mask can be described using the traditional IP address syntax (255.0.0.0) or a CIDR syntax (/8).	10.1.4.0/255.255.255.0 or 10.1.4.0/24
IP address list	A list of IP addresses separated by spaces but enclosed in quotes.	"10.1.4.4 10.1.5.5 10.1.6.6"
MAC address	A MAC address specified in the following forms: xx:xx:xx:xx:xx:xx or xxxxxx:xxxxxx or xxxxxx-xxxxxx	08:00:50:1a:2b:c3 or 080050:1a2bc3 or aabbcc-ddeeff
number	An integer number	100
numerical range	A number or a range of numbers to denote, for example, a single vlan or a range of vlans	50 or 70-100
port	A single port	interface ethernet 1/4, (or "int et 1/4"), or int gi 2/1, or int te 4/6
slot number	A list of one or more occupied card slots in the switch	1 or 7
string	A character string. To include spaces in a string, specify the entire string in double quotes (").	abc or "abc def"
URL	A Uniform Resource Locator. Both tftp and ftp URLs are supported, that is: TFTP: *tftp://host/pathname* FTP: *ftp://user:password@ip/pathname*	tftp://10.1.4.5/test/abc.txt

Setting CLI Parameters

The **terminal history** command specifies the number of commands that will be stored in the command history buffer. Commands stored in the buffer can be recalled without having to type the complete command again. When you hit the ↑ key, the CLI displays the commands that were entered, from the most recent.

To specify the number of commands stored in the command history buffer, enter the following command in Exec mode or Privileged Exec mode.

terminal history size <*num*>	Set the size of the command history buffer

Alternatively, you can display all the commands that were executed during a CLI session. To display the CLI commands, enter the following command.

show history	Display command history.

The CLI also provides commands for setting the terminal display. Use the following commands in Exec mode or Privileged Exec mode to set and display terminal settings.

Command	Task
terminal length <*num*>	Set the number of rows to be displayed (possible values range from 0 to 64)
terminal width <*num*>	Set the number of columns to be displayed (possible values range from 24 to 256)
show terminal	Display terminal settings
terminal history	Control command history display
terminal [no] **timestamp**	Print the system's time stamp for each line of display
[no] **terminal monitor**	Display debug output to the current line

Note

When setting the terminal length and width, you should select values that match your display window (or physical terminal) size. Selecting values outside the possible range, may cause problems with the system and the display.

When selecting terminal length, entering a value of 0 will disable the pager functionality. This will cause output that is longer than your display window to scroll off the display.

Getting Help with CLI Commands

Interactive help is available from CLI by entering the question mark (**?**) character at any time. The help is context-sensitive; the help provided is based on where you are in the command. For example, if you are at the Exec mode prompt, enter a question mark (**?**) as shown in the following example to list the commands available in Exec mode:

```
Switch 8100fl>?

 Exec commands:
  log          - Log all terminal input and output to a file
  ping         - Send echo messages
  show         - Show running system information
  telnet       - Open a Telnet connection to another host
  terminal     - Set terminal line parameter
  traceroute   - Trace route to destination
  ----------
  no           - Negate a command or set its defaults
  ----------
  enable       - Change privilege level (turn on privileged
commands)
  exit         - Exit from Exec mode
  logout       - Exit from Exec mode
```

You can also type the **?** character, or press the [**Tab**] key, while entering a command to see a description of the parameters or options that you can enter. Once the help information is displayed, the command line is redisplayed as before but without the **?** character. Continue asking the CLI for help in completing a command until you have fully qualified the command. At this point, CLI help will tell you to press [**Enter**] (the **<cr>** symbol stands for carriage return—which is the [**Enter**] key).

The following is an example of invoking help while entering a command:

```
Switch 8100fl(config)#router ?
  ospf   - Open Shortest Path First (OSPF)
  rip    - Routing Information Protocol (RIP)
Switch 8100fl(config)#router ospf ?
  <1..65535>  - Process ID
Switch 8100fl(config)#router ospf 2 ?
  <cr>
Switch 8100fl(config)#router ospf 2
```

Utilities and Conventions

Take note of the following commands or conventions when using the CLI.

Search Command

The **search** <WORD> command can locate strings that appear in your running configuration file. Strings can be words like vlan, or numbers such as 10.20.30.40, or even a Perl-style regular expression.

Although the search command accepts most special characters used in regular expressions, the character "?" is interpreted as a request for Help. That is, instead of accepting "?" as a search parameter, the CLI will try either to complete the command or to provide help for succeeding tokens in the command.

Entering Parameters

When the CLI asks for a string parameter that appears in uppercase (for example, WORD), enter a variable (for example, hello). When the CLI prompts you for a parameter in lowercase, (for example, name), enter it as it appears (that is, name). It is command syntax.

Address Notation

When you enter an IP address and subnet mask, you can enter it as either an IP address and subnet mask pair or as an IP address with CIDR notation. For example, the IP address and mask 15.127.43.21 255.255.255.0, can also be entered as 15.127.43.21/24.

The switch supports subnet masking as well as inverse masks. For example, both 255.255.0.0 and 0.0.255.255 are both valid masks.

Terminating Sessions and Exiting Modes

The switch includes the following commands that terminate your current mode level: **quit**, **end** (or [**Ctrl**][**Z**]), **exit**, and **logout**. Figure 2-1 illustrates the actions of each of these commands.

- The **logout** command is only available in Exec and Privileged Exec modes. Use it and the **quit** (a hidden command) or **exit** commands to disconnect the SSH or Telnet session. If you are connected directly to the console, then using the **quit**, **exit** and **logout** commands will end your session but not disconnect you.
- The **exit** command in any sub mode of the global config mode, returns the CLI to the previous mode.

- The **end** command when used at any level of the Configuration mode, returns the CLI to the Privileged Exec mode.

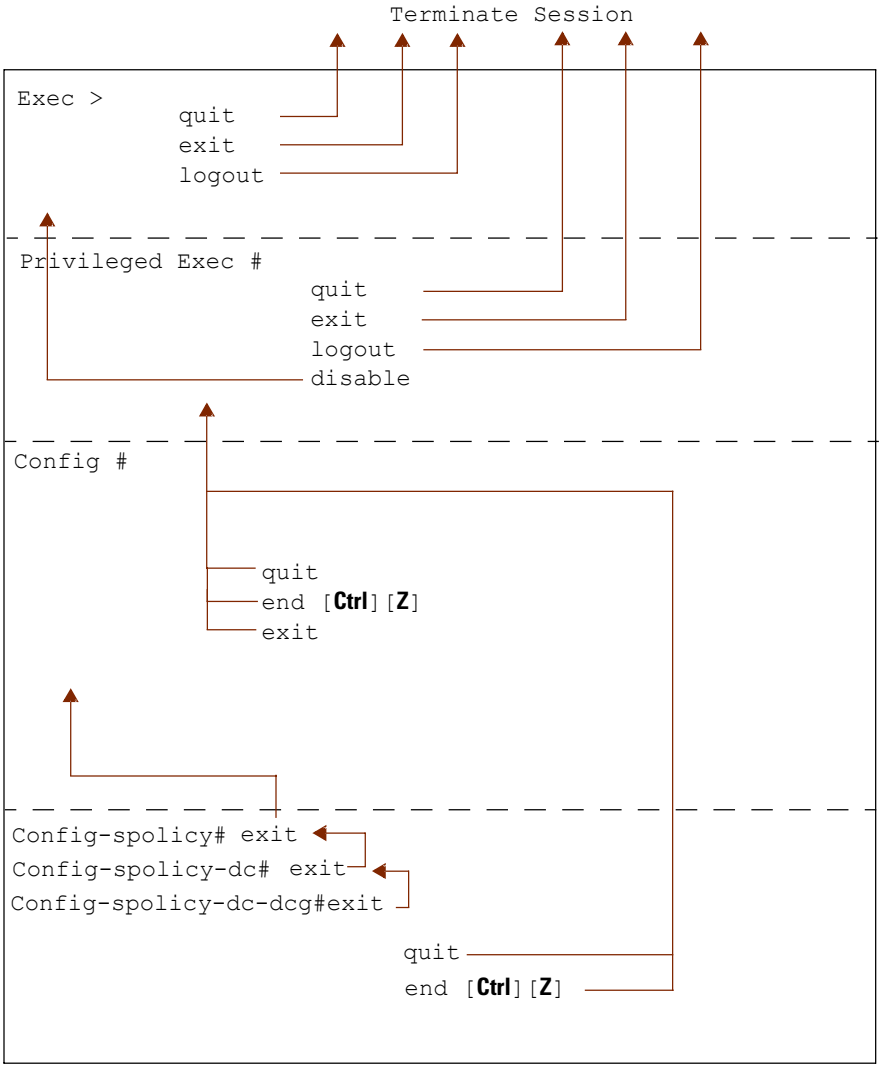


Figure 2-1. Using Terminating Commands

File and System Management

Contents

Maintaining Configuration Files

Note

Configuration commands you enter during a CLI session are stored in a temporary location called a *scratchpad*, and do not get entered into the running configuration until you perform a **save active** command.

The ProCurve 8100fl switches maintain in memory and on disk the following configuration files and commands:

scratchpad—The configuration commands you have entered during a CLI session.

- Each session generates its own unique scratchpad.
- Scratchpad commands are temporary and do not become active until you explicitly merge them with the active configuration by saving them. To apply the contents of your scratchpad, enter: **save active**.
- Saving one scratchpad has no effect on the scratchpads attached to other sessions on the same system.
- When you terminate your session, the contents of your scratchpad are deleted, again without effecting the contents of other scratchpads.

active or **running-config**—The running-config file includes both the startup-config file plus any configuration changes or additions that you have made active from the scratchpad.

Note

The terms *active* and *running-config* refer to the same thing. In this guide, you can substitute one for the other.

- The active configuration remains in effect until you power down or reboot the system.
- A reboot deletes the current running-config file and replaces it with a copy of the startup-config file.

Caution

The active configuration remains in effect only during the current power cycle. If you power off or reboot the switch without saving the active configuration changes to the startup configuration file, the changes are discarded.

startup-config—The configuration file the switch uses to configure itself when the system is powered on.

- The startup-config remains unchanged even when the system reboots.
- The ProCurve 8100fl switches ship with a factory-default startup-config file.

Saving Configuration Changes

Configuration commands you enter during a CLI session are stored in a temporary location called a “scratchpad”, and do not get entered into the running configuration until you perform a **save active** command. Figure 3-1 illustrates the configuration files and the commands you can use to save your configuration changes:

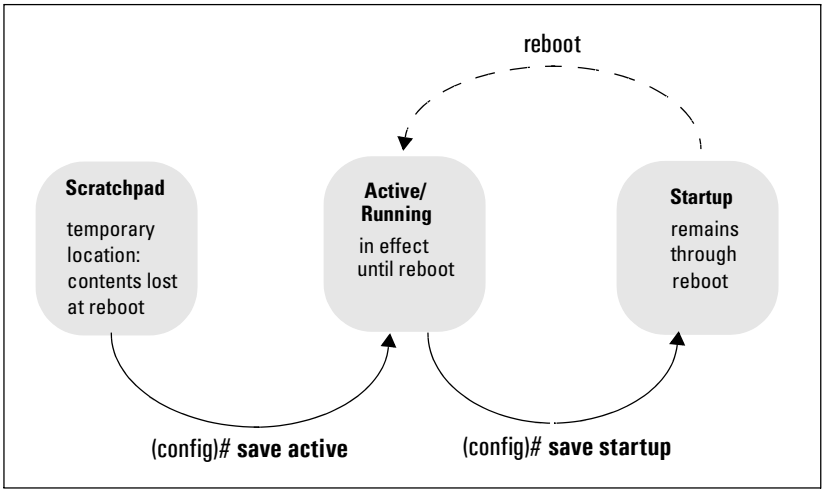


Figure 3-1. Commands to save configurations

Using the Scratchpad

Because some commands depend on other commands for successful execution, the scratchpad simplifies system configuration by allowing you to enter configuration commands in any order, even when dependencies exist. When you activate the commands in the scratchpad, the switch sorts out the dependencies and executes the commands in the proper sequence.

Saving Commands in Scratchpad to the Active Configuration

Use the following procedure to save the configuration commands in the scratchpad to the active configuration.

1. From the initial command prompt, ensure that you are in Privileged Exec mode by entering the **enable** command in the CLI.
2. Enter the **configure** command to access Configuration mode in the CLI.

3. Enter the configuration commands. They will be stored temporarily in the scratchpad where they can be viewed, edited, or saved to the running configuration.
4. Enter the **save active** command as shown:

```
Switch 8100fl(config)#save active
```

Note

The **save active** command performs a merge of the scratchpad with the running configuration.

Saving the Active Configuration to the Startup Configuration

After you save the configuration commands in the scratchpad into the running-config, the management module executes the commands and makes the corresponding configuration changes to the switch's active configuration. However, if you power off or reboot the switch, the unsaved changes are lost.

To save changes in the running-config into the startup configuration file (so that the switch reinstates the changes next time you reboot the software), use one of the following methods.

Replace the startup-config file with whatever is in the running configuration by entering the **save startup** or **write memory** commands.

From Privileged Exec mode in the CLI, enter the following command to copy the configuration changes in the active configuration to the startup configuration:

```
Switch 8100fl#copy active startup-config
```

The new configuration changes replace the startup-config file stored in the management module's boot flash.

Changing Configuration Information

The switch provides many commands for changing configuration information as shown in Table 3-1. For example, to "disable" a feature which has been enabled, use the **negate** command (followed by **save active**) on the relevant line of the active configuration.

Table 3-1. Commands to change configuration information

Mode	Command	Action
Privileged Exec	**copy** *<source> <destination>*	Copy between scratchpad, active configuration, startup configuration, TFTP server, or URL.
Configuration	**erase scratchpad**	Erase commands in scratchpad.
	erase startup-config	Erase startup configuration.
	negate *<line number>*	Negate one or more commands by line numbers.
	save active **save running-config**	Save scratchpad to active configuration.
	save startup-config **write memory**	Save active configuration to startup.

The following three examples show how to change configuration information under different usage scenarios:

Example 1.

If you do not want to save any of the configuration changes you have made:

From Configuration mode, enter the command:

```
Switch 8100fl(config)#erase scratchpad
```

Example 2.

If you decide you need selectively to erase some of the changes you have made to the running configuration:

1. From Configuration mode, enter the command:

   ```
   Switch 8100fl(config)#show active
   ```

   The configuration file is displayed with line numbers. (See "Viewing the Current Configuration" on page 3-7 for an example.)

2. Once you locate the line you want to delete and identify its line number, delete it by entering:

   ```
   Switch 8100fl(config)#negate <n>
   ```

   where *n* is the line number.

   Alternatively, delete a continuous range of line numbers by entering:

   ```
   Switch 8100fl(config)#negate <starting n>-<ending n>
   ```

3. When you enter **save active**, the selected line or lines are deleted from your active or running configuration.

Example 3.

If you need to start over:

1. Revert to the start up configuration by entering the following command:

   ```
   Switch 8100fl#copy startup scratchpad
   ```

2. Enter **save active**.

Note

The copy command can only be executed from Privileged Exec mode.

Displaying Configuration Information

The following table lists the commands that are useful for displaying the switch's configuration information. All commands can be run from either the Privileged Exec mode (`Switch 8100fl#`) or from the Configuration mode (`Switch 8100fl(config)#`).

Table 3-2. Commands to display configuration information

Command	Action
`show active`	Show running configuration of the system with line numbers
`show running-config`	Show running configuration of the system without line numbers
`show startup-config`	Show startup configuration of the system for the next reboot
`show scratchpad`	Show the non-activated configuration changes in the scratchpad

Viewing the Current Configuration

To view the current configuration as a numbered list showing each configuration command:

1. From the initial command prompt, enter the **enable** and **configure** commands to access Configuration mode in the CLI.
2. Enter the following command to display the status of each command line:

```
Switch 8100fl(config)#show active-config
```

The CLI displays the active configuration file as a numbered list showing each configuration command. For example:

```
Switch 8100fl(config)#show active
  !
 1 : interface Management0
 2 :   no shutdown
 3 :   ip address 10.10.1.45/21
!
 4 : ip telnet
 5 :   no shutdown
!
 6 : line vty 0
!
 7 : line vty 1
     !
     end
```

Note

You can also use the **show active** command in the Privileged Exec mode of the CLI. However, line numbers will not be displayed in this mode.

If there are errors, you will see the following message:

```
% Errored commands exist, type 'show errors' to review
```

To examine any errors, enter:

```
Switch 8100fl#show errors
```

To delete errors, enter:

```
Switch 8100fl#erase errors
Configuration errors erased
```

To re-apply the error commands, after fixing the conditions that triggered the errors, enter **save active**.

Note

An attempt to re-apply all commands in error-config will be made on each **save active** command.

Managing Files

The ProCurve 8100fl switches support two different memory storage devices located on the management module, typically used to store configuration files and system files. The storage device options on each management module include:

One 512 MB Internal Compact Flash Device

One removable PCMCIA Flash Device

Copy Command

Table 3-3 shows the local file systems supported by the **copy** command.

Table 3-3. Local File Systems

Local File System	Description
active or running-config	The current running configuration on the system. By copying to this file, the user merges the contents of the source file into the current running configuration. Users may only copy into **system:running-config** from **system:scratchpad**.
errors	This file contains the set of commands that failed execution during the last commit to the current running config. It is read-only.
history	History of commands used in the current session.
scratchpad	A temporary holding area where configuration commands are stored until users are ready to commit those commands into the current running configuration.
startup-config	The configuration saved on persistent storage that the system loads when restarted.

Note

Use the prefix **flash:** to locate the local flash storage area on the system.

The following URL types are supported by the **copy** command, allowing bi-directional file transfers both on and off the system:

ftp:—remote file accessed through the File Transfer Protocol (FTP)

tftp:—remote file accessed through the Trivial File Transfer Protocol (TFTP)

scp:—remote file accessed through the Secure Copy Protocol (SCP).

Note

If you use SCP, the file is encrypted when copied across the network and you will be prompted for a password. In addition, the remote host must be on the SSH known hosts list (see "Using SSH" on page 5-8 for details).

Table 3-4 shows syntaxes and examples for the various URL options used to perform remote file transfer.

Table 3-4. URL Syntaxes for Remote File Systems

URL	Syntaxes	Example
ftp:	ftp://[username[:password]@]location]/directory]/filename	ftp:remoteuser:secret@10.10.10.10/filename.txt
tftp:	tftp://location/directory/filename	tftp:10.10.10.10/filename.txt
scp:	scp://[username@]location]/directory/filename When using scp you will be prompted for a password	scp:remoteuser@10.10.10.10/filename.txt remoteuser@10.10.10.10's password: *******

For example, to do an FTP transfer you would need to provide both a username and password as follows:

copy ftp://remoteuser:secret@10.10.10.10/filename.txt flash:

File Management Commands

Because the ProCurve 8100fl switches allow a wide variety of file storage activities to a number of devices, it provides basic file management operations to manipulate these files. In these operations, wherever a "file" is specified, it may be specified as a URL indicating the file system. If you omit the file system, the present working directory is assumed. Use the file management commands in Privileged Exec mode to display, rename, and delete the configuration files stored on the primary management module. For a list of file management commands, see Table 3-5 on page 3-10.

Note

Relative file and directory names can be used for file operations. For example, if you are in the flash:/ directory and want to display a file named "history", you have the option to type any of the following commands: **more history**, or **more flash:/history**, or **more flash:history** to display the file in the CLI.

Table 3-5. File Management Commands

Command	Description	Syntax
cd	The **cd** command changes the present working directory of a user session from one file system to another. If *target-file-system* allows subdirectories, the **cd** command allows users to move into those subdirectories as well. Only file systems residing on hard drives may have subdirectories. The *target-file-system* must be either a logical or physical file system; it cannot be a remote file system.	`cd <target-file-system>`
pwd	The **pwd** command shows the user session's present working directory.	`pwd`

copy	The **copy** command uses a *url* to copy a *source-file* to a *target-file*, with the following conditions:
	• *url* can be used to specify any one of the following file transfer protocols: FTP, SCP, or TFTP. The *url* contains protocol, server, and the complete path.
	• *source-file* and *target-file* cannot be identical.
	• The *target-file* cannot specify a read-only file system.
	• Both the *source-file* and *target-file* cannot specify remote file systems.
		`copy <url> <source-file> <target-file>`
		
		or the inverse:
		
		`copy <source-file> <url> <target-file>`

delete	The **delete** command removes a *target-file* from the system. The *target-file* must reside on either a logical or physical file system that is not designated read-only; it cannot reference a remote file system.	`delete <target-file>`
dir	The **dir** command generates a listing of files located on *file-system*. *file-system* may be followed by a glob pattern to filter the listing for certain files. *file-system* must be either a logical or physical partition; it cannot be a remote file system.	`dir <file-system>`
erase	The **erase** command removes all file entries in *file-system* or any of the following configuration databases: scratchpad, error-config, or startup-config. The *file-system* may only be *flash:* .	`erase [ <file-system> | scratchpad | error-config | startup-config ]`

Table 3-5. File Management Commands (Continued)

Command	Description	Syntax
mkdir	The **mkdir** command creates a directory on a file system. *directory* must reference a file system based on a hard disk drive. If the directory already exists, no action is taken and a warning message is produced. If the directory is a subdirectory of another directory that does not exist, no action is taken and a warning message is produced.	`mkdir <directory>`
more	The **more** command displays the contents of the target file as paged output. By default, the system determines whether *target=file* is an ASCII text file or a binary file and displays the contents accordingly. The **`/binary`** option forces the file to be displayed as a binary file	`more [ /binary ] <target-file>`
rename	The **rename** command renames *source-file* to *target-file*. Both *source-file* and *target-file* must reside on logical or physical file systems; they cannot be on remote file systems. Both *source-file* and *target-file* must be writable.	`rename <source-file> <target-file>`
rmdir	The **rmdir** command deletes a directory off of a physical file system. If *directory* exists, it will be removed. If it does not exist, an appropriate failure message notifies the user. Everything contained in the directory will be removed as well.	`rmdir <directory>`

Backing Up and Restoring Files

ProCurve Networking recommends that you backup your system image and configuration files.

Backing Up System Files

To back up your system:

For local backups, enter the **copy** command and specify the file to be backed up and destination path and filename.

For remote backups, enter the **copy** <ftp | scp | tftp> command and specify the file URL and the destination path.

Note

To handle large images and for maximum reliability, it is recommended that you use **copy ftp** rather than **copy tftp** for all file and image downloads. For secure image transfer, you can also use the **copy scp** command.

To restore your system from a backup file, simply reverse the steps above (**copy** command for local restore and **copy** <ftp | scp | tftp> for remote restore).

Backing Up and Restoring Configuration Files

You can back up active, running-config, and startup configuration files to a local directory or to a remote host.

In case of a system failure, you can restore your system as you have it configured. In a worst case scenario of a system failure, you can download a new default startup configuration file from the ProCurve Networking web site at **http://www.procurve.com**.

Backing Up Startup Configuration

When you save the startup configuration file, the switch stores it in three places: in the boot flash and the PC card of the primary management module, and if there is a redundant management module, in its PC flash card as well. It is recommended that you store a backup of the startup configuration file on a central server.

To store a backup copy of the startup configuration file on to a server, use the **copy** command in Privileged Exec mode:

```
Switch 8100fl#copy startup |<filename>|<url>
```

To make a local backup in the management module, specify the following command in Privileged Exec mode:

```
Switch 8100fl#copy startup flash:startup.bak
```

where *startup.bak* represents the chosen filename for your local backup.

If the startup file is accidentally overwritten, the switch uses its default configuration. You can then use the copy command to overwrite the corrupted startup file with the backup file, as in the following example:

```
Switch 8100fl#copy flash:startup.bak startup
```

Managing System Devices and Software

Refer to the following procedures to upgrade system software and manage modules on the switch.

Determining Software Versions

To display the software versions running on the ProCurve Switch 8100fl, enter the **show version** command:

```
Switch 8100fl#show version [{management-module slot |
fabric-module slot | interface-module slot }]
```

You can display summary version information for all installed modules, or you can specify parameters to show more detailed information by slot.

The following example shows how to display version information for all installed modules (the default) by entering the show version command without specifying any additional parameters.

```
Switch 8100fl#show version

ProCurve Networking Switch 8100fl Series System Software
Version 1.0.0.0 Copyright (c) 1998-2005 by ProCurve
Networking.
Compiled on Sat Feb 12 19:09:34 GMT 2005 Bootloader Version
CM 1.0.22
Switch uptime is 1 minute, 56 seconds System restarted by
cold reset System image file is ms-1.0.0.0.ver

Switch 8100fl#
```

Upgrading Software

For easy software image management, the ProCurve Switch 8100fl supports the download and upload of software images between the compact flash on the management module and a server on the network (see "Backing Up and Restoring Files" on page 3-11).

To update the installed software or firmware on the switch, enter the **image** command at the Privileged Exec level of the CLI:

```
Switch 8100fl#image install <imagename> [chassis |
{management-module slot | fabric-module slot | interface-
module slot }]
```

where

imagename is the path and/or filename for the software distribution file.

Note

When you install a new image, it will automatically be placed in the opposite flash bank to the one currently in use. So if you are running in bank-A, it will be placed in bank-B and vice versa.

Rebooting

You can use boot commands to immediately initiate software boots from a software image stored in primary or secondary flash on the switch.

By default, the switch attempts to boot from the image stored in its primary flash, then its secondary flash, then a TFTP server.

To stop the system and force an immediate restart, enter the **reload** command at the Privileged Exec level of the CLI:

```
Switch 8100fl#reload [chassis | {management-module slot |
fabric-module slot | interface-module slot }] [soft | hard]
[reason]
```

To specify the boot image to use when rebooting the switch, enter the **boot system** command at the Privileged Exec level of the CLI:

```
Switch 8100fl#boot system [chassis | {management-module
slot | fabric-module slot | interface-module slot }] {bank-
A | bank-B}
```

Managing Modules

To control the power and administrative states of modules on the switch, enter the **set module** command from Configuration mode in the CLI:

```
Switch 8100fl(config)#set module {enable | disable}
{management-module slot | fabric-module slot | interface-
module slot }
Switch 8100fl(config)#set module {poweron | poweroff}
{fabric-module slot | interface-module slot}
```

where

enable sets the administrative state up,
disable sets the administrative state down,
poweron turns on power to the module, and
poweroff turns off the power to the module.

The following example shows how to power up the interface module in slot 3:

```
Switch 8100fl(config)#set module poweron interface 3
```

Management Modules and File Management

Management module redundancy adds another layer of complexity to file management. You may want to have the standby module to mirror the content on the active module so that the system can survive a switchover with as little change in environment as possible.

In normal operation, whenever changes are made to the primary management module's startup configuration file, the changes are copied to the redundant management module's configuration file. In this way, if the primary management module fails, the secondary module has the configuration information necessary to take over as the primary.

Note

The file management commands apply only to the primary Management Module. You cannot display, delete, or rename files in the backup Management Module.

Replacing Modules and Redundancy

If the primary management module fails, the redundant management module will reboot the fabric module and interface modules. During this time, service will be affected.

You can gracefully stop a management module or fabric module and cause the redundant module to take over by using the following command in Privileged Exec mode:

```
Switch 8100fl#halt <module>
```

Alternatively, you can power down the fabric module slot by issuing the following command:

```
Switch 8100fl#power down <fabric-module slot>
```

Note

When you power down a slot, the system turns power off to that slot and will keep power off until you enter the **power up** command. This is true even if you reboot the system.

Show Module Status

To see the status of all installed modules on the switch (management, fabric, and interface), enter the **show modules all** command.

To view the active or standby status of redundant management or fabric modules, enter the **show redundancy** command. This displays the following types of information:

```
Switch 8100fl#show redundancy
Slot   Module-Type          Model          State    Switch
-------------------------------------------------------------
MM-A   Management Module    fl Mgmt        active   auto
MM-B   Management Module    No module      standby  auto
FM-A   Fabric Module        8108fl Fabric  active   auto
MM-B   Management Module    No module      standby  auto
```

Switching Over Redundant Modules

To switch the system to the backup management or fabric module, enter the **redundancy switchover** command at the Privileged Exec level of the CLI:

```
Switch 8100fl#redundancy switchover {management-module |
fabric-module} [manual | force | lock | clear]]
```

Configuring Basic Features

Contents

Overview

This chapter describes how to configure basic, non-protocol features on the ProCurve Switch 8100fl using the CLI. The switch is configured at the factory with default parameters that allow you to use basic features of the system immediately. However, many of the advanced features such as VLANs or routing protocols must be enabled at the system (global) level before they can be configured.

Configuring Basic System Information

The first, and only essential, task in basic system configuration is assigning the IP address to the management port. This is the IP address that you use to access the ProCurve Switch 8100fl on the network.

Follow the procedures in this section to set the following system information:

- IP address for the management port on the management module
- System time and date
- System name
- System location
- Contact name (the person to contact regarding this switch)
- Log in banners
- Telnet

Note

You must use the **save active** command to save any changes in system information to the active or running configuration. The active configuration remains in effect only during the current power cycle. If you power off or reboot the switch without saving the active configuration changes to the Startup configuration file, the changes are lost. For more information, see "Saving Configuration Changes" on page 3-3.

Setting the Management Module IP Address

To set the system IP address and enable Telnet access to the switch:

1. Connect a serial console to the switch that uses VT-100 emulation.
2. At the command line prompt, enter the **enable** command to get to Privileged Exec mode in the CLI.
3. From the Privileged Exec mode, enter the **configure** command to get to Configuration mode in the CLI.
4. From Configuration mode, enter the following command to access the management interface:

   ```
   Switch 8100fl(config)#interface management 0
   ```

5. From Interface Management mode, enter the following command:

   ```
   Switch 8100fl(config-interface-management)#ip address
   <ipaddr> <mask>
   ```

   where <*ipaddr*> is the IPv4 unicast address and <*mask*> is the subnet mask you assign to the switch. You can specify the subnet mask in terms of mask bits. For example:

   ```
   ip address 10.10.1.45 255.255.248.0
   ```

   You can also specify the mask length. For example:

   ```
   ip address 10.10.1.45/21
   ```

6. Enable the management port, by entering the following command:

   ```
   Switch 8100fl(config-interface-management)#no shutdown
   ```

   Here is an example, showing all of the preceding steps:

   ```
   Switch 8100fl>enable
   Switch 8100fl#configure
   Switch 8100fl(config)#interface management0
   Switch 8100fl(config-interface-management)#ip address
   10.10.1.45 255.255.248.0
   Switch 8100fl(config-interface-management)#no shutdown
   ```

7. From Configuration mode, enter the following commands to enable telnet on the switch:

   ```
   Switch 8100fl(config)#ip telnet
   Switch 8100fl(config-telnet)#no shutdown
   ```

8. Enable a terminal line to the switch by entering:

```
Switch 8100fl(config)#line vty 0
```

where *0* represents the vty terminal line connection.

Note

The ProCurve Switch 8100fl supports a maximum of ten incoming remote vty connections (0 through 9), plus one console connection.

9. Enter the **save active** command to save any changes in system information to the active or running configuration, and then enter **save startup** to save the active configuration to the startup configuration.

Setting the System Date and Time

To set the system date and time:

1. From the Privileged Exec mode (#) prompt, enter the **clock set** command:

```
Switch 8100fl#clock set <HH:MM:SS>□<1...31>□<month>□<year>□
```

2. To verify your settings, enter the **show clock** command:

```
Switch 8100fl#show clock
```

The following example shows how to set the clock to 10:45:29 AM on June 30, 2005, and then verify the settings:

```
Switch 8100fl#clock set 10:45:29 30 june 2005
Switch 8100fl#show clock
*10:45:29 UTC Tues June30 2005
```

Setting System Parameters

The following system parameters and commands can be entered only from Configuration mode in the CLI.

Setting the Host Name

To set the system name, enter the following command:

```
Switch 8100fl(config)#hostname
```

For example, to enter the hostname “PNB8108”, type the following:

```
Switch 8100fl(config)#hostname PNB8108
```

Setting System ID, Location, and Contact

To assign a chassis ID, a physical location, and a contact person to the system using snmp (Simple Network Management Protocol), enter the following commands:

```
Switch 8100fl(config)#snmp-server location <string>
Switch 8100fl(config)#snmp-server chassis-id <string>
Switch 8100fl(config)#snmp-server contact <string>
```

where <*string*> represents the text you enter to specify the location, chassis identity, and contact support information for the switch.

Setting the Log in Banners

When a user connects to the switch, they will encounter banners (if they have been configured) in the following order:

1. message of the day (MOTD) banner - displays whenever a user connects to the switch using either a serial or telnet connection.
2. login banner - displays during login if a password has been defined for the telnet line that they are connecting on.

To create a message of the day (MOTD), use the following command:

```
Switch 8100fl(config)#banner motd "message"
```

To create a login banner, use the following command:

```
Switch 8100fl(config)#banner login "login"
```

If the switch is configured for aaa authentication (see "Configuring Login Prompts" on page 5-13), you can then configure authentication and authentication failure banners with the following commands.

To create an authentication banner, enter:

```
Switch 8100fl(config)#aaa authentication banner "banner"
```

To create an authentication failure banner, enter:

```
Switch 8100fl(config)#aaa authentication fail-message "fail"
```

Configuring Terminal Services

The Series 8100fl Switch supports up to 10 concurrent Telnet sessions (numbered from 0 through 9) for a maximum of ten incoming remote connections.

Use the shutdown command to terminate the Telnet service. The following example shows how to terminate a Telnet connection:

```
Switch 8100fl(config)#ip telnet
Switch 8100fl(config-telnet)#
Switch 8100fl(config-telnet)#shutdown
```

Establishing a Telnet Connection

To open a Telnet connection to a specified host, from Exec or Privileged Exec mode enter the **telnet** command:

```
Switch 8100fl#telnet <target> [port | service] [/vrf vrfname]
```

For example, to create a telnet session to another device for which the ip address is 10.10.1.24, enter:

```
Switch 8100fl#telnet 10.10.1.24
```

Configuring Terminal Line Parameters

The Series 8100fl Switch supports multiple lines (up to 10 vty connections, plus a console connection), and provides some flexibility in configuring parameters for each connection. From Privileged Exec mode, type the **configure** command and then the terminal line number to enter Terminal Line Configuration mode. When in this mode, you can configure various terminal line parameters for the switch during operation.

For example, enter the following commands to turn on the Exec level banner:

```
Switch 8100fl#config
Switch 8100fl(config)#line vty 0
Switch 8100fl(config-line)#exec-banner
```

where **line vty 0** corresponds to the terminal line that you wish to configure.

Use similar steps to configure the following terminal line parameters:

To limit the amount of time (in minutes) that a session on this line can remain connected without activity, enter:

```
Switch 8100fl(config-line)#exec-timeout <value>
```

where *value* is an integer designating the timer inactivity in minutes (enter a value of *0* to set an unlimited time for each session).

To configure the system to perform local password checking on this line, enter:

```
Switch 8100fl(config-line)#login-authentication
<method-list>
```

Note

To use the default authentication list (instead of a named method list), enter **login authentication default**.

To configure authorization for this line:.

```
Switch 8100fl(config-line)#authorization {commands level
| exec} method-list
```

To configure a password that users must enter to gain access to this line:

```
Switch 8100fl(config-line)#password {[0] cleartext-pw |
5 hidden-pw}
```

where 0 indicates a permanently unencrypted password, 5 indicates an encrypted password.

Note

For more information on setting up prompts and passwords for your switch, see "Configuring Passwords" on page 5-2.

Saving and Using the New Configuration

1. To activate the system commands entered in the previous steps, use the following command:

```
Switch 8100fl#save running-config
```

The CLI displays the following message:

```
Switch 8100fl(config)#save running-config
Please wait, acquiring configuration lock...done in 0.00
seconds
Now activating configuration changes, status report will
be returned shortly.
Switch 8100fl(config)#
0w0d: %SYS-7-CONFIG_RESULTS: 0 fail in 0 seconds

Switch 8100fl(config)#
```

2. To display the active configuration, enter the following command:

```
Switch 8100fl(config)#show active
```

Here is an example of displaying the active configuration:

```
Switch 8100fl(config)#show active
 1 : interface GigabitEthernet1/1
 2 :   no shutdown
 3 :   ip address 10.16.100.2/30
 4 : interface GigabitEthernet1/2
 5 : interface GigabitEthernet1/3
 6 : interface GigabitEthernet1/4
 7 : interface Loopback0
 8 :   ip address 10.16.130.4/32
 9 : ProCurve 8100fl ospf 1
10 :   network 10.16.0.0 0.0.255.255 area 10.16.0.0
11 : ip route 10.200.0.0/16 10.203.255.254
12 : ip route 10.201.0.0/16 10.203.255.254
     !
     end
Switch 8100fl(config)#
```

Note

For more details on interacting with configuration files, see "Maintaining Configuration Files" on page 3-2.

Configuring Port Parameters

Changes to port parameters are made using the **interface** <*port*> command to adjust to attached devices or other network requirements. Follow the procedures in this section to set the following port parameters:

Activating or disabling ports

Modifying port speed

Modifying port mode

Disabling or re-enabling flow control

Assigning a description to an interface

Note

To modify Layer 2, Layer 3, or Layer 4 features on a port, see the appropriate section in other chapters. For example, to modify Spanning Tree Protocol (STP) parameters for a port, see Chapter 12, "Bridging Configuration".

Specifying Slot and Port Numbers

The term port refers to a physical connector on an interface module installed in the switch. At the CLI, each port is referred to in the following manner:

<*type*> <*slot-number*>/<*port-number*>

where:

<*type*> is the type of module and/or the type of logical or physical interface, and is one of the following:

- **ethernet** – Ethernet (IEEE 802.3[z])
- **gigabitethernet** – GigabitEthernet (IEEE 802.3z)
- **lag** – Link aggregation interface
- **loopback** – Loopback interface
- **management** – Management interface
- **null** – Null interlace
- **tengigabitethernet** – 10 GigabitEthernet (IEEE 802.3ae)
- **vlan** – VLAN interface

<*slot-number*> is determined by the switch model and the physical slot in which the module is installed (see "Slot Numbering" on page 4-10).

<*port-number*> is the number assigned to the physical connector on the interface module. The range and assignment of port numbers varies by the type of module. Ports are numbered 1 to *n*, top to bottom.

For example, the port name **gi 3/2** refers to a port on the Gigabit Ethernet interface module located in slot 3, port 2.

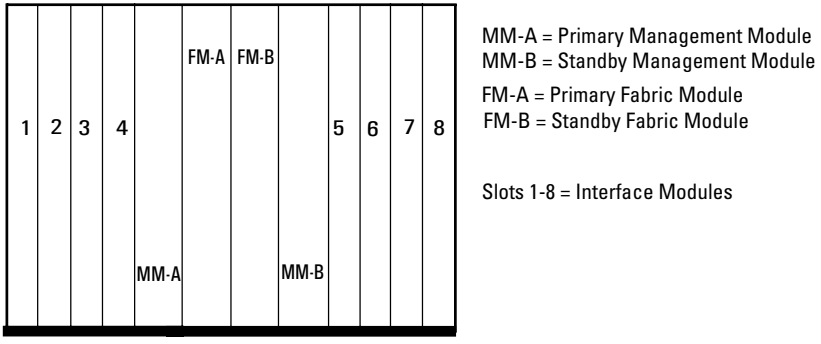
Note

You can build a configuration for a module that is not yet installed. For example, although slot 2 is empty, you can configure an interface **gi 2/1** and add an IP address to it, and then save the configuration for later use. Similarly, a 10 gigabit interface (**te 2/1**) could be configured for the empty slot in the same way.

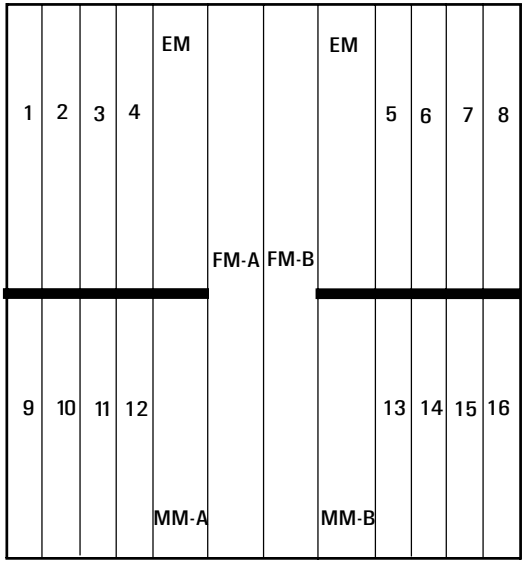
Slot Numbering

To configure or address individual modules and ports, you must know the corresponding slot name or number. Slot numbering is determined by the switch model and the physical slot into which an interface module is installed .

Switch 8108fl Slot Numbering. The Switch 8108fl chassis contains 8 slots for interface modules, 2 slots for management modules, and 2 slots for fabric modules. The following illustration shows the numbering and naming scheme used to identify each slot in the CLI.



Switch 8116fl Slot Numbering. The Switch 8116fl chassis contains 16 numbered slots for interface modules, 2 slots for management modules, 2 slots for fabric modules, and 2 expansion slots. The following illustration shows the numbering and naming scheme used to identify each slot in the CLI.



Activating or Disabling Ports

The **shutdown** and **no shutdown** commands allows you to enable and disable specified ports. The default value for an interface is disabled (**shutdown**).

For example, to activate the ethernet port 8 in slot 1, enter the following:

```
Switch 8100fl(config)#interface et 1/8
Switch 8100fl(config-if)#no shutdown
```

To disable this interface once activated, enter the following:

```
Switch 8100fl(config)#interface et 1/8
Switch 8100fl(config-if)#shutdown
```

Note

By default interfaces are shutdown initially. Use the **no shutdown** command to activate them once they have been created.

Modifying Port Speed

The ProCurve Switch 8100fl ports are designed to auto-negotiate the speed and mode of the connected device. If the attached device does not support auto-negotiation, you can manually enter the port speed to operate at either 100 Mbps or 1000 Mbps.

To modify a port speed, enter the **speed** command from the interface configuration mode. For example, to change the port speed of gigabitethernet port 5 in slot 2 to 100 Mbps, enter the following:

```
Switch 8100fl(config)#interface gi 2/5
Switch 8100fl(config-if)#speed fastethernet
```

The speed <*value*> that you enter can be one of the following:

- *auto* – sets the speed to auto-negotiate mode
- *fastethernet* – sets the speed to 100Mbps
- *gigabit* – sets the speed to 1000Mbps

Modifying Port Mode

The **mode** command can be used to control behavior for a specified interface. The two command options are **no mode** (the default) or **mode slave**, which sets the mac to act in slave mode. For example, to change the port speed of ethernet port 6 in slot 1 to act in slave mode, enter the following:

```
Switch 8100fl(config)#interface et 1/6
Switch 8100fl(config-if)#mode slave
```

Disabling or Re-enabling Flow Control

The **flowcontrol** and **no flowcontrol** commands allows you to configure ports on the switch to operate with or without flow control (802.3.x). Flow control is enabled by default.

For example, to disable flow control on ethernet port 8 in slot 1, enter the following:

```
Switch 8100fl(config)#interface et 1/8
Switch 8100fl(config-if)#no flowcontrol
```

To re-enable flow control:

```
Switch 8100fl(config)#interface et 1/8
Switch 8100fl(config-if)#flowcontrol
```

Note Flow control is enabled by default and is not reported in show interface configuration displays. The flow control state is only reported when it is disabled (**no flowcontrol**).

Assigning a Description

The **description line** command allows you to enter a text description for a specified interface.

For example, to label the gigabitethernet port 3 in slot 1 as **MAIN TRUNK**, enter the following:

```
Switch 8100fl(config)#interface gi 1/3
Switch 8100fl(config-if)#description line MAIN TRUNK
```

To remove this description:

```
Switch 8100fl(config)#interface gi 1/3
Switch 8100fl(config-if)#no description
```

Note You can assign a description to physical ports, virtual routing interfaces and loopback interfaces.

— This page is intentionally unused. —

Security Configuration

Contents

Overview

The ProCurve Switch 8100fl provides security features that help control access and filter traffic. Access to the switch can be controlled by:

- Terminal line password authentication
- Secure shell protocol (version 1 and 2, server and client)
- RADIUS
- TACACS+
- Local user names and passwords

Note

The ProCurve Switch 8100fl requires you to turn on access features that affect security. By default, these features are turned off.

Configuring Passwords

The switch provides password authentication for accessing the User and Privileged Exec modes. If TACACS+ or RADIUS is not enabled on the switch, only switch-level password authentication is performed (if configured).

Preventing Lock Outs

Caution

To avoid being locked out of the CLI when implementing password changes, note the following precautions:

- Verify parameter values by using the **show scratchpad** and **show active** commands before saving security commands to the active or startup configuration file on the switch. Any misconfiguration can effectively lock you out of the CLI.
- If you forget your line-level passwords, you can log on using the console and enter new passwords. Once you save the new passwords to the active configuration, other users who access these lines can use the new passwords. To make the changes permanent, save the active configuration to the startup configuration.

To test your configuration safely, leave your startup configuration unchanged. Add your planned changes to the running config, and then verify that you can log on safely before saving any changes to the startup config.

If your changes to the running config lock you out, you can (as a last resort) power cycle the switch to revert to your unchanged startup config.

If your changes to the startup config lock you out, refer to "Recovering from Forgotten Passwords" on page 5-6 for recovery procedures.

If the switch cannot reach the RADIUS server and there is no other authentication method configured, then you will be locked out of the CLI. In this case, one can configure TACACS+ and/or local authentication as a fallback so that when RADIUS authentication fails, the next available method is tried.

Specifying the CLI-level Password

To configure a switch enable password (that is, a Privileged Exec mode password), enter the following command in Configuration mode:

```
Switch 8100fl(config)#enable secret [encrypt|0|5] <string>
```

By default the password you enter will be encrypted to prevent it from being displayed in clear text in the configuration file output.

The optional encryption parameters [encrypt | 0 | 5] control password encryption in the following ways:

Specifying **encrypt** causes the switch to encrypt the clear text password (<string>). The encrypted password will appear in the configuration file as enable secret 5 <encrypted string>. This is the same as the default.

Specifying a **0** forces the switch to display the password entry as clear text in your configuration files.

Specifying a **5** indicates the string that follows has been encrypted.

Notes

When you enter the password string, type it as clear text. When you press [**Enter**], the scratchpad will perform the hash algorithm for you and produce the encrypted output

The enable password does not prevent users from accessing the switch—it restricts access to Exec mode only.

For example:

To create an encrypted password called “mysecretpassword” enter;

```
Switch 8100fl(config)#enable secret mysecretpassword
```

When you do **save active** and then enter **show running-config**, this will appear as a line in the running configuration as follows:

```
enable secret 5 $1$ZyK.$8NHx2DJBsiGQyhTBmUakz1
```

where *5* indicates that the password has been encrypted.

To allow passwords to be displayed in an unencrypted format, enter **0** before you enter the password.

For example:

To create a clear text password called “mypassword” enter;

```
Switch 8100fl(config)#enable secret 0 mypassword
```

When you save this setting, it will appear as a line in the running configuration as follows:

```
enable secret 0 mypassword
```

where *0* indicates that the password is not encrypted.

Caution

Test all new passwords before saving the active configuration to the Startup configuration file.

After password protecting access to the CLI, you can still access the switch without a password by connecting a terminal line configured without password protection. To password protect terminal lines from users connecting with Telnet or SSH, see “Specifying Line-level Passwords”.

Specifying Privilege Levels

The ProCurve 8100fl supports two levels of privileges to which you can assign passwords, level 0 and level 15:

Level **0** places users at the Exec mode and limits their access to the commands at this level.

Level **15** places users at the Privileged Exec mode and allows them full access to the CLI commands.

To configure a switch password and set the privilege level, enter the following command in Configuration mode:

```
Switch 8100fl(config)#enable secret level <lvl>
[encrypt|0|5] <string>
```

where

<lvl> is either 0 (Exec mode) or 15 (Privileged Exec mode); and
[0|5] can be either 0 (an unecrypted password) or 5 (hidden or encrypted)

For example, to set and encrypt a Privileged Exec mode password as abcd1234, you would enter the following command:

```
Switch 8100fl(config)#enable secret level 15 5 abcd1234
```

By default, the ProCurve 8100fl allocates users full access at privilege level 15.

To create a user with restricted access (Exec mode only), assign privilege level 0 by entering the following command in Configuration mode:

```
Switch 8100fl(config)#username <name> privilege <0>
password <string>
```

Note

Exec mode privileges provide only limited access to the system. It allows users to perform basic system-level tasks such as launch ping requests, show running system information, and set terminal line parameters; it does not allow the user to make any configuration changes.

Specifying Line-level Passwords

The ProCurve Switch 8100fl supports up to 10 vty remote connections for which you can assign line-level passwords. To prevent unauthorized access, it is recommended you assign line-level password protection to all the terminal lines that you configure.

To add password protection to each terminal (or console) line:

1. From Configuration mode, enter the following command:

   ```
   Switch 8100fl(config)#line vty 0
   ```

   where *0* represents the line number of the vty terminal line connection.

2. Enter the following command to configure a password to the line that you have specified:

```
Switch 8100fl(config-line)#password <password>
```

By default the password you enter will be encrypted to prevent it from being displayed in the configuration file output.

For example:

To create an encrypted password called “mysecretpassword” enter;

```
Switch 8100fl(config-line)#password mysecretpassword
```

When you do **save active** and then enter **show running-config**, this will appear as a line in the running configuration as follows:

```
password 5 $1$ZyK.$8NHx2DJBsiGQyhTBmUakz1
```

where *5* indicates that the password is encrypted as an MD5 hash.

To allow passwords to be displayed in an unencrypted format, enter **0** before you enter the password.

For example:

To create a clear text password called “mypassword” enter;

```
Switch 8100fl(config)#enable secret 0 mypassword
```

When you save this setting, it will appear as a line in the running configuration as follows:

```
enable secret 0 mypassword
```

where *0* indicates that the password is not encrypted.

Note

By default all passwords are encrypted, meaning that they will not display in readable text in the configuration files. To specify an unencrypted password, you must enter a **0** before entering the password text.

3. Repeat steps 1 and 2 to assign a line-level password to each terminal line that you have configured on the switch.

Recovering from Forgotten Passwords

To recover from a lost password, you can reboot the switch, bypass startup configuration processing, and then reset the password from Boot mode in the CLI. To do so, follow the steps below:

Note

The following procedure may only be performed via the serial console. Because this procedure allows passwords to be changed without actually logging onto the switch, physical security should be maintained at all times.

1. If you have two Management Modules installed, pull the backup module out of its slot so that only one Management Module is active.
2. Connect a serial console to the switch, and then perform a power cycle to reboot the switch.
3. Interrupt the normal bootup sequence—at the point where it says: "Press <Enter> to execute or <ESC> to abort"—by pressing the [**Esc**] key.

   This will put you into Boot mode—the prompt will change to **PMON>**.
4. From Boot mode, enter the following commands:

```
PMON> set STARTUP_MODE skip_config
PMON> boot
```

   The configuration process will then skip the startup configuration and the switch will boot up with the factory default settings.
5. At the reboot, enter a new password and save it to the active configuration using the **save active** command.

   If you have removed a backup management module, insert it back into its slot before saving the new settings to the startup configuration—this will synchronize both active and backup startup configurations to use the new password.
6. To make the changes permanent, save the new password setting to the startup configuration using the **save startup** command.

   The next time you reboot the system, you and other users will be able to use the new password for access and authentication.

Notes

When the boot environment variable STARTUP_MODE is set to "skip_config" this indicates the configuration process should skip loading the startup config while initializing. This variable is reset after use, so the next boot will again use the startup configuration if present.

You can use the same procedure to bypass a bad startup configuration or start the switch in factory-settings mode for troubleshooting purposes.

Using SSH

SSH provides more secure communications than using Telnet because connections are authenticated and communications over the network are encrypted. Secure shell (SSH) is a protocol based on OpenSSH that allows you to log in to a remote switch and execute commands on that system.

The switch provides both an SSH server and client. To configure the SSH server use the Configuration mode command **ip ssh** (and the SSH Configuration mode **no shutdown** command).

Both server and client support SSH version 1 and 2. If TACACS+ or RADIUS authentication is enabled on the switch, passwords are authenticated by the TACACS+ or RADIUS server. Private and public keys on a per-user basis are *not* supported.

Establishing SSH Sessions

The SSH Server parameters are:

```
Switch 8100fl(config)#ip ssh
Switch 8100fl(config-ssh)#?
Secure Shell Server commands:
  address     - Set address on which to accept Secure Shell connections
  ciphers     - Set ciphers to allow for Secure Shell protocol version 2
  macs        - Set Message Authentication Codes to allow for SSH protocol
version 2
  negate      - Delete a command from running-config, scratchpad, or
error-config
  port        - Set default port for Secure Shell
  power       - Power control various slots
  shutdown    - Stop Secure Shell service
  ssh         - Open a Secure Shell connection to another host
  version     - Set Secure Shell protocol versions to use
```

The SSH client parameters are:

```
Switch 8100fl(config)#ssh ?
  -1   - Force protocol version 1
  -2   - Force protocol version 2
  -c   - Specify encryption algorithm
  -e   - Specify escape character
  -l   - Specify a user name to log in as
  -m   - Specify MAC algorithm for protocol version 2
  -p   - Specify the port to connect to on the remote host
  WORD(1..1024) - Target address or hostname
```

Note

The **WORD** parameter must be the last parameter you enter. If you enter the **WORD** parameter as the second parameter for example, the CLI will not allow you to enter any of the other display parameters. Instead, the CLI will prompt you for an optional **LINE**, which you can use to enter a command string (like **show users**). This string will be run upon connection, you will see the output of this command. Then the connection will terminate. This is useful for situations where you just want information but do not want to maintain a connection. The **-e** option (specify an escape character) is useful as a toggle to other SSH sessions.

To establish SSH connections between a ProCurve Switch 8100fl host and any terminal, you must match the version level (typically version 2), port to use, encryption algorithm, and so on.

If a login password has been configured on the switch, you will be prompted for it. Because communications between the SSH client and server is now encrypted within the SSH session, the password cannot be read by other users on the network. You can use CLI commands in the SSH session as you normally would through a console or Telnet connection.

To end your SSH session, type **exit** (or **quit**, or **logout**). If necessary, repeat entering **exit** to disconnect your connection. You can also terminate your SSH session by typing **~.** (To end your console session, type **exit**, **quit**, or **logout**. The console terminates your Exec session and resets the terminal line.)

Monitoring SSH Sessions

The switch allows up to 10 simultaneous SSH (or Telnet) sessions. Use the **show users** command to see current Telnet and SSH users and session IDs.

Using SSH and Telnet Sessions

You can combine SSH connections with Telnet connections to reach your destination. Figure 5-1 shows different ways to mix secure and unsecure connections and the consequences experienced.

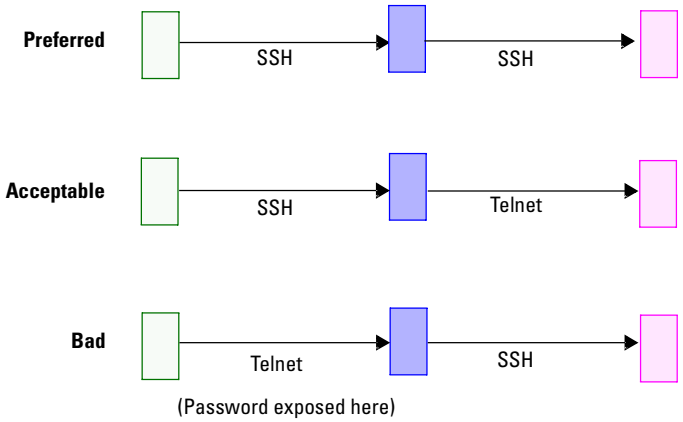


Figure 5-1. Security Considerations when Mixing Telnet and SSH Sessions

Configuring Authentication

You can configure authentication at the following levels:

Line

Enable mode

Local user

RADIUS/TACACS+ server groups

To configure the authentication lists for logging in, enter the following command in Configuration mode:

```
Switch 8100fl(config)#aaa authentication login <method
list> group <group name> line |local | enable |none
```

The parameters can be used as follows:

Method lists include individual lists you create or **default**. Specifying **default** is equivalent to entering **no aaa authentication login**.

Caution

Ignoring options after the *method list* option will apply the default behavior of denying access. This is useful if intended. Unintended use will result in being locked out of the system (that is, if you specify **line**, you effectively apply default values and preserve your ability to log into the system).

The *group* option allows you to use the default RADIUS or TACACS+ server group or to use a *group name* to specify a defined server group.

Specify **enable** to use the enable password.

Specify **line** to apply authentication to individual console or terminal line connections to the switch.

Specify **local** to enable users (configured with the **username** command) to access the system.

Specify **none** to bypass authentication entirely.

Configuring Authentication Method Lists

To define the authentication method list for Privileged Exec mode, enter the following command from Configuration mode:

```
Switch 8100fl(config)#aaa authentication enable default
[group {radius | "tacacs+" | <group name>}...[enable | line
| none]
```

The parameters can be used as follows:

The *group* option allows you to use the default RADIUS or TACACS+ server group.

The *group name* option allows you to specify a defined server group.

Specify **enable** to set up an authentication method list for Privileged Exec mode.

Specify **line** to use a line password.

Specify **none** to bypass authentication.

Configuring Authorization

To restrict network access for individual users, enter the following command in Configuration mode:

```
Switch 8100fl(config)#aaa authorization {commands <priv-
level> | exec} {default | listname} [group {radius | tacacs+
| group-name}]... [if-authenticated | none]
```

The parameters can be used as follows:

Specify a *command* and *privilege level* (see page 5-8) to check authorization for individual commands.

Specify **default** to apply a default authorization list—the *listname* option allows you to specify a defined authorization list.

The *group* and *group-name* options allow you to use defined RADIUS or TACACS+ server groups.

Specify **if-authenticated** to authorize if authenticated.

Specify **none** to bypass authentication.

Configuring Login Prompts

To configure the user login prompts for user name and password, enter the following commands in Configuration mode:

Table 5-1. Configuring Login Prompts

Command	Action
`Switch 8100fl(config)#aaa authentication username-prompt <prompt>`	Configure the login user name prompt
`Switch 8100fl(config)#aaa authentication password-prompt <prompt>`	Configure the login password prompt

To configure a banner to display prior to login, enter the following command in Configuration mode:

```
Switch 8100fl(config)#aaa authentication banner
<C_TEXT_C(0..1023)  - Banner text>
```

where *C_TEXT_C* means delimited text. Whatever character you first enter, will be interpreted as the delimiting text, that is, the character you must enter to terminate banner text entry. This convention allows you to enter multiple lines - totalling up to 1023 characters long. Without a convention to define the delimiting text, your banner text entry would terminate when you pressed the [**Enter**] key.

A common and useful delimiting character is the double quote (“). Note that the delimiting character does not print. For example, if you enter your banner text as:

“Welcome to the Acme ProCurve 8108fl”

your screen will display:

Welcome to the Acme ProCurve 8108fl

However, if you enter:

Welcome to the Acme ProCurve 8108fl

your screen will display:

elcome to the Acme ProCurve 8108fl

and you will have to enter a capital **W** to terminate banner text entry.

Configuring Accounting

Accounting collects data about user activity and system events and sends it to a server (or servers) when specified events occur on the switch such as a logoff or a reboot.

To provide accounting information for billing or security purposes, enter the following command in Configuration mode:

```
Switch 8100fl(config)#aaa accounting {{commands priv-level
| exec} {default | listname} | {system {info | warning |
error | fatal} | cfg-change {running-config | startup-
config}} default} {none | {start-stop | stop-only}
[broadcast] [group {radius | tacacs+ | group-name}]...}
```

where:

commands account for shell commands,
default or *listname* specifies the accounting list to be used,
system accounts for system event messages,
cfg-change accounts for changes to the system configuration,
broadcast sends records to multiple servers, and
group specifies the server group to be used.

The following example accounts for commands assigned to user-defined level 15 and specifies that accounting records be sent after the first acknowledgement to the default RADIUS server group.

```
Switch 8100fl(config)#aaa accounting commands 15 no-
broadcast group radius
```

Note Use the **no aaa accounting command** to disable aaa accounting.

Configuring RADIUS

You can secure Exec or Privileged Exec mode access to the switch by enabling a Remote Authentication Dial-In User Service (RADIUS) client. (See RFCs 2865 and 2866 for more information on RADIUS.) A RADIUS server responds to the switch RADIUS client to provide authentication.

You can configure multiple RADIUS server targets on the switch. You can configure a timeout value to tell the switch how long to wait for a response from RADIUS servers.

Note

The following list of commands contain parameters such as **deadtime** which are implemented globally. Some of the **radius-server** commands allow you to set the same parameter for a specific server. This specific setting overwrites the global setting. However, if you set this parameter at a server-group level, it overwrites both the individual server and global parameter setting.

To configure RADIUS security, enter the following commands in Configuration mode:

Table 5-2. Configuring RADIUS Security

Command	Action
radius-server challenge-noecho	Disable user input echoing to screen during an Access-Challenge
radius-server deadtime <minutes>	Set time that RADIUS server is ignored after it has failed. (That is, after a timeout has expired, and all the retransmits have been expended, do not try to contact this server for the specified amount of time.)
radius-server key [0 | 7]	Set shared secret key for RADIUS server.

Table 5-2. Configuring RADIUS Security (Continued)

Command	Action
radius-server host <server-options>	Uniquely define the host. Minimally, you can define an IP address or hostname, authentication port (default is 1812), and accounting port (default is 1813). If you specify authentication port or accounting port is 0, they will not be used. (You cannot specify that both authentication port and accounting ports be 0.)
radius-server source <address>	Sets the hostname or IP address of the RADIUS server to use for transactions.
radius-server timeout <seconds>	Set the maximum time to wait for a RADIUS server reply.
radius-server retransmit <number>	Set the number of retries to the active server.
aaa group server radius <group name>	Specify the name of the RADIUS server group (accesses the RADIUS server group mode). The <*group name*> parameter cannot be "**radius**," which is reserved for system use.

Monitoring RADIUS

To monitor RADIUS by showing server statistics, enter the **show radius servers** command in Privileged Exec mode.

The following example shows a sample configuration for two RADIUS servers:

```
Switch 8100fl(config)#radius-server host 172.2.100.1 auth-
port 1812 acct-port 1813 key hello
Switch 8100fl(config)#radius-server host 172.2.100.2 auth-
port 1200 acct-port 1201 key xyz

Switch 8100fl(config)#aaa group server radius MYGROUP

server 172.2.100.1 auth-port 1812 acct-port 1813
server 172.2.100.2 auth-port 1200 acct-port 1201
```

Configuring TACACS+

You can secure Exec or Privileged Exec mode access to the switch by enabling a TACACS+ client. A TACACS+ server responds to the switch TACACS+ client to provide authentication.

You can configure multiple TACACS+ server targets on the switch. You can configure a timeout value to tell the switch how long to wait for a response from TACACS+ servers.

To configure TACACS+ security, enter the following commands in Configuration mode:

Table 5-3. Configuring TACACS+ Security

Command	Action
tacacs-server deadtime <minutes>	Set time that TACACS+ server is ignored after it has failed.
tacacs-server key [0 | 7]	Set shared secret key for TACACS+ server.
tacacs-server host <server-options>	Uniquely defines a TACACS+ server. Minimally you must configure an IP address and a port number (which cannot be 0). Default port address is 49.
tacacs-server source <address>	Set the hostname or IP address of the TACACS+ server to use for transactions.
tacacs-server timeout <seconds>	Set the maximum time to wait for a TACACS+ server reply.
tacacs-server single-connect <number>	Limit the server to use one TCP connection. This feature allows multiple connections over a single connection as opposed to repeatedly building up and tearing down connections.
aaa group server tacacs+ <group name>	Specify the name of the TACACS+ server group (accesses the TACACS+ server group mode). The <*group name*> parameter cannot be "**radius**," which is reserved for system use.

Monitoring TACACS+

To monitor TACACS+ by showing server statistics, enter the **show tacacs servers** command in Privileged Exec mode.

The following example shows a configuration for two TACACS+ servers:

```
Switch 8100fl(config)#tacacs-server host 172.2.100.2 port
49 key testing123
Switch 8100fl(config)#tacacs-server host 172.2.100.1 port
49 key testing123

Switch 8100fl(config)#aaa group server tacacs+ MYTGROUP

server 172.2.100.1
server 172.2.100.2
```

6

RIP Configuration

Contents

Overview

This chapter describes how to configure the Routing Information Protocol (RIP) on the ProCurve Switch 8100fl. RIP is a distance-vector routing protocol for use in small networks. A router running RIP broadcasts updates at set intervals. Each update contains paired values where each pair consists of an IP network address and an integer distance to that network. RIP uses a hop count metric to measure the distance to a destination.

The ProCurve Switch 8100fl provides support for both RIP Version 1 and 2.

RIP1 is described in RFC 1058

RIP2 is described in RFC 1723

RIP Version 2 support enables the switch to implement plain text and MD5 authentication methods

The protocol-independent features that apply to RIP are described in Chapter 14, "IP Routing Configuration".

Configuring RIP on the Switch

By default, RIP is disabled on the switch and on each of the attached interfaces. All of the following procedures require you to be in Router Configuration mode. To enter this mode from Configuration mode, enter:

```
Switch 8100fl(config)#router rip
Switch 8100fl(config-router)#
```

To configure RIP on the switch, follow these steps:

1. Start the RIP process by entering the Configuration mode **router rip** command.
2. Use the Routing Configuration mode **network** command to enable routing on the specified interfaces that are within the IP networks.
3. Use the Interface Configuration mode **ip rip** command to configure interface-specific settings.

Enabling and Disabling RIP

To enable or disable RIP on the switch, enter one of the following commands in Configuration mode:

To enable RIP, enter **router rip**.

To disable RIP, enter **no router rip**.

Enabling Routing on a Network

To allow routing on a network, you must enter:

```
Switch 8100fl(config-router)#network <IP address>
```

Summarizing Routes

RIP version 2 supports the automatic summarization of routes. Route summarization enables the switch to collect boundary subprefixes when traversing (classful) network boundaries. If your network includes disconnected subnets, you can disable route summarization to force the switch to include subnet and host routing information when sending traffic across network boundaries.

To enable route summarization (which is on by default), enter:

```
Switch 8100fl(config-router)#auto-summary
```

To disable route summarization, enter:

```
Switch 8100fl(config-router)#no auto-summary
```

Distributing Default Information

To enable the distribution of default information, enter:

```
Switch 8100fl(config-router)#default-information originate
```

Setting Default Metrics

To set the default metric of distributed routes, enter:

```
Switch 8100fl(config-router)#default-metric <number>
```

Defining Administrative Distance

The administrative distance is a metric used to determine the best path to use when more than one route to the same destination exists but in different routing protocols. For the ProCurve Switch 8100fl, administrative distance is a number between 0 and 255. The lower the value, the more preferable the route.

To specify the size of the administrative distance, enter:

```
Switch 8100fl(config-router)#distance <number>
```

To revert to the default distance (this is, 100), enter:

```
Switch 8100fl(config-router)#no distance
```

Filtering Updates

You can manage the amount of update information the ProCurve 8100fl receives by filtering routing updates by source gateway, by prefix, and by access list.

For filtering on access lists, you must further specify the direction (in or out) and the interface. For filtering on a prefix, you must specify the gateway, the direction, and the interface. For filtering on a gateway, you just need to specify the direction and the interface affected.

Limiting Updates

To suppress updates on a specified interface, enter:

```
Switch 8100fl(config-router)#passive-interface <interface>
```

Limiting Paths

Your RIP routing table can track up to four paths to another router. You can set that number to as low as one path. To limit the number of connections your routing tables will maintain to any one IP address, enter:

```
Switch 8100fl(config-router)#maximum-paths <number>
```

Filtering Networks in Updates

To filter the networks that you receive in routing updates, enter the **distribute-list** command:

```
Switch 8100fl(config-router)#distribute-list {aclname |
gateway gwprefix} {in [intf] | out [intf | ospf
ospf-process-num [vrf vrfid] | rip | static | connected]}
```

The following example shows how to filter incoming updates using access list 101:

```
Switch 8100fl(config)#router rip
Switch 8100fl(config-router)#
Switch 8100fl(config-router)#distribute-list 101 in
```

Redistributing Traffic from a Different Protocol

RIP has the ability to redistribute routes to a network using a different routing protocol such as OSPF. To redistribute RIP, or static routes, or connected routes, enter:

```
Switch 8100fl(config-router)#redistribute rip
<metric|route-map>
Switch 8100fl(config-router)#redistribute static
<metric|route-map>
Switch 8100fl(config-router)#redistribute connected
<metric|route-map>
```

Adjusting Timers

Tasks such as when to perform routing updates are controlled by timers. You can adjust these timers to fine tune RIP performance. Use the following command to update timers:

```
Switch 8100fl(config-router)#timers basic <interval>
```

Specifying the Version

The ProCurve 8100fl supports RIP version 1 and version 2. To specify which version of RIP is supported on this ProCurve Switch 8100fl, enter:

```
Switch 8100fl(config-router)#version {1|2}
```

Configuring an Interface for RIP

To configure RIP in the switch, you must first add interfaces in the Interface Configuration mode to inform RIP about attached interfaces.

Table 6-1. Configuring an Interface for RIP

Command	Action
ip rip authentication {key-chain | mode}	Enable RIP authentication on an interface
ip rip version {1|2}	Specify version for the interface
ip rip v2-broadcast	Enable version 2 broadcasts on an interface

RIP version 2 supports authentication (key-chain or mode). To enable RIP authentication, enter:

```
Switch 8100fl(config-if)#ip rip authentication key-chain <name of key chain>
```

To enable an interface to use Message Digest authentication or plain text authentication (the default) for a specified interface, enter:

```
Switch 8100fl(config-if)#ip rip authentication mode {md5 | <text>}
```

Specifying RIP Version

The ProCurve 8100fl supports RIP version 1 and version 2. To specify which version of RIP is supported on a specific interface, enter:

```
Switch 8100fl(config-if)#ip rip version {1|2}
```

Enabling IP Broadcasts

To send IP broadcast version 2 updates, enter:

```
Switch 8100fl(config-if)#ip rip v2-broadcast
```

Note RIP version 1 does not support multicast RIP packets.

Configuration Example

The following configuration example configures Gigabit Ethernet ports 3 and 4 in slot 1 to support Version 2 RIP and to apply MD5 authentication control to incoming RIP traffic.

The ProCurve Switch 8100fl is also configured to support RIP Version 2, to redistribute traffic from OSPF.

Routing is enabled for networks 192.168.12.0 and 192.168.22.0. Authentication is also provided by a key-chain with two passwords.

The active configuration for this RIP configuration would look like:

```
Switch 8100fl#show running-config
...
Current configuration:
! Last modified on 2003-04-30T19:41 by trial@Router
version 1.0
enable secret 5 $1$dcwu$rg0NelE9NV9pI.SoMeB.L0
hostname ProCurve 8100fl
!
interface GigabitEthernet1/3
  no shutdown
  ip address 192.168.99.1 255.255.255.0
  ip rip authentication key-chain starchain
  ip rip authentication mode md5
  ip rip version 2

!
interface GigabitEthernet1/4
  no shutdown
  ip address 192.168.101.1 255.255.255.0
  ip rip authentication key-chain starchain
  ip rip authentication mode md5
  ip rip version 2
...
interface Management0
  no shutdown
  ip address 10.203.11.27 255.255.0.0
!
router rip
  network 192.168.12.0
  network 192.168.22.0
  version 2
  redistribute ospf
```

Related Topics

For more about the protocol-independent features that apply to RIP, such as configuring authentication and routing policies, refer to Chapter 13, "Configuring Routing Policies".

For information on how to configure IP interfaces and general non-protocol-specific routing parameters, refer to Chapter 14, "IP Routing Configuration".

OSPF Configuration

Contents

Overview

Open Shortest Path First (OSPF) is a modern, scalable, and fast link-state routing protocol. It is an interior routing protocol (IGP), used to distribute routing information within the boundaries of an Autonomous System (AS). Each OSPF route chooses the shortest path to any known destination based on complete knowledge of the routing topology within the AS, and using Djikstra's SPF algorithm. Each OSPF router is responsible for informing the others about networks that are attached to it. It accomplishes that by sending Link State Advertisements, representing routers and links. Each OSPF router is also responsible for maintaining all LSAs received from other routers in its local Link State Database (LSDB).

Supported Features

The ProCurve Switch 8100fl implementation is compliant with the OSPFv2 specification, documented in RFC 2328. The NSSA Option, defined in RFC 1587, is also supported.

The ProCurve Switch 8100fl supports the following OSPF features:

- Definition of areas, including stub areas and Not So Stubby Areas (NSSAs) (RFC 1587).
- Link-state Advertisements or LSAs
- Authentication: Simple password and MD5 authentication methods are supported within an area and on an interface.
- Up to 55 OSPF adjacencies
- Configuration of virtual links
- Inter-area route summarization
- Summary filter
- External route summarization
- Static multi-path
- Configuration of parameters at the area, interface or global level. Parameters include retransmission interval, interface transmit delay, router priority, router dead and hello intervals, and authentication key.
- Route Redistribution: routes learned by OSPF from RIP can be redistributed into OSPF. OSPF routes can be redistributed into RIP . For more information on route redistribution, refer to Chapter 13, “Configuring Routing Policies”.

Multipath Support

The switch supports OSPF and static Multi-path. If multiple equal-cost OSPF or static routes have been defined for any destination, then the switch "discovers" and uses all of them. The switch will automatically learn up to sixteen equal-cost OSPF or static routes and retain them in its forwarding information base (FIB). The forwarding module then installs flows for these destinations in a round-robin fashion.

OSPF Areas

OSPF areas are a collection of subnets that are grouped in a logical fashion. Each area maintains its own link state database. The area topology is known only within the area. A router maintains a separate link state database for each area to which it is connected.

The goal of forming areas is to limit the number of routers that need to directly exchange routing information with each other, and to permit summarization of routing information on area boundaries. It is the link state nature of the protocol, combined with its ability to support hierarchy via areas, that results in significantly higher scalability than Distance Vector routing protocols, such as RIP.

There are several types of OSPF Areas supported by the ProCurve 8100fl, which differ in the way they handle External routes.

Backbone (area 0): The backbone is responsible for distributing routing information between non-backbone areas. OSPF areas communicate with other areas via the backbone area. The OSPF area backbone contains all area border routers (ABRs).

Normal: Normal areas can have Stubs in them, which redistribute externals into the area. Such Externals will be passed through the ABR which connects this area to the Backbone. Externals originated in other areas will also be injected into the Normal area via the ABR.

Stub: An ASBR cannot be placed in a Stub area, and therefore no Externals can be injected in it. An ABR that connects a Stub area to the Backbone will propagate into the Stub area only a default route, accompanied by an Inter-area route.

NSSA (Not So Stubby Area): Defined in RFC 1587, NSSAs can have ASBRs in them, therefore Externals can be injected into NSSAs. Those externals are propagated by the ASB towards the backbone. Externals injected into other areas are NOT propagated into the NSSA. In summary, the NSSA handles Externals as a Normal area in the direction from the NSSA to the Backbone, and as a Stub Area in the direction of the Backbone to the NSSA.

Area Routers. In connection to areas, the following terms are used in OSPF:

ABR (Area Border Router) — A router that connects the Backbone (area 0) with some other area(s).

ASBR (Autonomous System Border Router) —A router that redistributes routes in OSPF (from connected, statics, or another routing protocol). An ASBR can exist in the Backbone area, or in any other non-stub area.

OSPF Routes

Once areas have been configured, four types of OSPF routes can be encountered in the network. They are in order of preference:

Intra-area routes—represent destinations within the same area

Inter-area routes—reflect destinations from other OSPF areas

External Type 1 routes

External Type 2 routes

External routes. Both types of External routes represent destinations that are outside of the OSPF routing domain. These routes were injected into OSPF via redistribution, either from static or direct (connected) or from another routing protocol.

The difference between the two types of Externals is in the way their cost is calculated as routes are propagated within the OSPF domain.

In both types, Externals are redistributed by an ASBR within the OSPF domain. The ASBR always redistributes routers with a given cost.

For Type 1 Externals, the given redistribution cost is added to the OSPF link costs as the route is propagated downstream. As a result, the cost of a Type 1 route will increase as the route travels downstream in the OSPF domain.

The redistribution cost of a Type 2 External does not change as the route is propagated down the OSPF domain.

Note By default, the ProCurve 8100fl redistributes Type 2 Externals.

Configuring OSPF Router Parameters

To configure OSPF on the switch in the Router Configuration mode, perform the following tasks:

1. Enable OSPF
2. Set the router ID
3. Create and configure OSPF area
4. Add interfaces to the area
5. If necessary, configure virtual links
6. Optionally, configure redistribution
7. Optionally, configure parameters at the global, area, and/or interface level

Enabling OSPF

OSPF is disabled by default on the switch.

To enable OSPF, enter the following command:

```
Switch 8100fl(config)#router ospf <process ID>
```

Note You can only configure one process ID for OSPF.

Setting the Router ID

The router ID uniquely identifies the switch. To set the router ID to be used by OSPF, enter the following command from Routing Configuration mode.

```
Switch 8100fl(config-router)#ip router-id <IPaddr>
```

When setting the router ID, note the following conditions:

If you do not explicitly specify the router ID (using the **ip router-id** command), then an ID is chosen implicitly by the switch. The address on the loopback interface is the most preferred candidate for selection as the router ID for the switch.

If there are no addresses on the loopback interface, the switch will set the default router ID to the address of the first interface that is in the up state that the switch encounters (except the interface management0, which is the Management Module's interface). The address of a non point-to-point interface is preferred over the local address of a point-to-point interface.

If the router ID is implicitly chosen to be the address of a non-loopback interface, and if that interface were to go down, then the router ID is changed.

If you change the router ID (by using the **clear ip ospf** <*process ID*> command, an OSPF router has to flush all its LSAs from the routing domain.

If you explicitly specify a router ID, and if all the interfaces were to go down, the router ID would not change.

Configuring OSPF Areas

On the switch, you can create multiple OSPF areas, but at least one of them should be an area backbone for the router to function as an Area Border Router (ABR).

To configure a backbone area, set the **area** parameter to **0**.

```
Switch 8100fl(config-router)#area 0
```

Note The **area** parameter for a backbone area can also be set to **0.0.0.0**.

To configure an OSPF area, including a stub area or an NSSA, you must first create an area by entering the following command:

```
Switch 8100fl(config-router)#area <area ID>
```

where *area ID* is the OSPF area ID in decimal or in IP address format. After you create an area, the prompt will change so that you can set various OSPF area parameters.

```
Switch 8100fl(config-ospf-area)#
```

Configuring Summary Ranges

To reduce the amount of routing information propagated between areas, you can configure summary-ranges on Area Border Routers (ABRs). On the switch, summary-ranges are created using the **range** command

```
Switch 8100fl(config-ospf-area)#range <ipaddr-mask>
```

The networks specified using this command describe the scope of an area. Intra-area Link State Advertisements (LSAs) that fall within the specified ranges are aggregated into a single summary LSA that is advertised into other areas as inter-area routes. To turn off advertising the aggregated range, specify **not-advertise**.

You can also specify **no-discard** to specify that the router not generate a discard route for this range.

Configuring Stub Areas

The switch provides two ways to reduce the number of summary link advertisements (LSA Type 3) sent into a stub area.

To prevent the router from sending any Type 3 LSAs into the stub area, specify the **no-summary** keyword with the **stub** command. :

```
Switch 8100fl(config-ospf-area)#stub no-summary
```

Note

Use this no summary option if you do not want inter-area routes to be propagated into the stub area. In this case, default routing is used to reach inter-areas as external destinations.

Alternatively, you can configure summary filters to filter out specific summary LSAs from the stub area. Use this command for Type 3 LSAs you want to block. Type 3 LSAs that are not specified in this command will be sent into the stub area.

Configuring Stub Area Networks

If you have hosts and networks that are attached to a router that you want to be redistributed into OSPF, and you do not want to run OSPF on the interface, you can use the following stub network and stub host commands.

```
Switch 8100fl(config-ospf-area)#stubnetwork <ipaddr-mask>
cost <costvalue>
```

```
Switch 8100fl(config-ospf-area)#stubhost <ipaddr> cost
<costvalue>
```

To specify the cost to inject into a stub area:

```
Switch 8100fl(config-ospf-area)#default-cost <num>
```

To use a prefix-list to filter specific summary LSAs from a stub area, enter the following command:

```
Switch 8100fl(config-ospf-area)#summary-filter <prefix>
```

Configuring Not-So-Stubby Areas (NSSA)

NSSAs are similar to stub areas, in that they are used to restrict the AS-external routing for routers in the area. But unlike stub areas, NSSAs can originate and advertise Type-7 LSAs. Type-7 LSAs carry external route information which allow external routing within an NSSA.

Note

Type 7 LSAs are advertised only within a single NSSA; they are not flooded into the backbone area or any other area by border routers. NSSA border routers translate Type-7 LSAs into Type-5 LSAs and flood them to all Type-5 capable areas. However, the switch supports the configuration of NSSAs and the ability to add networks to an NSSA.

To define an area as an NSSA, enter the following command:

```
Switch 8100fl(config-ospf-area)#nssa
```

Enabling Authentication

The ProCurve 8100fl supports message-digest authentication for OSPF areas. To enable message-digest authentication for an area, enter:

```
Switch 8100fl(config-ospf-area)#authentication
[message-digest]
```

Note that you specify the actual keys to be used for authentication at the interface level.

Creating Virtual Links

You can create a virtual link to:

Connect an area via a transit area to the backbone

Create a redundant backbone connection via another area

Each ABR must be configured with the same virtual link. Note that virtual links cannot be configured through a stub or NSSA area.

To create a virtual link, enter:

```
Switch 8100fl(config-ospf-area)#virtual-link <ipaddr>
Switch 8100fl(config-ospf-area-virtuallink)#
```

The command options and parameters for configuring a virtual link include:

authentication enables authentication, using either a *message-digest* key or an *authentication-key* that specifies the OSPF password.

dead interval specifies the dead router detection time in seconds

hello interval configures the hello packet interval (in seconds) on this virtual link

retransmit interval configures the LSA retransmit interval (in seconds) on this virtual link

transmit delay configures LSA transmission delay in seconds

Configuring General OSPF Parameters

The switch provides several parameters that can be set at the OSPF router level. These parameters define:

OSPF router ID

Network in OSPF area

Default routes and metrics

Auto-cost

RFC 1583 compatibility

Internal and external distances

Adjacency change logging

Redistribution of OSPF traffic to other protocols

Timers

Configuring the OSPF Router

To specify the OSPF router ID, enter:

```
Switch 8100fl(config-router)#router-id <ip addr>
```

For information on setting router IDs, see "Setting the Router ID" on page 7-5.

Associating a Network with the OSPF Area

To identify which network IP addresses belong to an OSPF area, enter the following command:

```
Switch 8100fl(config-router)#network <ip addr> <mask> area
<area ID>
```

Distributing Default Information

You can define the metric to use for default routes, and you can specify the use of OSPF external metric Type 1 or Type 2 for that route. By default, OSPF ASBRs will not propagate a default AS-external route into the OSPF domain.

To enable the redistribution of default route into OSPF, enter:

```
Switch 8100fl(config-router)#default-information originate
<ospf metric> metric-type <1 | 2>
```

Setting the Reference Bandwidth

The switch uses the reference bandwidth to calculate the cost of an OSPF interface. The default reference bandwidth is 1000 Mbps. You can change this value by entering the following command:

```
Switch 8100fl(config-router)#auto-cost reference-bandwidth
<number in Mbps>
```

Configuring RFC 1583 Compatibility

To turn on support for RFC 1583, enter the following command:

```
Switch 8100fl(config-router)#compatible rfc1583
```

Logging Adjacency Changes

Support for logging changes in the adjacency states of OSPF neighbors is enabled by default. To turn it off, enter the following command:

```
Switch 8100fl(config-router)#no log-adjacency-changes
```

Redistribution

You can redistribute routes from another protocol into the OSPF domain.

To redistribute connected routes, enter the following command:

```
Switch 8100fl(config-router)#redistribute connected
[metric <default metric value>| metric-type <1 | 2>
|route-map <name> | tag <value>]
```

To redistribute static routes, enter the following command:

```
Switch 8100fl(config-router)#redistribute static [metric
<default metric value>| metric-type <1 | 2> |route-map
<name> | tag <value>]
```

Setting Default Metric for Redistributed Routes

Whenever you redistribute OSPF into another protocol, you must abide by the rules of that protocol. Specifically, the metric you configure must match the metric used by that protocol.

You can define the default metric to use for redistributed routes by specifying it in the **redistribute** command, or you can specify it separately. To directly specify the default metric to use for redistributed routes, enter:

```
Switch 8100fl(config-router)#default-metric <number>
```

Configuring Shortest Path First Computation Timers

To configure timers to control the delay in Shortest Path First (SPF) calculations, enter the following command:

```
Switch 8100fl(config-router)#timers spf <delay>
```

The *delay* setting is the time between receiving a change and initializing the spf computation.

Configuring OSPF Interface Parameters

To set OSPF interface parameters, use the **ip ospf** command for each interface in an OSPF area. The following parameters can be set at the interface level.

Parameter	Description
authentication	Enable authentication
authentication-key	Authentication password (key)
cost	Interface cost
dead-interval	Interval after which a neighbor is declared dead
hello-interval	Time between HELLO packets
message-digest-key	Message digest authentication password (key)
mtu-ignore	Ignore MTU check in DD packet
priority	Router priority
retransmit-interval	Time between retransmitting lost link state advertisements
transmit-delay	Link state transmit delay

Using OSPF Authentication

You enable OSPF authentication by specifying the authentication method at the area interface level, then specify the keys to be used for authentication at the interlace level.

To enable simple password-based authentication on an interface, enter:

```
Switch 8100fl(config-if)#ip ospf authentication
Switch 8100fl(config-if)#ip ospf authentication-key
<password>
```

You can also use the **ip ospf authentication** command at the area level and then specify the authentication-key for the interfaces in the area.

To use a message digest authentication password key, enter:

```
Switch 8100fl(config-if)#ip ospf authentication
message-digest-key <keyID>
```

You can also use the **ip ospf authentication message-digest** command at the area level and then specify the keys for the interfaces in the area.

To override authentication specified at the area level by specifying the authentication method at the interface level, enter:

```
Switch 8100fl(config-if)#ip ospf authentication null
```

Specifying **null** turns off authentication for this interface even if area authentication is specified.

Specifying the Interface Cost

The switch calculates the default cost of an OSPF interface using the reference bandwidth and the interface bandwidth. The default reference bandwidth is 1000 Mbps. It can be changed by using the **auto-cost reference-bandwidth** command.

A VLAN that is attached to an interface could have several ports of differing speeds. The bandwidth of an interface is represented by the highest bandwidth port that is part of the associated VLAN. The cost of an OSPF interface is inversely proportional to this bandwidth. The cost is calculated using the following formula:

Cost = reference bandwidth * 1,000,000/ *interface bandwidth (in bps)*

The following is a table of the port types and the OSPF default cost associated with each type:

Port Media Type	Speed	OSPF Default Cost
Ethernet 1000	1000 Mbps	1
Ethernet 10/100	100 Mbps	10
Ethernet 10/100	10 Mbps	100

To specify the cost to this interface (and override any automatically configured cost), enter:

```
Switch 8100fl(config-if)#ip ospf cost <num>
```

Specifying Intervals

OSPF allows you to control transmitting advertisements and waiting for other routers to send updates.

To limit the time to wait for a neighbor before declaring it dead, enter:

```
Switch 8100fl(config-if)#ip ospf dead-interval <num>
```

To limit the time between HELLO packets, enter:

```
Switch 8100fl(config-if)#ip ospf hello-interval <num>
```

To limit the time to wait before retransmitting lost-link-state advertisements, enter:

```
Switch 8100fl(config-if)#ip ospf retransmit-interval
<num>
```

To limit the link-state transmit delay, enter:

```
Switch 8100fl(config-if)#ip ospf transmit-delay <num>
```

Ignoring Maximum Transmission Unit Checks

To turn off checking the MTU checking on OSPF Database Description packets, enter:

```
Switch 8100fl(config-if)#ip ospf mtu-ignore
```

Setting the Priority Level

To specify the router's priority level (from 0 to 255), enter:

```
Switch 8100fl(config-if)#ip ospf priority <num>
```

Suppressing Routing Updates

To suppress routing updates on a specified interface, enter:

```
Switch 8100fl(config-router)#passive-interface <intf>
```

For example, to suppress routing on the Ethernet 3/1 interface, enter:

```
Switch 8100fl(config)#router ospf 4
Switch 8100fl(config-router)#
Switch 8100fl(config-router)#passive-interface ethernet 3/1
```

Note To suppress routing updates on all interfaces, use the **passive-interface default** command.

Alternative Area Border Router (ABR)

The switch automatically supports the alternative ABR implementation, as defined in the IETF "Alternative OSPF ABR Implementations" Internet Working Draft. This feature improves the behavior of a router connected to multiple areas without an active backbone connection. Behavior modifications allow the alternative ABR to successfully forward routes to the backbone and other areas despite not being actively connected to the backbone.

Note

The switch implements the alternative ABR feature automatically. No configuration changes are necessary.

ProCurve's OSPF implementation considers a router to be an ABR if it satisfies the following three requirements:

One or more non-backbone areas actively attached. As defined in the IETF working draft, "An area is considered *actively attached* if the router has at least one interface in that area in the state other than Down."

Area 0 configured.

An interface in the Up state in Area 0. This requirement is satisfied even if the adjacent interface on the Area 0 peer is in the Down state. As long as the ABR's interface in Area 0 has not been administratively shut down, it will continue to function as an ABR. A loopback interface belonging to the backbone will also be considered as an active attachment to the backbone.

If an ABR that is actively attached to more than one non-backbone area ceases to satisfy the above Area 0 requirements (configured and an interface in the Up state), it no longer functions as an ABR, provided that its non-backbone areas are connected to the backbone themselves.

Note

For meaningful routing to occur, the areas to which the Alternative ABR connects, must themselves be connected to the backbone. As the IETF draft reiterates, "[This feature does] not obviate the need of virtual link configuration in case an area has no physical backbone connection at all. The methods described here improve the behavior of a router connecting two or more *backbone-attached* areas."

OSPF Configuration Example

Figure 7-1 shows a sample OSPF configuration of a ProCurve 8100fl and several neighboring routers. The interfaces are GigabitEthernet ports and have MD5 authentication enabled. Except where noted in the configuration, all other OSPF interface and router parameters use default values:

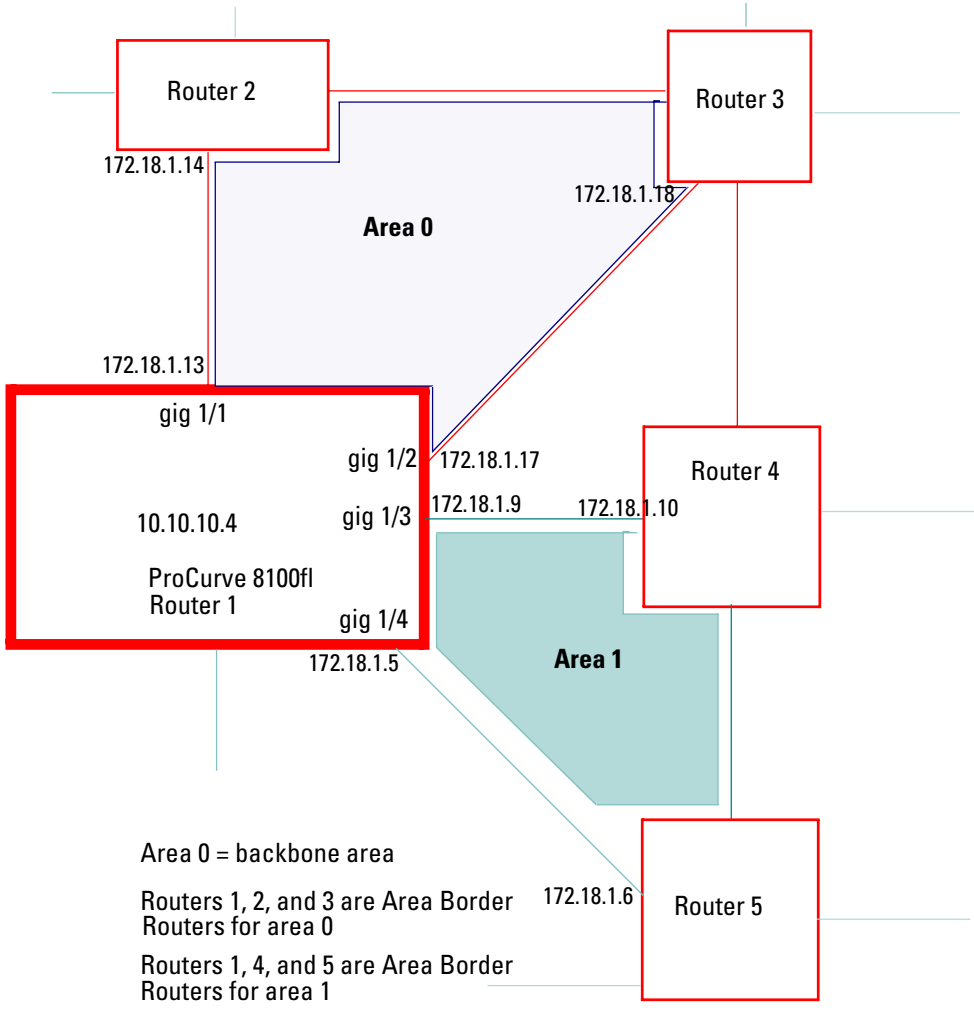


Figure 7-1. OSPF Configuration Example

The configuration for this sample OSPF configuration would look like:

```
 ...
 interface GigabitEthernet1/1
   no shutdown
   ip address 172.18.1.13
   ip OSPF message-digest-key 109 md5 2router1
   ip OSPF authentication message-digest
!
interface GigabitEthernet1/2
  no shutdown
  ip address 172.18.1.17
  ip OSPF message-digest-key 109 md5 2router2
  ip OSPF authentication message-digest
!
interface GigabitEthernet1/3
  no shutdown
  ip address 172.18.1.9
  ip OSPF message-digest-key 109 md5 2router3
  ip OSPF authentication message-digest
!
interface GigabitEthernet1/4
  no shutdown
  ip address 172.18.1.5
  ip OSPF message-digest-key 109 md5 2router4
  ip OSPF authentication message-digest
!
...
router OSPF 100
  area 0.0.0.0 stubhost 4.4.4.4 cost 100
  network 172.18.1.12 255.255.255.252 area 0.0.0.0
  network 172.18.1.16 255.255.255.252 area 0.0.0.0
  network 172.18.1.8 255.255.255.252 area 0.0.0.1
  network 172.18.1.4 255.255.255.252 area 0.0.0.1
ip route 0.0.0.0 0.0.0.0 172.17.4.1
ip router-id 10.10.10.4
...
```

Monitoring OSPF

The **show ip ospf** commands allow you to display detailed versions of the various OSPF tables. The **show ip ospf** commands can only display OSPF tables for the switch on which the commands are being entered (see the following examples and commands).

Example. Show ip ospf border routers:

```
Switch 8100fl#show ip ospf border-routers
OSPF Router with ID(0.0.0.0) (Process ID 0)
Destination Next Hop Cost Type Rte Type Area SPF No
32.32.32.32 3.3.3.2 1 ABR/ASBR INTRA 0.0.0.0 14
33.33.33.33 4.4.4.2 1 ABR/ASBR INTRA 0.0.0.0 14
```

Example. Show ip ospf counters:

```
Switch 8100fl#show ip ospf counters
Counters for OSPF Process:
Packets Received:
Monitor 0
Hello 492
Database Desciption 38
LS Request 1
LS Update 14
LS Acknowledgement 52
Packets Sent:
Monitor 0
Hello 604
Database Desciption 41
LS Request 3
LS Update 42
LS Acknowledgement 14
Errors:
Confusing Master/Initial flags 31
```

Example. Show ip ospf database:

```
Switch 8100fl#show ip ospf database
OSPF Router with ID(66.1.1.1) (Process ID 11)
Router Link States (Area 0.0.0.0)
Link ID ADV Router Age Seq# Checksum Link Count
32.32.32.32 32.32.32.32 1364 0x80000003 0x2414 1
33.33.33.33 33.33.33.33 1371 0x80000003 0x1416 1
66.1.1.1 66.1.1.1 1362 0x8000000D 0x42C4 4
67.1.1.1 67.1.1.1 161 0x80000011 0xA783 3
Net Link States (Area 0.0.0.0)
Link ID ADV Router Age Seq# Checksum Router Count
2.2.2.1 67.1.1.1 161 0x80000002 0x9AE8 2
3.3.3.1 66.1.1.1 1362 0x80000001 0xC087 2
4.4.4.1 66.1.1.1 1372 0x80000001 0xCE72 2
Type-5 AS External Link States
Link ID ADV Router Age Seq# Checksum Tag
5.5.5.0 66.1.1.1 185 0x80000003 0xB8C5 0
6.6.6.0 66.1.1.1 1636 0x80000001 0x98E4 0
```

Command. Show ip ospf flood list <*interface*>:

```
Switch 8100fl(config)#show ip ospf flood-list
gigabitethernet 4/1
```

Command. Show ip ospf interface <*interface*>:

```
Switch 8100fl(config)#show ip ospf interface
gigabitethernet 4/1
```

Command. Show ip ospf neighbors:

```
Switch 8100fl(config)#show ip ospf neighbors
```

Command. Show ip ospf request-list:

```
Switch 8100fl(config)#show ip ospf request-list
```

Command. Show ip ospf retransmission-list:

```
Switch 8100fl(config)#show ip ospf retransmission-list
```

Command. Show ip ospf virtual-links:

```
Switch 8100fl(config)#show ip ospf virtual-links
```

— This page is intentionally unused. —

VRRP Configuration

Contents

Overview

This chapter explains how to set up and monitor the Virtual Router Redundancy Protocol (VRRP) on the switch. VRRP is defined in RFC 2338.

In many networks, end hosts are often configured to send packets to a statically configured default router. If this default router becomes unavailable, all the hosts that use it as their first hop router become isolated on the network.

VRRP was developed as a way to ensure the availability of an end node's default router by assigning the IP address that end hosts use as their default route to a "virtual router." A Master router is assigned to forward traffic designated for the virtual router. If the Master router should become unavailable, a backup router takes over and begins forwarding traffic for the virtual router. As long as one of the routers in a VRRP configuration is up, the IP addresses assigned to the virtual router are always available, and the end hosts can send packets to these IP addresses without interruption.

Note

As of this release, the ProCurve Switch 8100fl is limited to two virtual router configurations per physical port. If VRRP is enabled on VLAN interfaces, care should be taken to ensure that trunk ports do not carry more than two VRRP enabled VLANs.

Configuration Parameters

This section covers settings you can modify in a VRRP configuration, including backup priority, advertisement interval, pre-empt mode, and authentication key. It also explains that a physical ports on a router are owned by that router. Using VRRP, you can configure other routers to take over as virtual routers for ports they do not own, but these virtual routers can never be owners of these ports.

Setting the IP Address of the Virtual Router

To assign the virtual router's IP address to be 10.50.50.5, enter:

```
Switch 8100fl(config-if)#vrrp 1 ip 10.50.50.5
```

Labeling the Virtual Router

You can label each virtual router for easy identification in configurations and the show commands. Labels can be up to 64 characters long (without spaces). To identify this virtual router as Site_5_Virtual_Router, enter:

```
Switch 8100fl(config-if)#vrrp 1 description
Site_5_Virtual_Router
```

Setting the Backup Priority

You can specify which Backup router takes over when the Master router goes down by setting the priority for the Backup routers. To set the priority for a Backup router, enter the following command in Configuration mode:

To specify 200 as the priority used by virtual router 1 on interface int1:

```
Switch 8100fl(config-if)#vrrp 1 priority 200
```

Priority levels can be between 1 (lowest) and 255. The default is 100. The priority for the IP address owner is 255 and cannot be changed.

Setting the Advertisement Interval

The VRRP Master router sends periodic advertisement messages to let the other routers know that the Master is up and running. In other words, VRRP routers learn timer settings from each other, not from their configuration file settings. By default, advertisement messages are sent once per second. To change the VRRP advertisement interval, enter the following command in Configuration mode:

To set the advertisement interval to 3 seconds:

```
Switch 8100fl(config-if)#vrrp 1 timers advertise 3
```

Learning the Master Configuration

When the Master router goes down, the Backup router takes over. When an interface comes up, the Master router may become available and take over from the Backup router. Before the Master router takes over, it may have to update its routing tables.

To learn the VRRP configuration for a Master router before the Backup takes over:

```
Switch 8100fl(config-if)#vrrp 1 timers learn
<authentication|description|ip|preempt|priority>
```

Setting Pre-empt Mode

When a Master router goes down, the Backup with the highest priority takes over the IP addresses associated with the Master. By default, when the original Master comes back up again, it takes over from the Backup router that assumed its role as Master. When a VRRP router does this, it is said to be in *pre-empt mode*. Pre-empt mode is enabled by default on the switch. You can prevent a VRRP router from taking over from a lower-priority Master by disabling pre-empt. To do this, enter the following command in Configuration mode:

```
Switch 8100fl(config-if)#no vrrp 1 preempt
```

Note If the IP address owner is available, then it will always take over as the Master, regardless of whether pre-empt mode is on or off.

Setting an Authentication Key

By default, no authentication of VRRP packets is performed on the switch. You can specify a clear-text password to be used to authenticate VRRP exchanges. To enable authentication, enter the following command in Configuration mode

For example, to authenticate VRRP exchanges on virtual router 1 on interface int1 with a password of ‘yago’, enter:

```
Switch 8100fl(config)#vrrp 1 authentication yago
```

Note The ProCurve Switch 8100fl does not currently support the IP Authentication Header method of authentication.

VRRP Configuration Notes

The Master router sends keep-alive advertisements. The frequency of these keep-alive advertisements is determined by setting the Advertisement interval parameter. The default value is 1 second.

If a Backup router doesn't receive a keep-alive advertisement from the current Master within a certain period of time, it will transition to the Master state and start sending advertisements itself. The amount of time that a Backup router will wait before it becomes the new Master is based on the following equation:

Master-down-interval = (3 * advertisement-interval) + skew-time

The skew-time depends on the Backup router's configured priority:

Skew-time = ((256 - Priority) / 256)

Therefore, the higher the priority, the faster a Backup router will detect that the Master is down. For example:

- Default advertisement-interval = 1 second
- Default Backup router priority = 100
- Master-down-interval = time it takes a Backup to detect the Master is down

  = (3 * adv-interval) + skew-time

  = (3 * 1 second) + ((256 - 100) / 256)

  = 3.6 seconds

If a Master router is manually rebooted, or if its interface is manually brought down, it will send a special keep-alive advertisement that lets the Backup routers know that a new Master is needed immediately.

A virtual router will respond to ARP requests with a virtual MAC address. This virtual MAC depends on the virtual router ID:

virtual MAC address = 00005E:0001*xy*

where *xy* is the virtual router ID (in hexadecimal format)

This virtual MAC address is also used as the source MAC address of the keep-alive Advertisements transmitted by the Master router.

If multiple virtual routers are created on a single interface, the virtual routers must have unique identifiers. If virtual routers are created on different interfaces, you can reuse virtual router IDs.

As specified in RFC 2338, a Backup router that has transitioned to Master will not respond to pings, accept Telnet sessions, or field SNMP requests directed at the virtual router's IP address.

By not responding the Backup router allows network management to notice that the original Master router (that is, the IP address owner) is down.

Configuring VRRP

This section presents two sample VRRP configurations:

- A basic VRRP configuration with one virtual router
- A symmetrical VRRP configuration with two virtual routers

Note

As of this release, the ProCurve Switch 8100fl is limited to two virtual router configurations per physical port.

Basic VRRP Configuration

Figure 8-1 shows a basic VRRP configuration with a single virtual router. Routers R1 and R2 are both configured with one virtual router (**VRID=1**). Router R1 serves as the Master and Router R2 serves as the Backup. The four end hosts (H1 - H4) are configured to use 10.0.0.1/16 as the default route. IP address 10.0.0.1/16 is associated with virtual router **VRID=1**.

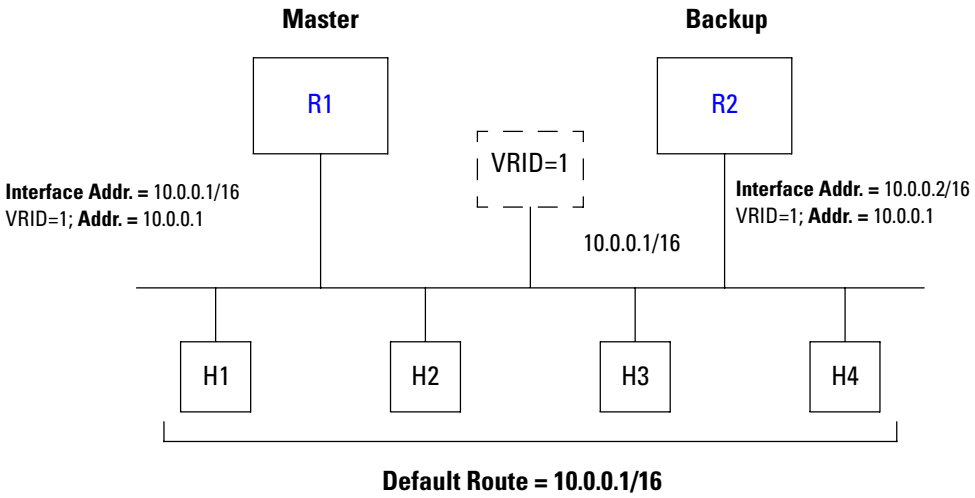


Figure 8-1. Basic VRRP configuration

If Router R1 should become unavailable, Router R2 would take over virtual router **VRID=1** and its associated IP addresses. Packets sent to 10.0.0.1/16 would go to Router R2. When Router R1 comes up again, it would take over as Master, and Router R2 would revert to Backup.

Configuration of Router R1

The following is the configuration file for Router R1 in Figure 8-1.

```
1: interface gigabitethernet 1/1
2: vrrp 1 ip 10.0.0.1
```

Line 1 declares the physical interface. Line 2 creates virtual router **VRID=1** on this interface and associates IP address 10.0.0.1 with virtual router **VRID=1**.

In VRRP, the router that owns the IP address associated with the virtual router is the Master. Any other routers that participate with this virtual router are the Backups. In this configuration, Router R1 is the Master for virtual router **VRID=1** because it owns 10.0.0.1, the IP address associated with virtual router **VRID=1**.

Configuration for Router R2

The following is the configuration file for Router R2 in Figure 8-1.

```
1: interface gigabitethernet 1/1
2: vrrp 1 ip 10.0.0.1
```

The configuration for Router R2 is nearly identical to Router R1. The difference is that Router R2 does not own IP address 10.0.0.1/16. Since Router R2 does not own this IP address, it is the Backup. It will take over from the Master if it should become unavailable.

VRRP Configuration with Two Routers

Figure 8-2 shows a symmetrical VRRP configuration with two routers and two virtual routers. Routers R1 and R2 are both configured with two virtual routers (**VRID=1** and **VRID=2**).

Router R1 serves as:

Master for **VRID=1**

Backup for **VRID=2**

Router R2 serves as:

Master for **VRID=2**

Backup for **VRID=1**

This configuration allows you to load-balance traffic coming from the hosts on the 10.0.0.0/16 subnet and provides a redundant path to either virtual router.

Note

This symmetrical configuration is the recommended configuration on a network using VRRP.

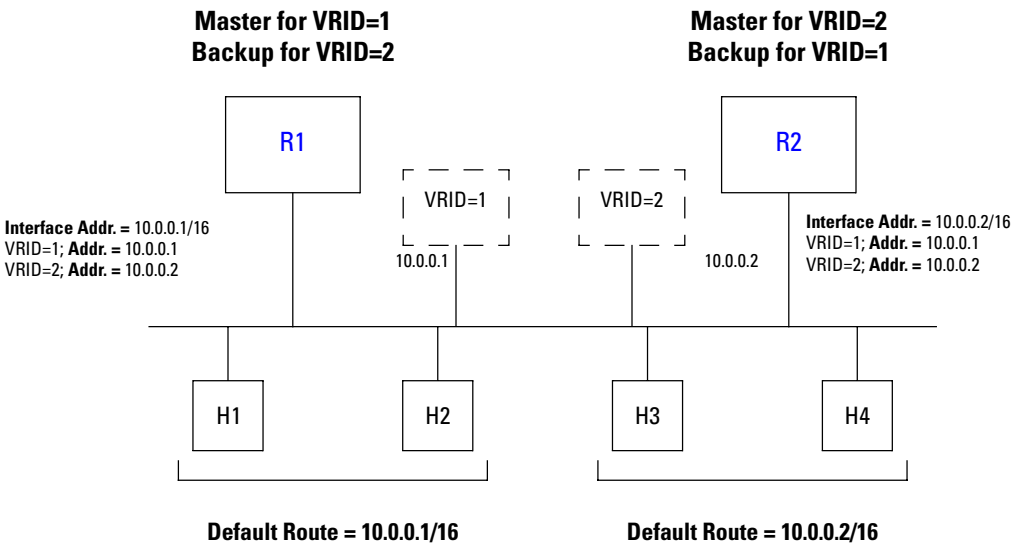


Figure 8-2. Symmetrical VRRP configuration

In this configuration, half the hosts use 10.0.0.1/16 as their default route, and half use 10.0.0.2/16. IP address 10.0.0.1/16 is associated with virtual router **VRID=1**, and IP address 10.0.0.2/16 is associated with virtual router **VRID=2**.

If Router R1, the Master for virtual router **VRID=1**, goes down, Router R2 would take over the IP address 10.0.0.1/16. Similarly, if Router R2, the Master for virtual router **VRID=2**, goes down, Router R1 would take over the IP address 10.0.0.2/16.

Configuration of Router R1

The following is the configuration file for Router R1 in Figure 8-2.

```
1: interface gigabitethernet 1/1
2: vrrp 1 ip 10.0.0.1
3: vrrp 2 ip 10.0.0.2
```

Router R1 is the owner of the IP address 10.0.0.1. Line 2 associates this IP address with virtual router VRID=1, so Router R1 is the Master for virtual router VRID=1.

On line 3, Router R1 associates IP address 10.0.0.2 with virtual router VRID=2. However, since Router R1 does not own IP address 10.0.0.2, it is not the default Master for virtual router VRID=2.

Configuration of Router R2

The following is the configuration file for Router R2 in Figure 8-2.

```
1: interface gigabitethernet 1/1
2: vrrp 2 ip 10.0.0.2
3: vrrp 1 ip 10.0.0.1
```

Router R2 is made owner of IP address 10.0.0.2. Line 2 associates this IP address with virtual router VRID=2, so Router R2 is the Master for virtual router VRID=2. Line 3 associates IP address 10.0.0.1 with virtual router VRID=1, making Router R2 the Backup for virtual router VRID=1.

Monitoring VRRP

The **show vrrp** command reports information about a VRRP configuration. You can specify individual VRIDs, or interfaces. You can tailor the display to show summary information or details, and you can focus displayed information using output modifiers to customize the information returned.

To show information about all virtual routers on GigabitEthernet interface 4/5

```
Switch 8100fl#show vrrp interface gig 4/5

Interface Gig4/5 - Group 5 ("This_is_a_test_description")
-----------------------------------------
Uptime                  Interface is currently down
State                   Init
Priority                1 (configured by the user)
Virtual MAC address     00005E:000105
Advertise Interval      25000 msec(s) (configured by the
user)
Preempt Mode            enabled delay = 0 msec(s)
Master Down Interval    75000
Authentication          Simple Text (configured by the user)
Primary Address         10.10.20.1
Associated Addresses    10.50.50.5
```

To display VRRP statistics for virtual router 5 on interface GigabitEthernet 4/5:

```
Switch 8100fl#show vrrp 5

Interface Gig4/5 - Group 5 ("This_is_VRID_5")
-----------------------------------------
Uptime                  Interface is currently down
State                   Init
Priority                1 (configured by the user)
Virtual MAC address     00005E:000105
Advertise Interval      25000 msec(s) (configured by the
user)
Preempt Mode            enabled delay = 0 msec(s)
Master Down Interval    75000
Authentication          Simple Text (configured by the user)
Primary Address         10.10.20.1
Associated Addresses    10.50.50.5
```

To display VRRP information on all interfaces and VRIDs, enter the **show vrrp** command in Privileged Exec mode.

9

Link Aggregation Configuration

Contents

Overview

This chapter explains how to configure:

- A manual (or static) SmartTRUNK group (LAG) on the switch
- A dynamic link using Link Aggregation Control Protocol (LACP).

Link aggregation on the ProCurve Switch 8100fl has the following features and characteristics:

- Link aggregation performs load balancing (based on the aggregation hash applied on the ingress ports), and load sharing across a number of ports.
- Link aggregation builds high-performance, high-bandwidth links between ProCurve's switching platforms.
- A link aggregation is a group of two or more physical ports that have been combined into a single logical port.
- Multiple physical connections between devices are aggregated into a single, logical, high-speed path that acts as a single link.
- As flows are set up on the SmartTRUNK, traffic is balanced across all ingress ports in the combined link, balancing overall available bandwidth.
- Link aggregations also provide improved data link resiliency—if one link fails, its flows are distributed among the remaining links.
- Link aggregations can interoperate with switches, routers, and servers from other vendors.
- Link aggregations allow administrators to increase bandwidth at congestion points in the network, eliminating potential traffic bottlenecks.
- Link aggregations are compatible with all switch features, including VLANs, STP, VRRP, and so on. In fact, switch link aggregation supports the bridging of any type of traffic, including protocols not currently supported. Non-IP traffic is passed using the source and destination MAC headers. IP traffic is passed on L2, L3, or L4 header information.

Note

The hash function distributes traffic making sure that each traffic stream uses the same link so packets do not have to be reordered. At Layer 2, a hash function is applied to the SMAC, DMAC addresses and vid. At Layer 3, hashing is based on the IPv4 source address and IPv4 destination address fields. The ProCurve Switch 8100fl also supports hashing based on Layer 4 SP and DP fields of the frame. You can configure hashing independently of the LAG mode; for example, you can use L4-based hashing on an L2 LAG.

Configuring Link Aggregation

The steps for creating and configuring a link aggregation are:

1. Create a SmartTRUNK, or LAG.
2. Add physical ports to the link aggregation.

Creating a LAG

When creating a SmartTRUNK, assign an ID to the SmartTRUNK. Here is an example of creating a SmartTRUNK with the ID of 11:

```
Switch 8100fl(config)#aggregator 11
```

Adding Physical Ports to the SmartTRUNK

You can add any number of 100/1000 Ethernet ports to a link aggregation, and ports can span across any number of interface modules. If one link should go down, traffic is redirected seamlessly to the remaining operational links.

Here is an example of adding port **gigabitethernet 2/5** to LAG 11:

```
Switch 8100fl(config)#interface gigabitethernet 2/5
Switch 8100fl(config-interface-gig2/5)#lag 11
```

Repeat these steps for each port you want to add to a specific LAG.

To remove a port from a LAG use the **no lag** command. For example:

```
Switch 8100fl(config)#interface gigabitethernet 2/5
Switch 8100fl(config-interface-gig2/5)#no lag 11
```

Link Aggregation Port Limitations

Ports added to a link aggregation must meet the following criteria:

Running in full duplex mode

Be a member of the default VLAN (VLAN 1)

Ethernet

Configured with the same bandwidth

Configuring Dynamic Aggregations

To configure and maintain a link aggregation group automatically, you must use 802.3ad LACP, which is supported on the switch. This protocol can detect the presence and capabilities of other aggregation capable devices automatically. In other words, using LACP, you can specify which links in a system can be aggregated.

The link aggregation is treated as the aggregator. The aggregator presents a standard IEEE 802.3 service interface and communicates with the MAC client. The aggregator binds to one or more ports, is responsible for distributing frames from a MAC client to its attached ports, and for collecting received frames from the ports and passing them to the MAC client transparently.

You can enable LACP on all Fast Ethernet, Gigabit Ethernet, and Tengigabit Ethernet ports on the switch. LACP ports exchange LACP Protocol Data Units (PDUs) with their peers and form one or more link aggregations. (This PDU traffic comes at the expense of data traffic. For this reason, manual LAG may be a better aggregation technology alternative to dynamic aggregation.)

After joining an aggregation, the port attaches to an appropriate aggregator. The benefit of being able to aggregate links using a combination of these same speed Ethernet ports on a single logical link is that it increases the options available when you have one remaining gigabit port and a number of 100M bit/sec ports available between switches. Network traffic is distributed across ports dynamically, so administration of what data actually flows across which port is managed within the aggregated link automatically. As with manual link aggregation, traffic is balanced by assigning flows to the least-used ingress ports in the aggregation.

Configuring Link Aggregations

To create and configure dynamic LAG, take the following steps:

1. Create the aggregation.
2. Specify the system.
3. Specify how flows are allocated on the LAG's ports.
4. Define the partner system LAG parameters.

Creating the Aggregation

The first thing you must do in creating a dynamic aggregation is to define the aggregation. Do this by entering from Configuration mode:

```
Switch 8100fl(config)#aggregator <aggregator ID number>
```

Specifying the System

1. To specify the system priority value for each host, from Configuration mode enter:

   ```
   Switch 8100fl(config)#lacp sys-priority 5
   ```

2. Repeat for the partner host.

Configuring the Port

1. To specify the 802.3ad parameters to define the port properties, from Interface Configuration mode enter:

   ```
   Switch 8100fl(config-if)#lacp activity
   Switch 8100fl(config-if)#lacp aggregation
   Switch 8100fl(config-if)#lacp port-key <number>
   Switch 8100fl(config-if)#lacp port-priority <number>
   Switch 8100fl(config-if)#lacp timeout
   Switch 8100fl(config-if)#lacp enable
   ```

2. To change the link-assignment algorithm, for instance, to L2 link aggregation and IP equal-cost multi-path (ECMP) routes, enter:

   ```
   Switch 8100fl(config-if)#aggr-mode {mac-based | l3-based
   | l4-based}
   ```

Note

The default link-assignment algorithm is Layer 3-based link assignment (l3-based).

3. Repeat for each port in the aggregation.

Configuring the Partner System

You have the option to configure the end-to-end specifications for the link aggregation. To do so, you must configure both ends of the links:

1. From the Configuration mode enter:

```
Switch 8100fl(config)#aggregator <lag ID number>
Switch 8100fl(config)#partner-sys-id <mac address>
Switch 8100fl(config)#partner-sys-priority <mac address>
```

2. Repeat for the other system.

Configuration Restrictions

Keep the following parameter configuration requirements in mind when setting up an aggregation.

Port's **port-key** value must be same as the aggregator's **actorkey** value

Aggregator's **partnerkey** value must be the same as the port's **partnerkey** value

Aggregator's **port-type** must be the same as the port's **port-type** (Fast Ethernet, Gigabit Ethernet, or Tengigabit Ethernet)

Aggregator's **aggregation** setting must be the same as the port's **aggregation** setting (**aggregatable** or **individual**)

If specified by the user, the aggregator's **partner-sys-priority** and **partner-sys-id** (MAC) must equal the port's **partner-sys-priority** and **partner-sys-id** (MAC).

Note All ports on which LACP is enabled are devoted solely to LACP. All ports controlled by any aggregator must have the same bandwidth.

Link Aggregation Configuration Example

Figure 9-1 shows manual LAG11 connecting five ports on System Blue to five ports on System Red. It also shows LACP 22 connecting three ports on System Blue to three ports on System Red.

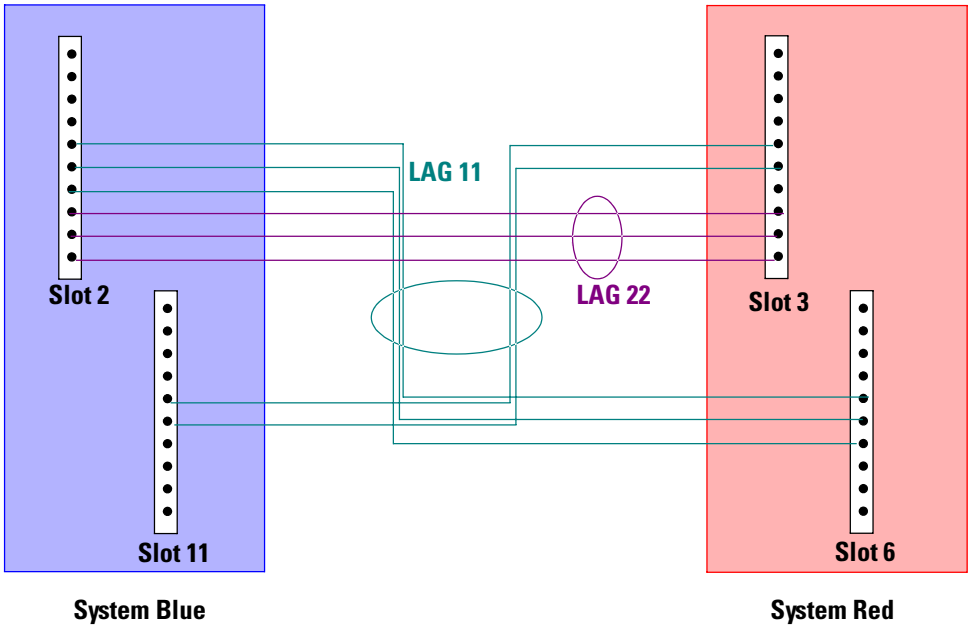


Figure 9-1. Link Aggregation Examples

Configuring A Manual Link Aggregation

In this example, the manual aggregator ID is set to 11, which becomes the LAG number assigned to System Blue's ports 5-7 of the Gigabit Ethernet ports on the interface module in slot 2, and to ports 5 and 6 on the interface module in slot 11. These ports connect to System Red's GigabitEthernet ports 5-7 on the interface module in slot 6, and to ports 5 and 6 on the interface module in slot 3.

The LACP aggregator ID is set to 22 and it describes the aggregation of ports 8-10 on the GigabitEthernet interface module in slot 2 on System Blue and connected to ports 8-10 on the GigabitEthernet interface module in slot 3 on System Red.

Figure 7-2 shows the configuration for these two aggregations on System Blue.

```
vlan 2-101
bridge stp
bridge-priority 1000
aggregator 11
aggregator 22
port-type gigethernet actorkey 12 partnerkey 50
interface GigabitEthernet2/5
no shutdown
lag 11
interface GigabitEthernet2/6
no shutdown
lag 11
interface GigabitEthernet2/7
no shutdown
lag 11
interface GigabitEthernet2/8
lacp enable
lacp port-key 12
no shutdown
interface GigabitEthernet2/9
lacp enable
lacp port-key 12
no shutdown
interface GigabitEthernet2/10
lacp enable
lacp port-key 12
no shutdown
interface GigabitEthernet11/5
no shutdown
lag 11
interface GigabitEthernet11/6
no shutdown
lag 11
interface LAG11
switchport mode trunk
switchport trunk-vlans 2-101
stp enable
interface LAG22
aggr-mode mac-based
switchport mode trunk
switchport trunk-vlans 2-101
stp enable
```

Figure 7-2. System Blue Configuration for LAG Aggregation 11 and LACP Aggregation 22

Figure 7-3 shows the corresponding configuration on System Red.

```
vlan 2-101
bridge stp
bridge-priority 2000
aggregator 11
aggregator 22
port-type gigethernet actorkey 50 partnerkey 12
interface GigabitEthernet3/5
no shutdown
lag 11
interface GigabitEthernet3/6
no shutdown
lag 11
interface GigabitEthernet3/8
lacp enable
lacp port-key 50
no shutdown
interface GigabitEthernet3/9
lacp enable
lacp port-key 50
no shutdown
interface GigabitEthernet3/10
lacp enable
lacp port-key 50
no shutdown
interface GigabitEthernet6/5
no shutdown
lag 11
interface GigabitEthernet6/6
no shutdown
lag 11
interface GigabitEthernet6/7
no shutdown
lag 11
interface LAG11
switchport mode trunk
switchport trunk-vlans 2-101
stp enable
stp cost 10
interface LAG22
switchport mode trunk
switchport trunk-vlans 2-101
stp enable
```

Figure 7-3. System Red Configuration for LAG Aggregation 11 and LACP Aggregation 22

Monitoring LAG and LACP

The following section shows commands and examples to use to view LAG and LACP configuration information and statistics.

Monitoring LAG Configurations

The **show port summary** command displays information on LAG configurations (see the examples for details).

The following example displays the categories of information available for port assignments.

```
Switch 8100fl#show port summary
N = Native VLAN, A = Access VLAN
Port   Admin Link  Mode  LAG VLAN-ID STP Auto Duplex Bridge Speed
Gig2/1 Down  Down Accs  None 1(N)   Dis  On   Full   Addr   1000
Gig2/2 Down  Down Accs  None 1(A)   Dis  On   Full   Addr   1000
Gig2/3 Down  Down Accs  None 1(A)   Dis  On   Full   Addr   1000
Gig2/4 Down  Down Accs  None 1(A)   Dis  On   Full   Addr   1000
Gig2/5 Up    Up   Trnk  11   1(N)   STP On   Full   Addr   1000
Gig2/6 Up    Up   Trnk  11   1(N)   STP On   Full   Addr   1000
Gig2/7 Up    Up   Trnk  11   1(N)   STP On   Full   Addr   1000
Gig2/8 Up    Up   Trnk  22   1(N)   STP On   Full   Addr   1000
Gig2/9 Up    Up   Trnk  22   1(N)   STP On   Full   Addr   1000
Gig2/10 Up   Up   Trnk  22   1(N)   STP On   Full   Addr   1000
Gig11/1 Down Down Accs  None 1(A)   Dis  On   Full   Addr   1000
Gig11/2 Down Down Accs  None 1(A)   Dis  On   Full   Addr   1000
Gig11/3 Down Down Accs  None 1(A)   Dis  On   Full   Addr   1000
Gig11/4 Down Down Accs  None 1(A)   Dis  On   Full   Addr   1000
Gig11/5 Up   Up   Trnk  11   1(N)   STP On   Full   Addr   1000
Gig11/6 Up   Up   Trnk  11   1(N)   STP On   Full   Addr   1000
Gig11/7 Down Down Accs  None 1(A)   Dis  On   Full   Addr   1000
Gig11/8 Up   Down Accs  None 1(A)   Dis  On   Full   Addr   1000
```

The following example displays the LAG attributes for LAGs.

```
Switch 8100fl#show lag all-lags lag-tuples
LAG Tuple Ports
--------------
[(1, 000a.af00.0dfe, 12, 0, 0), (255, --, 65535, 0, 0)]

Gig11/8
[(1, 000a.af00.0dfe, 12, 0, 0), (1, 000a.af00.50fe, 50, 0,
0)]

Gig2/8

Gig2/9

Gig2/10
[(1, 000a.af00.0dfe, 12, 0, 0), (1, 000a.af00.13fe, 13, 0,
0)]

Gig11/9

Gig11/10
```

The following example displays the categories of information available for LAG connections.

```
Switch 8100fl#show lag all-lags connections
Lag Name Sport Handle   Remote Switch Remote Port State Actor Key Partner Key
LAG.22  0x4072000000000000 000a.af00.50fe  0x148    Up    12          50
LAG.22  0x4082000000000000 000a.af00.50fe  0x149    Up    12          50
LAG.22  0x4092000000000000 000a.af00.50fe  0x14a    Up    12          50
```

The following example displays the categories of information available for LAG member ports.

```
Switch 8100fl#show lag all-lags member-ports
Lag Id Designated Port Member Ports Status
lag.11 Gig11/5         Gig2/5       enabled/up
                       Gig2/6       enabled/up
                       Gig2/7       enabled/up
                       Gig11/5      enabled/up
                       Gig11/6       enabled/up
Lag Id Designated Port Member Ports Status
lag.22 Gig2/10         Gig2/8       enabled/up
                       Gig2/9       enabled/up
                       Gig2/10      enabled/up
```

The following example displays the categories of information returned by the **show lag all-lag attributes** command for System Blue.

```
system_blue#show lag all-lags attributes
***********************************************
LAG 11 attributes
***********************************************
LAG Name : LAG11
Admin status : Up
Trunk status : Trunk
Native VLAN : 1
VLAN membership in : 101 VLANs (VLAN 1-101)
STP status : Enabled (VSTP 1)
Bridging Mode : Address Bridging
Aggr Mode : Layer 3
***********************************************
LAG 22 attributes
***********************************************
LAG Name : LAG22
Admin status : Up
Trunk status : Trunk
Native VLAN : 1
VLAN membership in : 101 VLANs (VLAN 1-101)
STP status : Enabled (VSTP 2)
Bridging Mode : Address Bridging
Aggr Mode : Layer 3
```

The following example displays the categories of information returned by the **show lag all-lag attributes** command for System Red.

```
system_red#show lag all-lag attributes
************************************************
LAG 11 attributes
************************************************
LAG Name : LAG11
Admin status : Up
Trunk status : Trunk
Native VLAN : 1
VLAN membership in : 101 VLANs (VLAN 1-101)
STP status : Enabled (VSTP 1)
Bridging Mode : Address Bridging
Aggr Mode : Layer 3
************************************************
LAG 22 attributes
************************************************
LAG Name : LAG22
Admin status : Up
Trunk status : Trunk
Native VLAN : 1
VLAN membership in : 101 VLANs (VLAN 1-101)
STP status : Enabled (VSTP 2)
Bridging Mode : Address Bridging
Aggr Mode : Layer 3
```

The following example displays the categories of information returned by the **show lag all-lag parameters** command for System Blue.

```
system_blue#show lag lag22 parameters

************************************************
LAG 22 parameters
************************************************
LAG Name : LAG22
Port Type : Gigabit Ethernet
Actor Key : 12
Partner Key : 50
Partner System Pri : 1
Partner System id : 000a.af00.50fe
Aggr Type : Aggregateable
```

The following example displays the categories of information returned by the **show lag all-lag parameters** command for System Red.

```
system_red#show lag lag22 parameters

*********************************************
LAG 22 parameters
*********************************************
LAG Name : LAG22
Port Type : Gigabit Ethernet
Actor Key : 50
Partner Key : 12
Partner System Pri : 1
Partner System id : 000a.af00.0dfe
Aggr Type : Aggregateable
```

Monitoring LACP

The **show lacp** command can be used to display information on LACP statistics and configurations (see the following examples for details).

The following example shows the information returned by the **show lacp <port> statistics** command for GigabitEthernet 5 on the interface module in slot 11.

```
Switch 8100fl#show lacp gi 11/5 statistics

LACP statistics (Gig11/9) :
LACP Pdus sent: 2071
Marker Response Pdus sent: 0
LACP pdus received: 2069
Marker pdus received: 0
```

Table 9-1 shows the information available from this command and explains the status being reported.

Table 9-1. show lacp <port> statistics

Fields	Description
LACP Pdus sent	The number of protocol data units sent on this interface since it was last activated.
Marker Response Pdus sent	The number of response protocol data units sent on this interface since it was last activated.

Table 9-1. show lacp <port> statistics

Fields	Description
LACP pdus received	The number of protocoldata units received on this interface since it was last activated
Marker pdus received	The number of response protocol data units received on this interface since it was last activated.

The following example displays the categories of information returned by the **show lacp <port> protocol** command.

```
Switch 8100fl#show lacp gi 2/9 protocol

port Gig2/9 LACP Protocol State:
   LACP State Machines:
      Receive FSM: Current State
      Mux FSM:     Collecting_Distributing State (LAG
0x8801080000000000 [0x21])
      Periodic Tx FSM: Fast Periodic State
   Control Variables
      BEGIN: FALSE
      Lacp Up: TRUE
      Ready_N: TRUE
      Selected: SELECTED
      Port_moved: FALSE
      NTT: FALSE
      port_enabled: TRUE
      PartnerSync: TRUE
      PartnerCollect: TRUE
  Timer counters
      periodic tx timer: 1
      current while timer: 3
      wait while timer: 0
```

The following example displays the categories of information returned by the **show lacp <port> parameters** command.

```
Switch 8100fl#show lacp gi 2/9 parameters
LACP parameters (Gig2/9) :
   Actor
        system priority:     1
        system mac addr:     000a.af00.0dfe
        port admin key:      12
        port oper key:       12
        port number:         1609
        port admin priority: 1
        port oper priority:  1
        LACP activity:       ACTIVE
        aggregation:         AGGREGATABLE
        timeout:             SHORT TIMEOUT
        synchronization:     TRUE
        collecting:          TRUE
        distributing:        TRUE
        defaulted:           FALSE
        expired:             FALSE
  Partner
        system priority:     1
        system mac addr:     000a.af00.13fe
        port oper key:       13
        port number:         643
        port priority:       1
        LACP activity:       ACTIVE
        aggregation:         AGGREGATABLE
        timeout:             SHORT TIMEOUT
        synchronization:     TRUE
        collecting:          TRUE
        distributing:        TRUE
        defaulted:           FALSE
        expired:             FALSE
```

10

Access Control Lists (ACLs)

Contents

Overview

This chapter explains how to configure and use Access Control Lists (ACLs) on the ProCurve Switch 8100fl. When used in conjunction with certain features, ACLs provide control over the forwarding of Layer 3 and layer-4 traffic as illustrated in Figure 10-1.

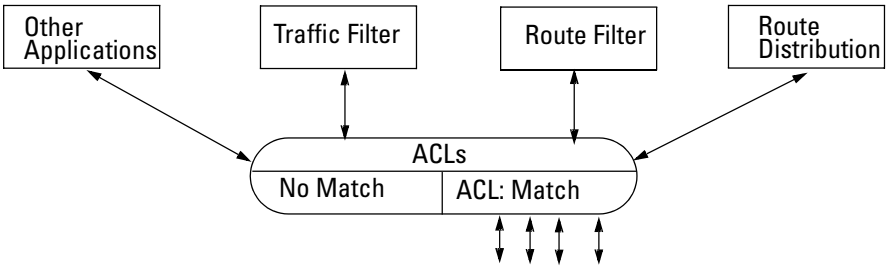


Figure 10-1. Using ACLs with Applications

Each ACL rule is a simple, logical statement of actions which are to be permitted or denied. Each rule is designed to act on a specific protocol, specific address, or specific destination. An ACL can be simple, consisting of only one rule, or complicated with many rules. Each rule tells the switch to either permit or deny the packet that matches the rule's packet description.

The real power in using ACLs is the ability to create multiple ACLs and apply them to the variety of traffic experienced in your network. Their main complexity consists in knowing how to apply them in an order that produces the desired results. This chapter explains the syntax of creating ACLs, and then shows how to order them to achieve the desired results.

The ProCurve Switch 8100fl supports two main categories of ACLs :

Layer 2 ACLs use the **l2acl** command to filter traffic based on source or destination MAC addresses (see "Layer 2 Access Control Lists (ACLs)" on page 10-13).

Layer 3/4 ACLs use the **access-list** command to filer traffic based on source or destination IP address, source or destination TCP/UDP port, ToS or protocol type for IP traffic. They also control access to services provided on the switch, for example, Telnet server and HTTP server.

Note

Currently, source filtering is available on switch interface modules; however, the application must take place on the entire interface module.

Configuring ACLs

An ACL consists of a protocol type and one or more rules which tell the switch to either *permit* or *deny* packets or routes that match the match criteria on which each rule is based. In this release, the Layer 3 ACL rules describe particular types of IP packets. ACLs can be simple, consisting of only one rule or they can be complicated, containing a number of rules for assessing packets.

ACLs can be created and configured using the **access-list** command from the Configuration mode of the CLI. The basic elements of a standard ACL are as follows:

```
Switch 8100fl(config)#access-list <n> <deny | permit>
<protocol | source> <destination>
```

where

<*n*> is the ACL ID or name
<*deny* | *permit*> represents the choices for action to be taken on a match
<*protocol* | *source ip*> is the protocol or source address
<*destination*> is the destination address

For example, the following ACL (*PermitTCP*) consists of a single **access-list** command that permits all IP packets from host *192.168.1.4* to go to host *10.203.101.1*.

```
Switch 8100fl(config)#access-list PermitTCP permit tcp 192.168.1.4
0.0.0.0 10.203.10.1 0.0.0.0
```

The following example is a more sophisticated ACL, consisting of three rules, that can be applied to inbound packets:

```
Switch 8100fl(config)#access-list 102 permit ip 10.121.96.0/24 any
Switch 8100fl(config)#access-list 102 deny ip 141.77.132.0/24 any
Switch 8100fl(config)#access-list 102 deny tcp any any
```

In the previous example, each rule is added to the ACL using separate entries of the **access-list** command.

Note

ACL rules are defined as either **permit** or **deny**. All ACL rules must either permit a packet or route or deny it. No other actions are permitted.

ACL Syntax

The syntax for creating an ACL is enforced by the CLI, which simplifies the process. Each ACL is identified by a name, consisting of alphanumeric characters. The ACL name can be a meaningful string such as *denyFTP* or it can be a simple number such as *100* or *101*.

Once you have specified a name (or number) for the ACL, and have specified a permit or deny action, then you must decide which of the following apply:

Table 10-1. Specifying the Action

Option	Description
<0..255>	The protocol number or name as shown in Table 10-6 on page 10-15
A.B.C.D	Source ip address
A.B.C.D and MASK	Source ip address with mask
AHP	Authentication Header Protocol
any	Any source host
host	A single source host
ICMP	Internet Control Message Protocol
IP	Any Internet Protocol
OSPF	OSPF routing protocol
UDP	User Datagram Protocol

From this list you can select a source address and a protocol. Once you specify a protocol for example, the other protocols are eliminated and the list narrows. Once you specify **any** source host, for example, you cannot then specify an individual source IP address.

At the next level or decision point, you must identify the destination:

Table 10-2. Identifying the Destination

Option	Description
A.B.C.D	Destination address
A.B.C.D and MASK	Destination address with Mask
any	Any destination host

Table 10-2. Identifying the Destination (Continued)

Option	Description
eq	Match only packets on a given port number (equal to)
gt	Match only packets with a port number greater than
host	A single destination host
lt	Match only packets with a port number less than
range	Match only packets in the port number range

Finally, you can continue to refine your ACL by specifying conditions for the traffic. This step is optional:

Table 10-3. ACL Options

Condition	Description
dscp	Match packets with given DSCP value
eq	Match only packets on a given port number
fragments	Check non-initial fragments
gt	Match only packets with a port number greater than
lt	Match only packets with a port number less than
range	Match only packets in the port number range

Note

There are three types of packets: Whole, unfragmented packets (W), Initial Fragments (IF), and Non-initial Fragments (NIF). If you do not specify **fragments** in the ACL rule, the ACL matches W and IF, but not NIF packets. If you do specify **fragments** in the ACL rule, the ACL matches only NIF packets.

The “Any” Parameter and Wild Cards

When defining an ACL it may be desirable to skip a match criteria field. For example, an ACL is defined where the source address is immaterial, but the destination address is required. Since each match criteria field is position-sensitive, you can use the keyword **any** to skip a field – in this case, the source address. In effect, **any** says “accept any value for this match criteria.”

For example, the following ACL denies IP traffic between any source and destination address and illustrates both the use of the **any** parameter and the use of wild carding:

```
Switch 8100fl(config)#access-list NoTelnet deny ip any any
```

Notice in the previous example that both the source address and the destination address are skipped over using the **any** parameter. The keyword **any** is needed only to skip a field in order to explicitly specify another field whose position is further along in the ACL.

How Multiple ACL Rules are Evaluated

The sequence of the rules within an ACL consisting of multiple rules is important. When an ACL application checks a packet or route against an ACL, it applies rules in the order in which they reside within the ACL – from first to last. The ProCurve Switch 8100fl also applies multiple ACLs in the order in which they are configured (the order in which they appear in the running-config). If a packet or route matches a rule, it is forwarded or dropped based on the **permit** or **deny** keyword in the rule. If there is no match, the packet or route is passed on to the next ACL.

Consequently, rules that are more specific (contain more match criteria) should be listed ahead of rules that are less specific. For example, the following ACL permits all TCP traffic except any TCP traffic from subnet **100.20.20.0/24**:

```
Switch 8100fl(config)#access-list 101 deny tcp 100.20.20.0/24 any
Switch 8100fl(config)#access-list 101 permit tcp any any
```

Notice in the previous example that ACL **101** includes two rules:

1. Deny TCP packets from subnet **100.20.20.0**
2. Permit TCP packets

A TCP packet coming from subnet **10.2.0.0/16** matches the first ACL rule, which results in the packet being dropped. However, a TCP packet coming from any other subnet does not match the first ACL rule. Instead, it matches the second ACL rule, which allows the TCP packet through.

Consider the case where the ACL rules in the previous example are reversed:

```
Switch 8100fl(config)#access-list 101 permit tcp any any
Switch 8100fl(config)#access-list 101 deny tcp 100.20.20.0/24 any
```

All TCP packets are allowed through, including packets from subnet **100.20.20.0**. Because TCP traffic coming from **100.20.20.0** matches the first rule, "all TCP packets are allowed through." The second rule is not applied because the first rule that matches determines the action taken on the packet.

Note

Remember that the first rule that applies to a packet is the only rule that affects the packet. The packet is permitted or denied according to the first rule it satisfies; none of the remaining ACL rules have any effect on the packet.

Implicit Deny Rule

At the end of each ACL, the switch automatically appends the *implicit deny rule*. For a packet or route that doesn't match any of the user-specified rules, the implicit deny rule acts as a catch-all rule that denies all packet or routes – all packets match this rule.

The implicit deny rule exists for security reasons. If an ACL is misconfigured, and a packet that should be allowed to go through is blocked by the implicit deny rule, the worst that happens is an inconvenience. However, a security breach results if a packet that should not be allowed through is sent through. As a result, the implicit deny rule serves as a fail-safe against the accidental misconfiguration of ACLs.

To illustrate how the implicit deny rule works, consider the following ACL:

```
Switch 8100fl(config)#access-list 101 permit ip 100.20.30.40/24 any
Switch 8100fl(config)#access-list 101 permit ip 124.123.220.10/24 any dscp
default
```

If a packet comes in and doesn't match either of the first two rules, the packet is dropped, because the third rule (the implicit deny rule) matches all packets. Although the implicit deny rule may seem obvious in the previous example, this is not always the case.

For example, consider the following ACL rule:

```
Switch 8100fl(config)#access-list 102 deny ip 172.124.200.0/24 any
```

If a packet comes in from a subnet other than **172.124.200.0/24**, one might expect the packet to go through, because it doesn't match the first rule, however, this is not the case. With the implicit deny rule attached, the rule looks like this:

```
Switch 8100fl(config)#access-list 102 deny ip 172.124.200.0/24 any
Switch 8100fl(config)#access-list 102 deny any
```

A packet coming from a subnet other than **172.124.200.0** would not match the first rule, but would match the implicit deny rule. As a result, no packets would be allowed through.

To allow packets from a subnet other than **172.124.200.0** to pass through, a rule must be explicitly defined to permit other packets to go through. To change the previous example so that it accepts packets from other subnets, a new rule must be added ahead of the implicit deny rule that permits packets to pass.

For example:

```
Switch 8100fl(config)#access-list 101 deny ip 10.1.20.0/24 any
Switch 8100fl(config)#access-list 101 permit ip any any
Switch 8100fl(config)#access-list 101 deny any
```

Notice that the second rule in this example forwards all IP packets that are not denied by the first rule, and this occurs before the implicit deny rule can be applied.

Because of the implicit deny rule, an ACL works similarly to a firewall that denies all traffic. ACL rules are then created that essentially open “doors” within the firewall that permit specific types of packets to pass.

Editing ACLs

To modify an ACL, edit it using a text editor on a remote workstation and upload it to the switch using TFTP or FTP. (You cannot edit existing ACLs from the CLI.) Edit, delete, replace, or reorder ACL rules and match criteria in a text file. The following example describes how to use this method to affect ACLs on the switch.

Suppose that ACL **104** is defined and applied to an interface on the switch, the following steps are performed to change the ACL using a text editor.

1. Use the **no** command to remove the definition and all references to ACL **104**:

   ```
   Switch 8100fl(config)#no access-list 104
   ```

2. On a workstation, enter the new ACL rules and references into the text file. In this example the text file is named **acl.changes**, which contains the changes to ACL **104** and its application to the GigabitEthernet interface:

   ```
   access-list 104 deny tcp 10.11.0.0/16 10.12.0.0/16
   access-list 104 permit tcp 10.11.0.0 any
   interface gigabitethernet 4/1
   access-list vlan 4098 in
   ```

3. Once you place the file **acl.changes** on a TFTP server (for example) that is reachable by the switch, and upload it to the switch, the changes are made active using the following commands:

```
Switch 8100fl#copy scp://10.1.1.12/config/acl.changes to scratchpad
Switch 8100fl#copy scratchpad to active
```

   The first **copy** command uploads the file **acl.changes** from the TFTP server to the configuration scratchpad. The next **copy** command makes the changes take effect by copying them into the active configuration.

   Copying the changes into the scratchpad allows the ACL changes to be checked before committing them to the active configuration. If you need to modify the ACL information in the scratchpad, make the changes in the file on the TFTP server, and then upload it to the scratchpad again.

Applying ACLs

Until it is applied, an ACL itself is simply a set of one or more rules made up of match criteria and an indicator that specifies whether to permit or deny packets that meet the rules. For an ACL to actually do something on the switch, it must be *applied* to an interface or to some application, which permits or denies traffic to or from the switch.

Applying ACLs to Interfaces

An ACL can be applied to an interface to make decisions about either inbound or outbound traffic. Inbound traffic is traffic coming into the switch. Outbound traffic is traffic going out of the switch. For each interface, only one ACL can be applied for the same protocol in the same direction. For example, you cannot apply two or more IP ACLs to the same interface in the inbound direction. You can apply two ACLs to the same interface if one is for inbound traffic and one is for outbound traffic. However, this restriction does not prevent you from specifying many rules in an ACL. Just put all of these rules into one ACL and apply it to the interface.

When a packet enters the switch through an interface where an inbound ACL is applied, the switch compares the packet to the rules specified by that ACL. If it is permitted, the packet is allowed into the switch. If not, the packet is dropped. The outbound packet is compared to the rules specified in this outbound ACL. Consequently, it is possible for a packet to go through two separate checks, once at the inbound interface and once more at the outbound interface.

To apply an ACL to an interface:

1. Within configuration mode, set your context to the interface where the criteria in the access list should be tested against inbound or outbound traffic.
2. Use the **ip access-group** command to apply an ACL to that interface.

   The following example shows how to apply an ACL called 101 to all inbound packets on the gigabit ethernet slot 4 port 1 interface:

```
Switch 8100fl(config)#interface gigabitethernet 4/1
Switch 8100fl (config-if)#ip access-group 101 in
```

ACL Viewing

The switch provides the following show commands that you can use to display the ACLs, their rules, and their association to interfaces, ports and services.

Table 10-4. ACL Show Commands

Show Command	Action
show access-list	Show all ACL definitions
show access-list debug lcpu-count debug lcpu-count <name>	Show a specific ACL definition
show access-list show implicit-acl <name> **| show implicit-acl | show implicit-deny**	Shows debug information
show access-list show implicit-acl	Show the syntax of the implicit ACL
show access-list show implicit-deny	Show the syntax of the implicit deny ACL

The following is an example of the display from the **show access-list** command:

```
Switch 8100fl#show access-list

Switch 8100fl#show access-lists
IP access list 401
        permit tcp 192.168.1.4 0.0.0.0  10.203.10.1 0.0.0.0

IP access list 403
        deny   tcp 10.20.20.0 0.0.0.255  any

        permit tcp any any

IP access list 404
        permit ip 123.1.3.10 0.0.0.255 any default

        permit ip any any

IP access list NoTelnet
        deny   ip any any

IP access list triple_rule
        permit ip 10.121.96.0 0.0.0.255 any

        deny   ip 141.77.132.0 255.255.255.255 any

        deny   tcp any  any
```

Notice that each ACL is listed along with its match criteria arranged on lines that represent the ACL's rules.

Layer 2 Access Control Lists (ACLs)

Layer 2 traffic filtering on the switch is provided by:

- Layer 2 filters - perform filtering on source or destination MAC addresses.
- Layer 2 access control lists - perform access control based on source or destination MAC address.

You can create Layer 2 filters at the port level using the **l2filter** command, or you can create a Layer 2 access control list using the **l2acl** command.

Note When MAC address filters and Layer 2 ACLs are enabled on the same port, MAC address filter processing precedes Layer 2 ACL processing; the device either forwards or drops the traffic based on the MAC filter policies, and the traffic is not subject to Layer 2 ACL processing.

Layer 2 Filters

To configure a Layer 2 filter, enter the following command and parameters:

```
Switch 8100fl(config)#l2filter <name of l2 filter list>
lock <port address> aaaa.bbbb.cccc <source MAC address>
vlan <VLAN ID> interface <port/slot>
```

Layer 2 ACLS

The following is an example of applying an ACL named *l2aclpermitany* to the source and destination MAC address:

```
Switch 8100fl(config)#l2acl l2aclpermitany permit any any
```

The following Layer 2 ACL denies traffic from MAC address 1111.2222.3333 to MAC address 4444.5555.6666:

```
Switch 8100fl(config)#l2acl l2denysome deny 1111.2222.3333
4444.5555.6666
```

To apply a Layer 2 ACL to a specified VLAN interface on input, enter the following command:

```
Switch 8100fl(config-if)#l2acl [police cir cbs ebs] aclname
vlan vlanid in
```

For example, to apply an ACL called 303 for traffic inbound to VLAN 220, you would enter;

```
Switch 8100fl(config)#interface gig 4/3
Switch 8100fl(config-if)#l2acl 303 vlan 220 in
```

Monitoring Layer 2 ACLs

Use the following commands to display information on Layer 2 ACLs.

Table 10-5. Monitoring Layer 2 ACLs

Command	Action
show l2acl	Show all L2 ACLs
show l2acl <name>	Show the specific L2 ACL
show l2acl resource-usage slot <number>	Show the impact on resource usage of L2 ACLs

The following is an example of the display from the **acl show all** command shows resource usage by interface module 1:

```
Switch 8100fl#show l2acl resource-usage interface-module 1
############################################################
L2 Rules=13, L3 Rules=4, Available=1007, Max=1024
############################################################
```

Protocols and Keywords

Table 10-6 shows the list of protocols you can use in an ACL. All of these protocols can be referenced by their Decimal number. Those protocols shown with a Keyword can alternately be referenced by this Keyword rather than by their decimal number.

Table 10-6. Protocol Decimal and Keyword Equivalents

Decimal	Keyword	Protocol/References
0		Reserved [JBP]
1	ICMP	Internet Control Message [RFC792,JBP]
2	IGMP	Internet Group Management [RFC1112,JBP]
3	GGP	Gateway-to-Gateway [RFC823,MB]
4	IP	IP in IP (encasulation) [JBP]
5	ST	Stream [RFC1190,IEN119,JWF]
6	TCP	Transmission Control [RFC793,JBP]
7	UCL	UCL [PK]
8	EGP	Exterior Gateway Protocol [RFC888,DLM1]
9	IGP	Any private interior gateway [JBP]
10	BBN-RCC-MON	BBN RCC Monitoring [SGC]
11	NVP-II	Network Voice Protocol [RFC741,SC3]
12	PUP	PUP [PUP,XEROX]
13	ARGUS	ARGUS [RWS4]
14	EMCON	EMCON [BN7]
15	XNET	Cross Net Debugger [IEN158,JFH2]
16	CHAOS	Chaos [NC3]
17	UDP	User Datagram [RFC768,JBP]
18	MUX	Multiplexing [IEN90,JBP]
19	DCN-MEAS	DCN Measurement Subsystems [DLM1]
20	HMP	Host Monitoring [RFC869,RH6]

Table 10-6. Protocol Decimal and Keyword Equivalents (Continued)

Decimal	Keyword	Protocol/References
21	PRM	Packet Radio Measurement [ZSU]
22	XNS-IDP	XEROX NS IDP [ETHERNET,XEROX]
23	TRUNK-1	Trunk-1 [BWB6]
24	TRUNK-2	Trunk-2 [BWB6]
25	LEAF-1	Leaf-1 [BWB6]
26	LEAF-2	Leaf-2 [BWB6]
27	RDP	Reliable Data Protocol [RFC908,RH6]
28	IRTP	Internet Reliable Transaction [RFC938,TXM]
29	ISO-TP4	ISO Transport Protocol Class 4 [RFC905,RC77]
30	NETBLT	Bulk Data Transfer Protocol [RFC969,DDC1]
31	MFE-NSP	MFE Network Services Protocol [MFENET,BCH2]
32	MERIT-INP	MERIT Internodal Protocol [HWB]
33	SEP	Sequential Exchange Protocol [JC120]
34	3PC	Third Party Connect Protocol [SAF3]
35	IDPR	Inter-Domain Policy Routing Protocol [MXS1]
36	XTP	XTP [GXC]
37	DDP	Datagram Delivery Protocol [WXC]
38	IDPR-CMTP	IDPR Control Message Transport Protocol [MXS1]
39	TP++	TP++ Transport Protocol [DXF]
40	IL	IL Transport Protocol [DXP2]
41	SIP	Simple Internet Protocol [SXD]
42	SDRP	Source Demand Routing Protocol [DXE1]
43	SIP-SR	SIP Source Route [SXD]
44	SIP-FRAG	SIP Fragment [SXD]
45	IDRP	Inter-Domain Routing Protocol [Sue Hares]
46	RSVP	Reservation Protocol [Bob Braden]
47	GRE	General Routing Encapsulation [Tony Li]

Table 10-6. Protocol Decimal and Keyword Equivalents (Continued)

Decimal	Keyword	Protocol/References
48	MHRP	Mobile Host Routing Protocol [David Johnson]
49	BNA	BNA [Gary Salamon]
50	SIPP-ESP	SIPP Encap Security Payload [Steve Deering]
51	SIPP-AH	SIPP Authentication Header [Steve Deering]
52	I-NLSP	Integrated Net Layer Security TUBA [GLENN]
53	SWIPE	IP with Encryption [JI6]
54	NHRP	NBMA Next Hop Resolution Protocol
55-60		Unassigned [JBP]
61		Any host internal protocol [JBP]
62	CFTP	CFTP [CFTP,HCF2]
63		Any local network [JBP]
64	SAT-EXPAK	SATNET and Backroom EXPAK [SHB]
65	KRYPTOLAN	Kryptolan [PXL1]
66	RVD	MIT Remote Virtual Disk Protocol [MBG]
67	IPPC	Internet Pluribus Packet Core [SHB]
68		Any distributed file system [JBP]
69	SAT-MON	SATNET Monitoring [SHB]
70	VISA	VISA Protocol [GXT1]
71	IPCV	Internet Packet Core Utility [SHB]
72	CPNX	Computer Protocol Network Executive [DXM2]
73	CPHB	Computer Protocol Heart Beat [DXM2]
74	WSN	Wang Span Network [VXD]
75	PVP	Packet Video Protocol [SC3]
76	BR-SAT-MON	Backroom SATNET Monitoring [SHB]
77	SUN-ND	SUN ND PROTOCOL-Temporary [WM3]
78	WB-MON	WIDEBAND Monitoring [SHB]
79	WB-EXPAK	WIDEBAND EXPAK [SHB]

Table 10-6. Protocol Decimal and Keyword Equivalents (Continued)

Decimal	Keyword	Protocol/References
80	ISO-IP	ISO Internet Protocol [MTR]
81	VMTP	VMTP [DRC3]
82	SECURE-VMTP	SECURE-VMTP [DRC3]
83	VINES	VINES [BXH]
84	TTP	TTP [JXS]
85	NSFNET-IGP	NSFNET-IGP [HWB]
86	DGP	Dissimilar Gateway Protocol [DGP,ML109]
87	TCF	TCF [GAL5]
88	IGRP	IGRP [CISCO,GXS]
89	OSPFIGP	OSPFIGP [RFC1583,JTM4]
90	Sprite-RPC	Sprite RPC Protocol [SPRITE,BXW]
91	LARP	Locus Address Resolution Protocol [BXH]
92	MTP	Multicast Transport Protocol [SXA]
93	AX.25	AX.25 Frames [BK29]
94	IPIP	IP-within-IP Encapsulation Protocol [JI6]
95	MICP	Mobile Internetworking Control Pro. [JI6]
96	SCC-SP	Semaphore Communications Sec. Pro [HXH]
97	ETHERIP	Ethernet-within-IP Encapsulation [RXH1]
98	ENCAP	Encapsulation Header [RFC1241,RXB3]
99		Any private encryption scheme [JBP]
100	GMTP	GMTP [RXB5]
101-254		Unassigned [JBP]
255		Reserved [JBP]

QoS Configuration

Contents

Overview

The ProCurve Switch 8100fl was designed with Quality of Service (QoS) in mind. QoS is performed globally and centrally by a scheduler that sees all the queues and all the priorities for every port. Therefore, the switch only has to queue traffic once on ingress to schedule traffic through the system, with the result that wire speed performance is not compromised.

The switch can guarantee bandwidth on an application by application basis, thus accommodating high-priority traffic even during peak periods of usage. QoS policies can be broad enough to encompass all the applications in the network, or relate specifically to a single host-to-host application flow.

Basic QoS Operation

The basic mechanism of QoS is to classify all traffic (in and out), then create policies to act on these classifications.The classification process uses what is known as a *class map;* and the policy process uses what is known as the *policy map*.

In addition to the classifier, there is a *bandwidth manager*, and a *WRED* (Weighted Random Early Detection) *engine*. Once you create a class map and a policy map, you attach the policy map to an incoming (ingress) port or outgoing (egress) port – or interface.

For the ProCurve Switch 8100fl, the QoS classifier processes incoming (ingress) traffic. The bandwidth manager and the WRED engine process outgoing (egress) traffic. The ProCurve Switch 8100fl also applies special policies on egress using **spolicy** commands.

The classifier engine enforces syntax for the policy map statement (what you see is what you get) making it very easy to use.

These processes are illustrated in Figure 11-1.

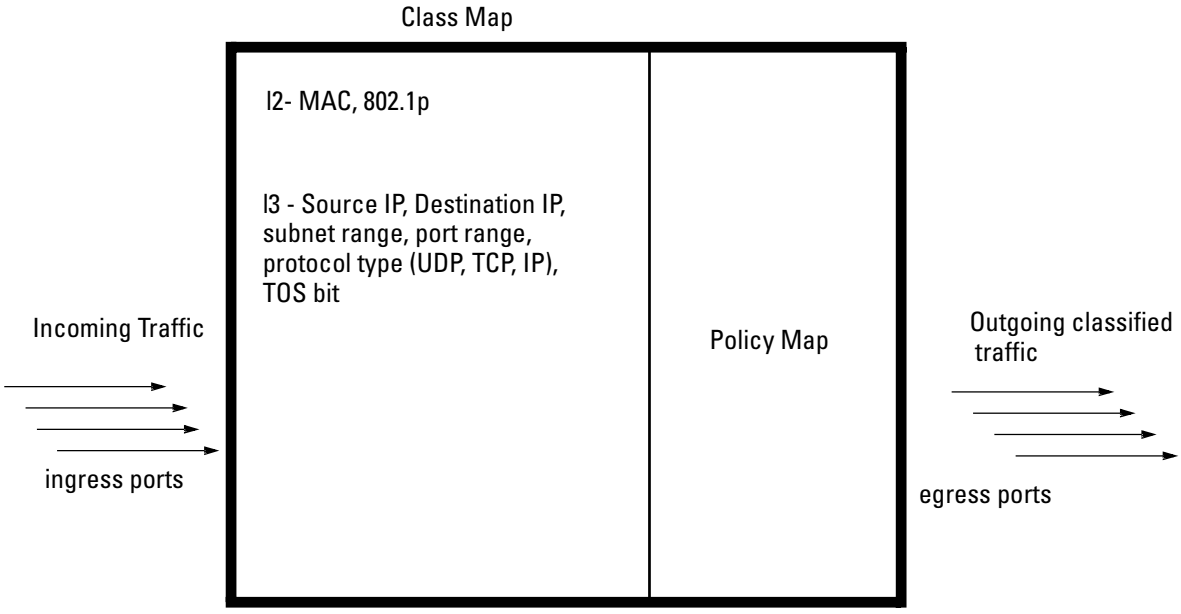


Figure 11-1. The QoS Classifier

Connecting Ingress and Egress Traffic

All incoming traffic is sorted into five queues or forwarding paths that can be controlled separately. Each queue allows for a different qualities of service and provides a specific type of treatment to traffic. All traffic on a single queue is treated the same.

These five queues or levels are:

EF (the priority queue)

DF (default forwarding)

AF1

AF2

AF3

You can customize any of these queues, although EF is the most restrictive. You can also configure the drop packet probabilities (1, 2, or 3) for the AF queues.

Using QoS Commands

This section explains the QoS commands available in this release.

Spolicy Input Commands

To access the special policy input mode, enter from Configuration mode:

```
Switch 8100fl(config)#spolicy-input-map <traffic policy name>
```

To access the spolicy input mode **map** command, enter from Policy Map Configuration mode:

```
 Switch 8100fl(config-pmap)#map <cos|ip-dscp|ip-precedence>
```

where *cos* matches the 802.1p Class of Service bits (0-7), *ip-dscp* matches the Differentiated Services Code Point bit (0 to 63), and *ip-precedence* matches the value of IP precedence (0-7).

Note When both cos and ip-dscp are configured, ip-dscp takes higher precedence and mapping will be done based on ip-dscp.

Spolicy Output Commands

To access the special policy output mode, enter from Configuration mode:

```
Switch 8100fl(config)#spolicy-output-map <traffic policy name>
```

The special policy output map mode controls access to four key QoS commands:

diff-class allows you to specify a diff-serv class and places you in the special output map differentiated class mode.

diff-group allows you to specify a diff-serv class group and places you in the special output map differentiated class group mode.

remap allows you to change the cos, ip-dscp, or ip-precedence setting on various diff-serv classes.

Differentiated Class

To configure a differentiated class, enter from Special Output Map mode

```
diff-class <diff-serv class>
```

Random Detection

Random detection allows you to control queues by specifying:

min-queue-fill (for the purpose of this discussion, call this A)

max-queue-fill (call this B)

max-queue-prob (call this C)

To configure random detection use the following command:

```
Switch 8100f1(config-spomap-dc)#random-detect <min-queue-
value> <max-queue-value> <max-queue-prob in %>
```

where **min-queue-fill** is expressed as an integer from 0 to 255 representing queue depth. This parameter is the threshold at which WRED (Weighted Random Early Detection, or random detection) is invoked. If you specify 0, WRED will always be on, If you specify 255, WRED will only be called when the queue is already full.

The **max-queue-fill** parameter is the other end of variable A and it works with the **max-queue-prob** parameter to determine at what queue saturation level packets are dropped all the time. Essentially, these B and C variables constitute coordinates on an *xy* plane as illustrated in Figure 11-2. The slope of your random detection algorithm is determined at one end by where you place A, and at the other end where C and B meet.

Note

Queue depths (variables A and B) are expressed in terms of a percentage of 256. Therefore 25% of 256 is 64 and 75% is 192. Queue probability (variable C) is simply a percentage.

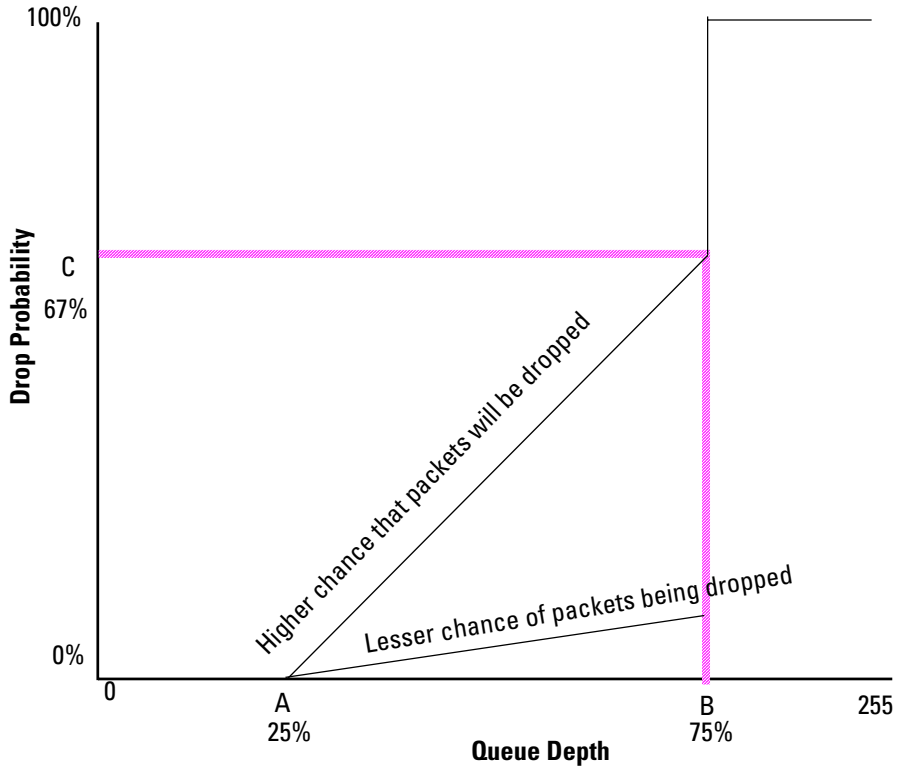


Figure 11-2. Calculating Random Detect Limits

In Figure 11-2, C represents the drop probability on a scale from 0 to 100%. A is the amount of queue that is full before WRED is invoked. And B is the amount of queue that is filled beyond which the drop probability is 100%.

In this mode, you can configure the **random-detect** command. This command configures how full the queue needs to be before the queue engine applies, at what point the drop probability is 1, and what the drop probability percentage is when the queue is full.

For example, if you want to invoke WRED when the queue is approximately 25% full, assign the drop probability to 1 when the queue is approximately 75% full, and drop all packets when the queue is completely full (drop probability is 100%), enter:

```
Switch 8100fl(config-spomap-dc)#random-detect 64 192 100
```

Differential Class Group

To configure differentiated class groups, enter from Special Output Map mode

```
Switch 8100fl(config-spomap)#diff-group <diff-class group>
```

There are five diff-class groups:

- af1x—Assured Forwarding Class 1
- af2x—Assured Forwarding Class 2
- af3x—Assured Forwarding Class 3
- df —Default Forwarding Class (aka best effort)
- ef—Expedited Forwarding Class (aka priority)

To configure Assured Forwarding Class 3 as the diff-class group, enter:

```
Switch 8100fl(config-spomap)#diff-group af3x
```

In the differentiated class group mode, you can use the **bandwidth** command to configure guaranteed bandwidth for traffic in this diff-class group. For example, to guarantee that 75% of bandwidth will be available for this diff-class group, enter:

```
Switch 8100fl(config-spomap-dcg)#bandwidth percent 75
```

You can guarantee a specific amount of the 10Gbps bandwidth (in Kbps). For example, if you want to guarantee at least 2,500,000 kbps, enter

```
Switch 8100fl(config-spomap-dcg)#bandwidth bandwidth 2500000
```

Interface Commands

The QoS traffic policy maps you create must be attached to an interface before they can process incoming traffic. For example, to define a service policy from an interface (Ethernet, GigabitEthernet, TenGigabitEthernet, etc.), enter:

```
Switch 8100fl(config-if)#service-policy <input|input-
spmap|output-spmap>
```

where *input* applies the named traffic policy, *input-spmap* applies the named input traffic policy, and *output-spmap* applies the named output traffic policy to this interface.

QoS Example

In the example shown in Figure 11-3, incoming packets arrive at a ProCurve Switch 8100fl on the edge of a diff-serv domain and are routed to a ProCurve Switch 8100fl in the core of the diff-serve domain. The ToS bit setting determines the priority handling through the domain and the switch processes the necessary queuing.

Consider how two packets (1 and 2) which arrive with ToS bits set to 7 and 8 respectively, are processed:

1. Packet 1 is assigned to AF1
2. Packet 2 is assigned to AF2.
3. The edge ProCurve Switch 8100fl routes the packets to the core ProCurve Switch 8100fl.
4. The core switch routes the packets to the diff-serv domain which knows how to route these packets to meet the requirements of the ToS bit.
5. The packets are assigned to different queues (2 and 4) reflecting their different service requirements and so are routed over different paths through the diff-serv domain.

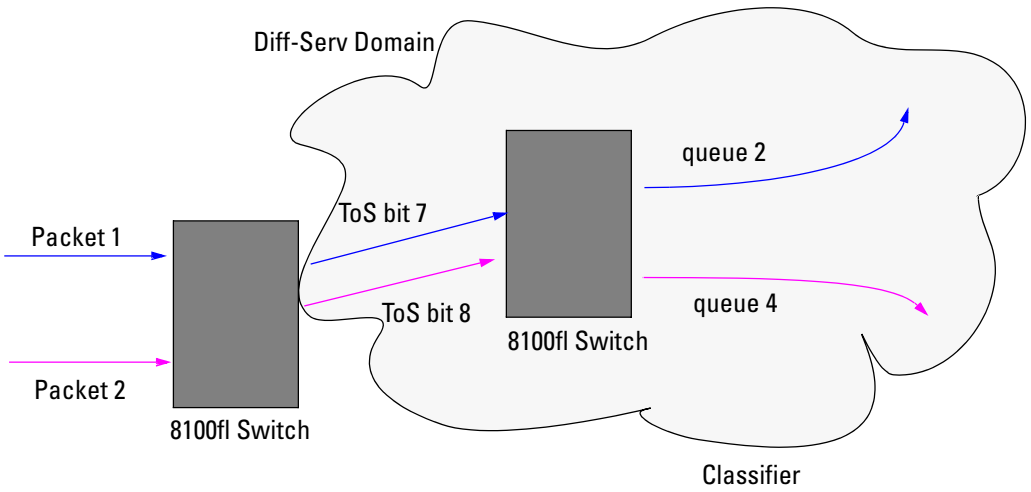


Figure 11-3. QoS Example

Bridging Configuration

Contents

Overview

The ProCurve Switch 8100fl provides the following bridging functions:

- Compliance with the IEEE 802.1D standard
- Wire-speed address-based bridging
- Ability to segment a transparently bridged network into logical virtual local-area networks (VLANs), based on physical ports
- Frame filtering based on MAC address for bridged traffic
- Integrated routing and bridging, which supports bridging of intra-VLAN traffic and routing of inter-VLAN traffic

Spanning Tree (IEEE 802.1D)

Spanning tree (IEEE 802.1D) allows bridges to discover dynamically a subset of the topology that is loop-free. In addition, the loop-free tree that is discovered contains paths to every LAN segment.

Bridging Modes

The ProCurve Switch 8100fl provides address-based wire-speed bridging: the switch performs this type of bridging by looking up the destination address in an L2 lookup table on the interface module that receives the bridge packet from the network. The L2 lookup table indicates the exit port(s) for the bridged packet. If the packet is addressed to the switch's own MAC address, the packet is routed rather than bridged.

VLAN Overview

Virtual LANs (VLANs) are a means of dividing a physical network into several logical (virtual) LANs. The division can be done on the basis of various criteria, giving rise to different types of VLANs. For example, the simplest type of VLAN is the port-based VLAN. Port-based VLANs divide a network into a number of VLANs by assigning a VLAN to each port of a switching device. Then, any traffic received on a given port of a switch *belongs* to the VLAN associated with that port.

VLANs are primarily used for broadcast containment. A Layer 2 (L2) broadcast frame is normally transmitted all over a bridged network. By dividing the network into VLANs, the *range* of a broadcast is limited. This means the broadcast frame is transmitted only to the VLAN to which it belongs. This reduces the broadcast traffic on a network by an appreciable factor.

Port-based VLANs

Ports of L2 devices (switches, bridges) are assigned to VLANs. Any traffic received by a port is classified as belonging to the VLAN to which the port belongs. For example, if ports 1, 2, and 3 belong to the VLAN named "Marketing", then a broadcast frame received by port 1 is transmitted on ports 2 and 3. It is not transmitted on any other port.

VLANs

VLANs are an integral part of the ProCurve Switch 8100fl, which can function both as Layer 2 (L2) switches and as fully-functional Layer 3 (L3) routers. Hence they can be viewed as a switch and a router in one box. To provide maximum performance and functionality, the L2 and L3 aspects of the ProCurve Switch 8100fl are tightly coupled.

The switch can be used purely as an L2 switch. Frames arriving at any port are bridged and not routed. In this case, setting up VLANs and associating ports with VLANs is all that is required.

The switch can also be used purely as a router, that is, each physical port of the switch is a separate routing interface. Packets received at any interface are routed and not bridged. In this case, no VLAN configuration is required.

Note that VLANs are still created implicitly as a result of creating L3 interfaces for IP. However, these implicit VLANs do not need to be created or configured manually. The implicit VLANs created by the switch are subnet-based VLANs.

Most commonly, the ProCurve Switch 8100fl is used as a combined switch and router. For example, the switch may be connected to two subnets S1 and S2. Ports 1-5 belong to S1 and ports 6-10 belong to S2. The required behavior of the switch is that intra-subnet frames be bridged and inter-subnet packets be routed. In other words, traffic between two workstations that belong to the same subnet should be bridged, and traffic between two workstations that belong to different subnets should be routed.

The ProCurve Switch 8100fl uses VLANs to achieve this behavior. This means that a Layer 3 subnet (that is, an IP subnet) is mapped to a VLAN. A given subnet maps to exactly one and only one VLAN. With this definition, the terms *VLAN* and *subnet* are almost interchangeable.

To configure a ProCurve 8100fl switch as a combined switch and router, the administrator must create VLANs whenever multiple ports of the switch are to belong to a particular VLAN/subnet. Then the VLAN must be *bound to* an L3 (IP) interface so that the switch knows which VLAN maps to which IP subnet.

Ports, VLANs, and L3 Interfaces

The term *port* refers to a physical connector on the switch, such as a GigabitEthernet port. Each port must belong to at least one VLAN. When the ProCurve Switch 8100fl is unconfigured, each port belongs to a VLAN called the "default VLAN." By creating VLANs and adding ports to the created VLANs, the ports are moved from the default VLAN to the newly created VLANs.

Unlike traditional routers, the ProCurve Switch 8100fl has the concept of logical interfaces rather than physical interfaces. An L3 interface is a logical entity created by the administrator. It can contain more than one physical port. When an L3 interface contains exactly one physical port, it is equivalent to an interface on a traditional router. When an L3 interface contains several ports, it is equivalent to an interface of a traditional router which is connected to a Layer 2 device such as a switch or bridge.

Explicit and Implicit VLANs

As mentioned earlier, VLANs can either be created explicitly by the administrator (explicit VLANs) or are created implicitly by the switch when L3 interfaces are created (implicit VLANs).

Access Ports and Trunk Ports (802.1P and 802.1Q support)

The ports of the ProCurve Switch 8100fl can be classified into two types, based on VLAN functionality: **access ports** and **trunk ports**. By default, a port is an access port. An access port can belong to at most one VLAN. Frames transmitted out of an access port contain no special information about the VLAN to which they belong. These frames are classified as belonging to a particular VLAN based on the VLAN configured on the receiving port.

Trunk ports (802.1Q) are usually used to connect one VLAN-aware switch to another. They carry traffic belonging to several VLANs.

For example, suppose that two separate ProCurve 8100fl switches (switch A and switch B) are both configured with VLANs V1 and V2. Then a frame arriving at a port on switch A must be sent to switch B, if the frame belongs to VLAN V1 or to VLAN V2. Thus the ports on switch A and B which connect the two switches together must belong to both VLAN V1 and VLAN V2. Also, when these ports receive a frame, they must be able to determine whether the frame belongs to VLAN V1 or to VLAN V2. This is accomplished by “tagging” the frames, that is, by prepending information to the frame in order to identify the VLAN to which the frame belongs.

In the ProCurve Switch 8100fl, trunk ports normally transmit and receive tagged frames only. (The format of the tag is specified by the IEEE 802.1Q standard.) If you configure Spanning Tree Protocol, frames are transmitted as untagged frames. You can also configure native VLANs to enable 802.1Q trunk ports to receive and transmit untagged frames by entering.

```
Switch 8100fl(config-if)#switchport trunk-native-vlan <VLAN
ID>
```

Configuring Spanning Tree

You may need to adjust certain spanning-tree parameters if the default values are not suitable for your bridge configuration. Parameters affecting the entire spanning tree are configured with variations of the bridge global configuration command.

Note

Only network administrators with a good understanding of how bridges and the Spanning-Tree Protocol work should make adjustments to spanning-tree parameters. Poorly chosen adjustments to these parameters can have a negative impact on performance. A good source on bridging is the IEEE 802.1D specification.

Setting the Bridge Priority

You can globally configure the priority of an individual bridge when two bridges tie for position as the root bridge, or you can configure the likelihood that a bridge will be selected as the root bridge. The lower the bridge's priority, the more likely the bridge will be selected as the root bridge. This priority is set by default; however, you can change it.

To set the bridge priority (from 0 to 65,535), enter the following command in Bridge Configuration mode:

```
Switch 8100fl(config-stp)#bridge-priority <value>
```

Adjusting Bridge Protocol Data Unit (BPDU) Intervals

You can adjust BPDU intervals as follows.

Adjusting the Interval between Hello BPDUs. You can specify the interval between hello BPDUs. To adjust this interval, enter the following command in Bridge Configuration mode.

```
Switch 8100fl(config-stp)#bridge hello-interval <value>
```

Defining the Forward Delay Interval. The forward delay interval is the amount of time spent listening for topology change information after an interface has been activated for bridging and before forwarding actually begins. To change the default forward delay interval setting, enter the following command in Bridge Configuration mode:

```
Switch 8100fl(config-stp)#bridge forward-delay <value>
```

Defining the Maximum Age. If a bridge does not hear BPDUs from the root bridge within a specified interval, it assumes that the network has changed and recomputes the spanning-tree topology. To change the default interval setting, enter the following command in Bridge Configuration mode:

```
Switch 8100fl(config-stp)#bridge max-age <value>
```

Configuring the Spanning Tree Interface

This section discusses the steps necessary to configure Spanning Tree ports. All commands listed are entered in Bridge Configuration mode.

Setting a Port Priority

To set an interface priority, enter the following command:

```
Switch 8100fl(config-if)#stp priority <value>
```

Assigning Port Costs

Each interface has a port cost associated with it. By convention, the port cost is 1000/data rate of the attached LAN, in Mbps. You can set different port costs. To assign port costs, enter the following command:

```
Switch 8100fl(config-if)#stp cost <value>
```

Enabling Spanning Tree

By default, spanning tree is disabled on the switch. To enable spanning tree on one or more ports, enter:

```
Switch 8100fl(config-if)#stp enable
```

Configuring a VLAN

This section shows you how to create a VLAN and assign ports.

Creating a VLAN

The ProCurve Switch 8100fl supports standards-based VLAN trunking between multiple ProCurve 8100fl switches as defined by IEEE 802.1Q. 802.1Q adds a header to a standard Ethernet frame. These VLAN IDs extend the VLAN broadcast domain to more than one switch.

To create a VLAN, enter the following command in Configuration mode:

```
Switch 8100fl(config)#vlan <ID>
```

To create a range of VLANs, enter the following command in Configuration mode:

```
Switch 8100fl(config)#vlan <number range>
```

To set the VLAN aging time, enter the following command in VLAN Configuration mode.

```
Switch 8100fl(config-vlan)#aging <aging-time>
```

For example, to set VLAN 229 aging time to 30 seconds, enter

```
Switch 8100fl(config)#vlan 229
Switch 8100fl(config-vlan)#aging 30
```

To assign a name and description to the specified VLAN, enter the following commands in VLAN Configuration mode.

```
Switch 8100fl(config-vlan)#name <string>
Switch 8100fl(config-vlan)#description <string>
```

To set the Maximum Transmission Unit (MTU) size on this VLAN, enter.

```
Switch 8100fl(config-vlan)#mtu <mtu-val>
```

Adding Ports to a VLAN

To configure a port that belongs only to one VLAN (that is, a port that sends out untagged packets), use the **switchport** command in Interface Configuration mode:

```
Switch 8100fl(config-if)#switchport mode access
```

To configure VLAN trunk ports, (that is, ports that send out tagged packets to multiple VLANs), enter:

```
Switch 8100fl(config-if)#switchport mode trunk
```

To designate trunk VLANs or add a port to a native VLAN, enter.

```
Switch 8100fl(config-if)#switchport trunk-vlans <VLAN ID>
```

Note

A native VLAN is the destination VLAN used for untagged packets.

The Default VLAN and Trunk and Access Port Behavior

When configuring VLANs, keep these configuration guides in mind:

All ports on an ProCurve 8100fl belong to a default, non-configurable VLAN: VLAN 1.

These ports will be access ports in shut mode.

VLAN Nonstandard Defaults

The ProCurve Switch 8100fl maintains some default port behaviors for VLAN configurations that are not industry standard. Please note the following nonstandard defaults:

The default status for a layer2 or layer3 port is shutdown. You must use the **no shut** command to enable these ports. You can verify a port's status by examining the running configuration to see if **no shut** appears for each port's configuration.

The default status for a Layer 2 VLAN is **no shut**. You must use the **shutdown** command to turn it off. Because these VLANs are enabled by default, you will not see **no shut** statements in the running configuration file.

The default status for a Layer 3 VLAN is **shutdown**. you must use the **no shut** command to enable these ports. You can verify a port's status by examining the running configuration to see if **no shut** appears for each port's configuration

LAGs (both L2 & L3) do not have a concept of shutdown. So you do not need to enter the **no shut** command.

Access Port Behavior

When an access port is made a member of any other VLAN (**switchmode access vlan <vlan-id>**), it is removed from VLAN 1.

If the port is deleted from the VLAN, it is made a member of VLAN 1 again.

If you change an access port to a trunk port, the native VLAN on the trunk port is set to VLAN 1.

You must configure a port with the **no shutdown** command before it becomes active.

Trunk Port Behavior

To change a port from access mode to trunk mode, use the **switchport mode trunk** command for the desired interface

Once the port is made the trunk, the native VLAN for the trunk port is set to be VLAN 1.

To change the native VLAN for the trunk port using **switchport trunk-native-vlan <vlan-id>**, where VLAN-ID can be from 2-4094).

If you do not need a native VLAN on the trunk port, use the command **switchport trunk-native-vlan disallow**. This will clear the trunk-native VLAN and the port will no longer accept any incoming untagged packets (untagged L2 control packets are fine). Use the command **no switchport trunk-native-vlan disallow** will set the native VLAN back to 1.

Trunk port membership is established by using the command **switchport trunk-vlans <vlan-id>**, where VLAN-ID can be from 2-4094).

Vlan 1 on trunk ports can be used only for native VLAN.

If you change a trunk port to an access port, it is put in the default VLAN (VLAN 1).

You must configure a port with the **no shutdown** command before it becomes active.

Monitoring Bridging

To display bridging statistics and configuration information, enter the following commands in Privileged Exec mode.

Table 12-1. Monitoring Bridging

Command	Action
show ip route	Show IP routing table
show stp	Show STP information
show bridge mac-table	Show master MAC table information.
show l2-vlan-translate	Show L2 VLAN translation information.
show vlan	Show all VLANs.

Changing the MAC age time

To set the MAC table aging timer, enter the following command:

```
Switch 8100fl(config)#bridge mac-table aging-time <age>
```

where *age* is the timer value in seconds.

— This page is intentionally unused. —

13

Configuring Routing Policies

Contents

Overview

The ProCurve Switch 8100fl supports flexible routing policies. These allow the network administrator to control import and export of routing information based on criteria including:

- Source and destination interface
- Previous hop router
- Tag associated with routes
- Specific destination address

The network administrator can specify a preference level for each combination of routing information being imported by using a flexible masking capability.

The switch also provides the ability to create advanced and simple routing policies. Simple routing policies provide a quick route redistribution between various routing protocols (RIP and OSPF). Advanced routing policies provide more control over route redistribution.

Route Preferences

Preference (or distance) is the value the switch routing process uses to order preference of routes from one protocol or peer over another. Preference can be set using several different configuration commands. You can set preference based on one network interface over another, or from one remote gateway over another. However, you cannot use preference to control the selection of routes within an Interior Gateway Protocol (IGP). This is accomplished automatically by the protocol based on metrics.

You can use preference to select routes from the same Exterior Gateway Protocol (EGP) learned from different peers or autonomous systems. Each route has only one preference value associated with it, even though the preference can be set at many places using configuration commands. The last or most specific preference value set for a route is the value used. A preference value is an arbitrarily assigned value used to determine the order of routes to the same destination in a single routing database. The active route is chosen by the lowest preference value.

A default preference is assigned to each source from which the switch routing process receives routes. Preference values range from 0 to 255 with the lowest number indicating the most preferred route.

Table 13-1 summarizes the default preference values for routes learned in various ways. The table lists the CLI commands that set preference, and shows the types of routes to which each CLI command applies.

Table 13-1. Default Preferences for Routes

Preference	Defined by CLI Command

OSPF routes	`Switch 8100fl(config-router)#default-information originate metric`
	`Switch 8100fl(config-router)#distance internal`
	`Switch 8100fl(config-router)#distance external`

Static routes from config	`Switch 8100fl(config-router)#ip route`

RIP routes	`Switch 8100fl(config-router)#default-metric`
	`Switch 8100fl(config-router)#distance`
	`Switch 8100fl(config-router)#default-information originate`

Import Policies

Import policies control the importation of routes from routing protocols and their installation in the routing databases (Routing Information Base and Forwarding Information Base). Import Policies determine which routes received from other systems are used by the switch routing process. Every import policy can have up to two components:

Import-Source

Route-Filter

Import-Source

This component specifies the source of the imported routes. It can also specify the preference to be associated with the routes imported from this source.

The routes to be imported can be identified by their associated attributes:

Type of the source protocol (RIP and OSPF).

Source interface or gateway from which the route was received.

In some cases, a combination of the associated attributes can be specified to identify the routes to be imported.

The importation of RIP routes may be controlled by source interface and source gateway. RIP does not support the use of preference to choose between RIP routes. That is left to the protocol metrics.

Due to the nature of OSPF, only the importation of ASE routes may be controlled. OSPF intra-and inter-area routes are always imported into the routing table with a preference of 10. If a tag is specified with the import policy, routes with the specified tag will only be imported. It is only possible to restrict the importation of OSPF ASE routes when functioning as an AS border router. Like the other interior protocols, preference cannot be used to choose between OSPF ASE routes. That is done by the OSPF costs.

Route-Filter

This component specifies the individual routes which are to be imported or restricted. The preference to be associated with these routes can also be explicitly specified using this component.

The preference associated with the imported routes are inherited unless explicitly specified. If there is no preference specified with a route-filter, then the preference is inherited from the one specified with the import-source.

Every protocol (RIP and OSPF) has a configurable parameter that specifies the default-preference associated with routes imported to that protocol. If a preference is not explicitly specified with the route-filter, as well as the import-source, then it is inherited from the default-preference associated with the protocol for which the routes are being imported.

Export Policies

Export policies control the redistribution of routes to other systems. They determine which routes are advertised by the Unicast Routing Process to other systems. Every export policy can have up to three components:

Export-Destination

Export-Source

Route-Filter

Export-Destination

This component specifies the destination where the routes are to be exported. It also specifies the attributes associated with the exported routes. The interface, gateway, or the autonomous system to which the routes are to be redistributed are a few examples of export-destinations. The metric, type, tag, and AS-Path are examples of attributes associated with the exported routes.

Export-Source

This component specifies the source of the exported routes. It can also specify the metric to be associated with the routes exported from this source.

The routes to be exported can be identified by their associated attributes:

Their protocol type (RIP, OSPF, Static, Connected).

Interface or the gateway from which the route was received.

Tag associated with a route. Both OSPF and RIP version 2 currently support tags. All other protocols have a tag of zero.

In some cases, a combination of the associated attributes can be specified to identify the routes to be exported.

Route-Filter

This component specifies the individual routes which are to exported or restricted. The metric to be associated with these routes can also be explicitly specified using this component.

The metric associated with the exported routes are inherited unless explicitly specified. If there is no metric specified with a route-filter, then the metric is inherited from the one specified with the export-source.

If a metric was not explicitly specified with both the route-filter and the export-source, then it is inherited from the one specified with the export-destination.

Every protocol (RIP and OSPF) has a configurable parameter that specifies the default-metric associated with routes exported to that protocol. If a metric is not explicitly specified with the route-filter, export-source as well as export-destination, then it is inherited from the default-metric associated with the protocol to which the routes are being exported.

Authentication

Authentication guarantees that routing information is only imported from trusted routers. Many protocols like RIP V2 and OSPF provide mechanisms for authenticating protocol exchanges. A variety of authentication schemes can be used. Authentication has two components – an Authentication Method and an Authentication Key. Many protocols allow different authentication methods and keys to be used in different parts of the network.

Authentication Methods

There are two main authentication methods: simple password and MD5.

Simple Password Authentication. In this method, an authentication key of up to 8 characters is included in the packet. If this does not match what is expected, the packet is discarded. This method provides little security, as it is possible to learn the authentication key by watching the protocol packets.

MD5 Authentication. This method uses the MD5 algorithm to create a crypto-checksum of the protocol packet and an authentication key of up to 16 characters. The transmitted packet does not contain the authentication key itself; instead, it contains a crypto-checksum, called the digest. The receiving router performs a calculation using the correct authentication key and discards the packet if the digest does not match. In addition, a sequence number is maintained to prevent replay of older packets. This method provides a much stronger assurance that routing data originated from a router with a valid authentication key.

Many protocols allow specification of two authentication keys per interface. Packets are always sent using the primary keys, but received packets are checked with both the primary and secondary keys before being discarded.

RFC 2178. The ProCurve Switch 8100fl supports MD5 specification of OSPF RFC 2178 which uses the MD5 algorithm and an authentication key of up to 16 characters. Thus there are three authentication schemes available per interface: none, simple, and RFC 2178 OSPF MD5 authentication. It is possible to configure different authentication schemes on different interfaces.

RFC 2178 allows multiple MD5 keys per interface. Each key has two times associated with the key: a time period that the key will be generated; and a time period that the key will be accepted.

Note

The Procurve Switch 8100fl allows only one MD5 key per interface. There are no options to specify the time period during which the key would be generated and accepted; the specified MD5 key is always generated and accepted.

Authentication Keys

An authentication key permits the generation and verification of the authentication field in protocol packets. In many situations, the same primary and secondary keys are used on several interfaces of a router.

Key Chains and Key Management

To make key management easier, the concept of a *key-chain* was introduced. Each key-chain has an identifier and can contain up to two keys. One key is the primary key and other is the secondary key. Outgoing packets use the primary authentication key, but incoming packets may match either the primary or secondary authentication key.

In Configuration mode, instead of specifying the key for each interface (which can be up to 16 characters long), you can specify a key-chain identifier.

Configuring Keys

To configure a key for your switch:

1. From configuration mode, enter the **key-chain** command:

   ```
   Switch 8100fl(config)#key-chain <chain-name>
   ```

   where <*chain-name*> is the name that you assign to key-chain.

2. From key-chain configuration mode, enter a key identifier. For example:

   ```
   Switch 8100fl(config-key-chain)#key-id 4
   ```

3. From keychain-key configuration mode, specify the authentication text for the authentication key using the key-string command.

   The following example specifies the authentication text **123456789** for key identifier **44** in the key chain called **mainkey**:

   ```
   Switch 8100fl(config)#key-chain mainkey
   Switch 8100fl(config-key-chain)#key-id 44
   Switch 8100fl(config-keychain-key)#key-string 123456789
   ```

Using Route Maps

A route map defines conditions and actions to be taken for:

- importing routes or exporting routes
- redistributing routes from or into any routing protocol

A route map consists of one or more *conditions* and the *action* to be taken when the condition is met. Each condition tells the switch to either permit or deny a route that matches the criteria specified in the route map. To be imported, exported, or redistributed, a route needs to meet the conditions of a configured route map. Note that a route can meet the conditions of a route map where the keyword **deny** is explicitly specified; in this case, the route will *not* be imported, exported, or redistributed.

To create a route map, enter the following commands in Configuration mode:

```
Switch 8100fl(config)#route-map <number-or-string> permit
<sequence-number> <match-criteria> <action>
Switch 8100fl(config)#route-map <number-or-string> deny
<sequence-number> <match-criteria>
```

In the following example, when the prefix of a route matches the network address 15.4.0.0, the route is redistributed to a next hop of 12.10.4.13.

```
Switch 8100fl(config)#route-map 1 permit 1 match-prefix
network 15.4.0.0/16 set next-hop 12.10.4.13
```

Configuring Next Hop Options

To set the values that control where to put packets that pass a match clause of a route map for policy routing, enter the **set** command at the Route Map Configuration level:

```
Switch 8100fl(config-route-map)#set {ip next-hop ipaddr2...
[verify-availability] | metric aval | metric-type {type-1 |
type-2} | tag tval}
```

The following example shows how to specify the next hop for the route-map called mainroutemap and verify that it is reachable before sending traffic.

```
Switch 8100fl(config)#route-map mainroutemap
Switch 8100fl(config-route-map)#set ip next-hop 10.203.1.26
verify-availability
```

Configuring Simple Routing Policies

Simple routing policies provide an efficient way for routing information to be exchanged between routing protocols. The **redistribute** command can be used to redistribute routes from one routing domain into another routing domain. Redistribution of routes between routing domains is based on route policies. A route policy is a set of conditions based on which routes are redistributed. While the **redistribute** command may fulfill the export policy requirement for most users, complex export policies may require the use of the commands listed under Export Policies.

Every protocol (RIP and OSPF) has a configurable parameter that specifies the default-metric associated with routes exported to that protocol. If a metric is not explicitly specified with the redistribute command, then it is inherited from the default-metric associated with the protocol to which the routes are being exported.

Redistributing Static Routes

Static routes can be redistributed to another routing protocol such as RIP or OSPF by the following command. To redistribute static routes, enter one of the following commands in Router Configuration mode:

```
Switch 8100fl(config-router)#redistribute static
[metric|route-map]
```

Redistributing Directly Attached Networks

Routes to directly attached networks are redistributed to another routing protocol such as RIP or OSPF by the following command. To redistribute direct routes, enter the following command in Router Configuration mode:

```
Switch 8100fl(config-router)#redistribute connected
[metric|route-map]
```

Redistributing RIP into RIP

The switch routing process requires RIP redistribution into RIP if a protocol is redistributed into RIP.

To redistribute RIP into RIP, enter the following command in Router Configuration mode:

```
Switch 8100fl(config-router)#redistribute rip [metric|route-
map]
```

Redistributing RIP into OSPF

RIP routes may be redistributed to OSPF.

To redistribute RIP into OSPF, enter the following command in Router Configuration mode:

```
Switch 8100fl(config)#redistribute ospf [match
<external|internal|nssa-external> |metric|route-map]
```

Redistributing OSPF to RIP

For the purposes of route redistribution and import-export policies, OSPF intra-area and inter-area routes are referred to as **ospf** routes, and external routes redistributed into OSPF are referred to as **ospf-ase** routes.

OSPF routes may be redistributed into RIP. To redistribute OSPF into RIP, enter the following command in Router Configuration mode:

```
Switch 8100fl(config)#redistribute rip [metric|route-map]
```

14

IP Routing Configuration

Contents

Overview

The ProCurve Switch 8100fl supports standards-based unicast routing for protocols such as TCP, UDP, and IP. Unicast routing protocol support covers both Interior Gateway Protocols and Exterior Gateway Protocols . This chapter describes how to configure IP interfaces and general non-protocol-specific routing parameters.

Interior Gateway Protocols are used for routing networks that are within an "autonomous system," a network of relatively limited size. All IP interior gateway protocols must be specified with a list of associated networks before routing activities can begin. A routing process listens to updates from other routers on these networks and broadcasts its own routing information on those same networks.

The ProCurve Switch 8100fl supports the following Interior Gateway Protocols:

- Routing Information Protocol (RIP) Version 1, 2 (RFC 1058, 1723). Configuring RIP is described in Chapter 6, "RIP Configuration" .
- Open Shortest Path First (OSPF) Version 2 (RFC 1583). Configuring OSPF is described in Chapter 7, "OSPF Configuration" .

Note

Multicast Routing Protocols, used to determine how multicast data is transferred in a routed environment, are not supported in this version.

Configuring IP Interfaces

You can configure an IP interface to a single port or to a VLAN. This section provides an overview of configuring IP interfaces.

Interfaces on the ProCurve Switch 8100fl are logical interfaces. Therefore, you can associate an interface with a single port or with multiple ports:

To associate an interface with a single port, specify the slot and port with the **interface** command.

To associate an interface with multiple ports, first create an IP VLAN and add ports to it, then use the VLAN option with the **interface vlan** command.

The **ip** command creates and configures an IP interface. Configuration of an IP interface can include information such as the interface's name, IP address, netmask, broadcast address, and so on.

Configuring IP Interfaces to Ports

You can configure an IP interface directly to a physical port. Each port can be assigned multiple IP addresses representing multiple subnets connected to the physical port. For example, to assign an IP interface address to physical port GigabitEthernet 3/4, enter the following:

```
Switch 8100fl(config)#interface gigabitethernet ip address
10.50.0.0/255.255.0.0
```

Configuring IP Interfaces for a VLAN

You can configure one IP interface per VLAN. In this case the port will send out untagged packets. To configure a port for a VLAN, enter:

```
Switch 8100fl(config-if)#switchport mode access vlan <VLAN
ID>
```

If you need the port to send out tagged packets (that is, the port belongs to more than one VLAN), then enter:

```
Switch 8100fl(config-if)#switchport mode trunk
Switch 8100fl(config-if)#switchport trunk-vlans <VLAN ID>
```

Extending the IP Configuration

You can configure an ProCurve 8100fl interface to support the following configurations:

ip access-group specifies the name of an access control list to control packets

ip address sets the IP address of an interface

ip broadcast-address sets the broadcast address of an interface

ip helper-address specifies a destination IP address for UDP broadcast

ip ospf configures Open Shortest Path First (OSPF) protocol commands

ip policy route-map specifies the policy (route map) to be applied on the interface

ip prefix list builds a prefix list that defines traffic to forward and/or reject from a specified subnet.

ip rip configures ip Routing Information Protocol (RIP) interface commands

You can also enable the following IP functions on interfaces:

ip mask-reply enables the sending Internet Control Message Protocol (ICMP) Mask Reply messages

ip spoofing enables IP spoofing

ip unreachables enables IP unreachables which sends ICMP messages back to senders of unknown protocols or undeliverable packets

ip proxy-arp enables proxy ARP

ip redirects enables IP re-directs

Configuring Jumbo Frames

Certain ProCurve 8100fl interface modules support jumbo frames (frames larger than the standard Ethernet frame size of 1518 bytes).

To transmit frames of up to 9216 bytes, you increase the maximum transmission unit (MTU) size from the default of 1522. You must set the MTU at the port level with the **interface mtu** command. You can also set the MTU at the IP interface level; if you set the MTU at the IP interface level, the MTU size must be less than the size configured for each port in the interface. Note that the interface MTU only determines the size of the packets that are forwarded in software.

For this release, there are some limitations on the number of 1G ports on a module that can simultaneously support jumbo frames. The following two configurations are supported:

All ten 1G ports configured for frame sizes up to 4500 bytes

Four 1G ports configured for frame size of 9216, the remaining six 1G ports configured for 1518 bytes (the standard Ethernet frame size).

Six 1G ports configured for a frame size of 9216, no other ports configured.

In the following example, the ports GigabitEthernet 3/1 through GigabitEthernet 3/4 are configured with an MTU size of 9216 bytes. Ports GigabitEthernet 3/5 through GigabitEthernet 3/10 are configured with an MTU size of 1518 bytes.

```
Switch 8100fl(config)#interface gigabitethernet 3/1 mtu 9216
...
Switch 8100fl(config)#interface gigabitethernet 3/2 mtu 9216
...
Switch 8100fl(config)#interface gigabitethernet 3/4 mtu 9216
...
Switch 8100fl(config)#interface gigabitethernet 3/5 mtu 1518
...
Switch 8100fl(config)#interface gigabitethernet 3/7 mtu 1518
...
Switch 8100fl(config)#interface gigabitethernet 3/10 mtu 1518
```

Layer 2 Filters

Layer 2 filters on the ProCurve Switch 8100fl allow you to configure ports to filter specific MAC addresses. When defining a Layer 2 filter, you specify to which ports you want the filter to apply. You can specify the following filters:

Address filters. These filters block traffic based on the frame's source MAC address, destination MAC address, or both source and destination MAC addresses in flow bridging mode. Address filters are always configured and applied to the input port.

Port-to-address lock filters. These filters prohibit a user connected to a locked port or set of ports from using another port.

Configuring Layer 2 Address and Port-to-Address Lock Filters

If you want to control access to a source or destination on a per-MAC address basis, you can configure an address filter. Address filters are always configured and applied to the input port. You can set address filters on a source MAC address, which filters out any frame coming from a specific source MAC address

To configure Layer 2 address filters, enter the following commands in Configuration mode:

```
Switch 8100fl(config)#l2filter <name> lock <MACaddr> vlan
<VLAN-num> in-port-list <port-list> interface <interface-
slot-port>
```

Layer 2 Filter Examples

Figure 14-1 shows an example of the router connections for which Layer 2 filters will be configured.

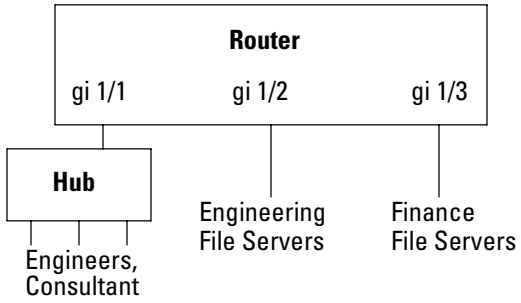


Figure 14-1. Filter example

Example: Address Filters

The following example configures a Layer 2 filter for GigabitEthernet port 2 in slot 1, for the specified MAC address which is in VLAN 2.

```
Switch 8100fl(config)#l2filter paull2test lock
0002.b34c.10cf 0000.0000.0000 vlan 2 in-port-list interface
gigabitethernet 1/2
Switch 8100fl(config)#
```

Configuring Address Resolution Protocol (ARP)

The ProCurve Switch 8100fl allows you to configure Address Resolution Protocol (ARP) table entries and parameters. ARP is used to associate IP addresses with media or MAC addresses. Taking an IP address as input, ARP determines the associated MAC address. Once a media or MAC address is determined, the IP address/media address association is stored in an ARP cache for rapid retrieval. Then the IP datagram is encapsulated in a link-layer frame and sent over the network.

Configuring ARP Cache Entries

To create an ARP entry for the IP address 10.8.1.2 at GigabitEthernet port 4/7 for 1200 seconds:

```
Switch 8100fl(config)#interface gigabitethernet 4/7
Switch 8100fl(config-if)#arp timeout 1200
```

To create a permanent ARP entry for the host *10.10.100.23* at GigabitEthernet port 4/6:

```
Switch 8100fl(config)#arp 10.10.100.23 aaaa.bbbb.cccc gig
4/6
```

To remove the ARP entry for the host 10.8.1.2 from the ARP table:.

```
Switch 8100fl#clear arp 10.8.1.2
```

To clear the entire ARP table.

```
Switch 8100fl#clear arp-cache
```

Note

The **clear arp** command is only used to clear an individual arp entry, while the **clear arp cache** command clears all arp entries from the entire arp table.

If the Startup configuration file contains **arp** add commands, the Management Module re-adds the ARP entries even if you have cleared them using the **clear arp** commands. To permanently remove an ARP entry, use the **negate** command or **no** command to remove the entry.

Configuring ARP Refresh Interval

The **arp refresh** command causes an ARP packet to be issued periodically for stations currently known in the router's ARP cache. This packet refreshes the entry to prevent timeout of the ARP entry. It also helps detect MAC station movement between different ports of a VLAN.

In the case of VLANs, implementing this command is highly recommended as it helps detect MAC movement changes due to either physical station moves or changes in network topology regardless of whether or not ARP packets are being re-issued by the moving station.

To configure the ARP refresh interval:

1. From Configuration mode, enter the VLAN interface.
2. Enter the ARP refresh interval using the **arp refresh** command.

For example, to configure Vlan 701 with an ip address of 171.1.1 255.255.255.0, an arp refresh interval of 120 seconds, and an arp timeout of 300 seconds, you would enter the following commands:

```
Switch 8100fl(config)#interface Vlan701
Switch 8100fl(config-if)#ip address 172.1.1.1 255.255.255.0
Switch 8100fl(config-if)#arp refresh 120
Switch 8100fl(config-if)#arp timeout 300
```

By default, ARP refresh is enabled and uses a refresh interval of 60 seconds, and can be overridden by specifying a different value. To prevent ARP refreshes from taking place on an interface, the value 0 can be specified.

Unresolved MAC Addresses for ARP Entries

When the switch receives a packet for a host whose MAC address it has not resolved, it tries to resolve the next-hop MAC address by sending ARP requests. Five requests are sent initially for each host, one every second.

Configuring Proxy ARP

The ProCurve Switch 8100fl can be configured for proxy ARP. The proxy ARP (as defined in RFC 1027) is used to help hosts with no knowledge of routing determine the MAC address of hosts on other networks or subnets. Through proxy ARP, the switch will respond to ARP requests from a host with a ARP reply packet containing the switch's MAC address. The following example enables proxy ARP for TenGigabitEthernet interface 8/1:

```
Switch 8100fl(config)#int ten 8/1
Switch 8100fl(config-if)#ip proxy-arp
```

To turn off proxy ARP for an interface, enter:

```
Switch 8100fl(config-if)#no ip proxy-arp
```

Monitoring ARP

Use the **show arp** command to verify and troubleshoot your ARP configurations.

Configuring Basic IP Parameters

This section explains how to configure the following basic IP parameters.

Configuring DNS Parameters

The ProCurve Switch 8100fl can be configured to specify DNS servers, which supply name services for DNS requests. You can specify up to three DNS servers.

For example, to configure the default DNS server with the domain name "ProCurve_8100.com", enter:

```
Switch 8100fl(config)#ip domain-name ProCurve_8100.com
Switch 8100fl(config)#ip domain-name-server 10.100.100.20
Switch 8100fl(config)#ip domain-lookup
```

To create a list of domain names to be used when resolving a host name, enter:

```
Switch 8100fl(config)#ip domain-list <DNS server name>
```

Configuring IP Services (ICMP)

The ProCurve Switch 8100fl provide ICMP message capabilities including ping and traceroute. The **ping** command allows you to determine the reachability of a certain IP host, while the **traceroute** command allows you to trace the IP gateways to an IP host.

Note

You can issue single or multiple ping tests with varying repetitions and timeout periods. Type **?** to list the full set of parameters and commands you can execute.

Configuring IP Helper

The **ip helper-address** Interface command allows you to forward specific UDP broadcast from one interface to another. Typically, broadcast packets from one interface are not forwarded (routed) to another interface. However, some applications use UDP broadcast to detect the availability of a service. Other services, for example BOOTP/DHCP require broadcast packets to be routed

so that they can provide services to clients on another subnet. An IP helper can be configured on each interface to have UDP broadcast packets forwarded to a specific host for a specific service or forwarded to all other interfaces.

You can configure the ProCurve Switch 8100fl to forward UDP broadcast packets received on a given interface to all other interfaces or to a specified IP address. You can specify a UDP port number for which UDP broadcast packets with that destination port number will be forwarded. By default, if no UDP port number is specified, the switch will forward UDP broadcast packets for the following six services:

BOOTP/DHCP (port 67 and 68)

DNS (port 37)

NetBIOS Name Server (port 137)

NetBIOS Datagram Server (port 138)

TACACS Server (port 49)

Time Service (port 37)

For example, to forward UDP broadcast packets received on interface GigabitEthernet 3/1 to the host 10.1.4.5 for the six default UDP services, enter:

```
Switch 8100fl(config-if)# ip helper-address 10.1.4.5
```

Enabling IP Forwarding

The **ip forward-protocol** (global) command allows you to control the forwarding of physical and directed IP broadcasts.

For example, to enable forwarding of IP broadcasts on the switch, enter:

```
Switch 8100fl(config)#ip forward-protocol udp bootps
```

Monitoring IP Parameters

The ProCurve Switch 8100fl provides display of IP statistics and configurations contained in the routing table. Information displayed provides routing and performance status.

The **show ip interface** commands display IP information, such as routing tables, status, and IP interface configuration, on the switch. The following example displays all established connections for the switch.

```
Switch 8100fl#show ip interface brief
Interface     IP-Address            Status Protocol  Vlan
Gig1/3        172.18.1.6/30         Up     Up        4096
Gig1/4        172.18.1.10/30        Up     Up        4097
Gig4/5        10.10.20.1/30         Up     Down      4098
Gig4/6        10.10.22.1/30         Up     Down      4099
Lag1000       172.18.1.14/30        Up     Up        4100
Loop0         10.10.40.4/32         Up     Up        0
Mgmt0         172.17.4.44/24        Up     Up        0
```

The following example displays the contents of the routing table. It shows that some of the route entries are for locally connected interfaces ("directly connected"), while some of the other routes are learned from OSPF.

```
Switch 8100fl#show ip route
Codes: R - RIP derived, O - OSPF derived, C - connected,
       S - static,
       * - candidate default route, IA - OSPF inter area route,
       E1 - OSPF external type 1 route, E2 - OSPF external type 2 route,
       N1 - OSPF NSSA external type 1 route,
       N2 - OSPF NSSA external type 2 route
       K  - Kernel route remnant after rosrd restart
       A  - Aggregate route

Gateway of last resort is 172.17.4.1 to network 0.0.0.0

S    0.0.0.0 [5/0] via 172.17.4.1, 1d20m49s, Mgmt0
     4.0.0.0/32 is subnetted, 1 subnets
C       4.4.4.4 is directly connected, Loop0
     172.17.0.0/24 is subnetted, 1 subnets
C       172.17.4.0 is directly connected, Mgmt0
     172.18.0.0/30 is subnetted, 4 subnets
O       172.18.1.0 [10/3] via 172.18.1.5, 1d19m39s, Gig1/3
                   [10/3] via 172.18.1.9, 1d19m39s, Gig1/4
C       172.18.1.4 is directly connected, Gig1/3
C       172.18.1.8 is directly connected, Gig1/4
C       172.18.1.12 is directly connected, Lag1000

        Number of Routes: 7
```

To display additional IP information, enter the following commands in Privileged Exec mode:

Table 14-1. Configuring an Interface for RIP

Command	Action
show ip arp	Show ARP table entries.
show ip traffic	Show traffic statistics.
show ip ospf	Show OSPF information.
show ip protocols administrative-distance	Show IP routing protocol parameters and statistics
show ip rxstats <slot>	Show Layer 3 receive statistics

Setting Memory Thresholds

The routing information base (RIB) is stored in the switch's memory. You can use the **ip table-partition** command to configure the percentage of the available memory that is used for storing IP route entries. (For the command to take effect, the interface modules in the system need to be rebooted.)

When the threshold level you configure is reached, no new routes are added.

For example, use the **ip table-partition** command to allocate 80% of the RIB for IP routes.

```
Switch 8100fl(config)#ip table-partition percent 80
```

Note ProCurve recommends table partition settings between 10% and 90%.

15

Time Configuration

Contents

Overview

This chapter discusses how to set time on the ProCurve Switch 8100fl and how to use the pool of Network Time Protocol (NTP) servers to set the clock to Universal Coordinated Time (UTC).

Setting the Date and Time

To set the date and time on the ProCurve Switch 8100fl, use the **clock set** command in Privileged Exec mode. Enter the time as UTC time. Once you enter the command, the date and time are written to the hardware real-time clock—not to scratchpad. Therefore you do not need to save the configuration to make it effective.

To set the system date and time (assuming you are at the Exec (>) prompt):

1. Enter the **enable** command to get to Privileged Exec mode in the CLI.
2. Enter the following commands to set the system time and date and to verify your settings.

```
Switch 8100fl#clock set <HH:MM:SS> <1...31> <month name>
<Year>

Switch 8100fl#show clock [details]
```

Notes

If NTP is running, using the **clock set** command will reset NTP.

Since the ProCurve Switch 8100fl displays system time in Greenwich Mean Time (GMT), also known as Coordinated Universal Time (UTC), you must enter the hours portion of the time with the correct UTC value. The switch does not convert your local timezone into the appropriate UTC timezone. For more information on converting local timezones into UTC zones, see: http://times.clari.net.au/.□

Changes made in Exec and Privileged Exec mode (such as setting the system date and time) do not require saving the scratchpad to the active or startup configuration to become part of the running configuration.

For example, the following command sets the date to March 5, 2006 and the time to 8:10:40 A.M., Pacific Standard Time:

```
Switch 8100fl#clock set 16:10:40 5 march 2006
```

In this example, to convert PST to UTC, first convert the local time into a 24-hour clock format, 08:10:40. Then add 8 to convert to UTC. This gives the time conversion as 16:10:40. A slightly more complicated conversion occurs when adding 8 (for PST) forces the 24-hour clock into the next morning. In this case, make sure you enter the date correctly - as the next day.

For example, to convert 8:30 PM on March 5 from PST to UTC, you would convert 8:30 PM to 20:30 hours. Adding 8 to convert to UTC gives you 04:30 hours, on March 6.

Use the **show clock** command in Privileged Exec mode to display the current date and time on the ProCurve Switch 8100fl. For example:

```
Switch 8100fl#show clock
*16:10:40 UTC Wed Mar 5 2006
```

Using NTP

Network Time Protocol (NTP) synchronizes the clocks on devices in a network to UTC, ensuring consistent and accurate times across network operations. If your local ProCurve 8100fl time differs too much from the time held by the pool of NTP servers (typical of when the ProCurve 8100fl is starting up), NTP will force the clock to the NTP time. Normally, NTP will make minor adjustments to keep time by speeding up or slowing down the ProCurve 8100fl clock. For more information on NTP, refer to RFC 1305.

Clock Synchronization

To cause the NTP client to synchronize its clock with an NTP server, use the **ntp** commands in Configuration mode. The **ntp server** command configures the ProCurve 8100fl to be an NTP peer. For example, the following command causes the NTP client to synchronize its clock with the NTP server with IP address 10.100.1.10:

```
Switch 8100fl(config)#ntp server 10.100.1.10
```

Optionally, you can configure an authentication key (using the key option) shared with NTP peers. You can also specify which version (1, 2, or 3) of NTP to use (with the version option). You can also configure multiple NTP servers and specify (with the prefer option) that a specific server, as long as it is accessible, should be used for time synchronization.

The **ntp peer** command defines the clients. (Essentially, the switch that is configured as the NTP server will share its time with its peers.) For example, the following command causes the NTP client with IP address 10.100.1.2 to synchronize its clock with the ProCurve 8100fl switch (configured in the **ntp server** command):

```
Switch 8100fl(config)#ntp peer 10.100.1.2
```

Figure 15-1 illustrates the relationships of peers, servers, and clients (which can be servers).

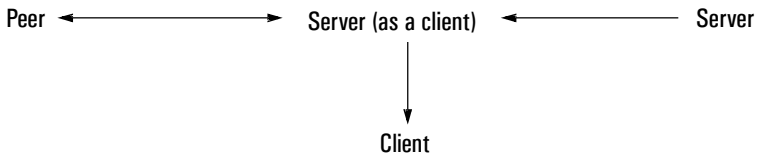


Figure 15-1. NTP Servers, Peers, and Clients

You can also specify which version (1, 2, or 3) of NTP to use (with the version option). You can also configure multiple NTP peers and specify (with the prefer option) that a specific peer, as long as it is accessible, should be used for time synchronization.

The following example configures an NTP pool of peers and identifies a peer that will get its time from the NTP servers.

```
Switch 8100fl(config)#ntp peer 10.220.1.7 key 22
Switch 8100fl(config)#ntp server 10.100.1.2 prefer key 22
Switch 8100fl(config)#ntp server 10.100.1.5 key 22
```

The following example shows a typical configuration of NTP servers. (The * symbol indicates the peer synchronized against, the = symbol identifies a server, and the + symbol identifies a peer.) Use the **detail** parameter to display all of the NTP statistics.

```
Switch 8100fl(config)#show ntp associations
remote          local          st  poll  reach  delay   offset    disp
==============================================================================
*10.200.1.2   10.203.11.27   3     64   377  0.00031  0.024771  0.00006
=10.200.1.5   10.203.11.27   3     64   377  0.00029  0.030663  0.00012
+10.203.11.27 0.0.0.5        0     64     0  0.00000  0.000000  0.00000
```

Monitoring NTP Status

The following example shows the NTP status of the peer at 10.200.1.2:

```
Switch 8100fl(config)#show ntp status
system peer:          10.200.1.2
system peer mode:     client
leap indicator:       00
stratum:              4
precision:            -18
root distance:        0.08955 s
root dispersion:      0.09282 s
reference ID:         [10.200.1.2]
reference time:       c21144bd.7e63736c Thu, Mar 6 2003 3:53:33.493
system flags:         monitor ntp kernel stats pps
jitter:               0.000397 s
stability:            1.837 ppm
broadcastdelay:       0.003998 s
authdelay:            0.000000 s
```

Note If your switch is not synchronized, the NTP status will not display.

16

SNMP Configuration

Contents

Overview

The Simple Network Management Protocol (SNMP) is an application layer protocol used to monitor and manage TCP/IP-based networks. It provides for the storage and exchange of management information.

The ProCurve Switch 8100fl supports the following SNMP versions:

- SNMP Version 1 (SNMPv1) (RFC 1157)
- SNMP Version 2c (SNMPv2c) (RFC 1901, RFC 1905, and RFC 1906)

Both versions of SNMP can coexist in the same managed network (RFC 2576). You should configure the switch to run the SNMP version(s) supported by the SNMP management stations. You can run both SNMP versions on the ProCurve Switch 8100fl, depending on the one used by the SNMP management stations. (For additional information on the different SNMP versions, refer to the RFCs for each version.)

You can use the CLI to configure various SNMP tasks. This chapter described how to perform these tasks. It contains the following sections:

- To configure access to the Management Information Base (MIB) objects, refer to "Configuring Access to MIB Objects" on page 16-3.
- To configure notifications, refer to "Configuring SNMP Notifications" on page 16-5.
- To configure SNMP MIB modules and for a list of MIB modules supported by the ProCurve Switch 8100fl, refer to "MIB Modules" on page 16-6.
- To information on verifying and troubleshooting your SNMP configuration, refer to "Displaying SNMP Information" on page 16-9.
- For a list of SNMP notifications, refer to "SNMP Notifications" on page 16-11.

Configuring Access to MIB Objects

The ProCurve Switch 8100fl supports many of the standard networking SNMP MIB modules. Each module is a collection of managed objects which can be accessed by the SNMP management stations. (For a list of MIB modules supported by the ProCurve Switch 8100fl, refer to "MIB Modules" on page 16-6).

SNMP management stations send SNMP SET and GET requests for the management objects stored in the MIB modules. The ProCurve Switch 8100fl run an SNMP agent, which is a software process that listens for these SNMP requests on UDP port 161. In SNMPv1 and v2c, the SNMP managers provide a community string (or password) when they send their requests. If the switch recognizes the community string, it processes the request. If it doesn't recognize the community string, it discards the message and increments the bad community name error counter (which can be viewed through the **show snmp** command).

The following sections describe how to configure access to the MIB modules for each SNMP version.

Note

By default, the ProCurve Switch 8100fl does not listen for traffic on port 161. When you configure a community, SNMP starts listening for traffic on this port and accumulates statistics that you can view using the **show snmp** command.

Configuring SNMP Access

Following are the tasks for configuring SNMP access if you are running SNMPv1 and v2c:

Configure a community string. By default, the ProCurve Switch 8100fl has a read-only community string called “public”.

Configure the agent’s identity.

Configuring Community Strings

To run SNMPv1 and v2c on the switch, you must have at least one community string. By default, the ProCurve Switch 8100fl has a read-only community string called “public”.

When you define an SNMP community string, you also need to specify its access level, which is either read-only (allows only SNMP GETs), or read-write (allows SNMP SETs and GETs). In the following example, separate community strings are defined for read-only access and for read-write access:

```
Switch 8100fl(config)#snmp community red ro
Switch 8100fl(config)#snmp community blue rw
```

An SNMP manager that sends a GET request for a MIB object can provide the community string *red* or *blue*; and an SNMP manager that sends a SET request should provide the community string *blue* (in this example).

Configuring the SNMP Agent

You can use the CLI to set certain MIB objects, such as those that describe the agent’s identity, as shown in the following example:

```
Switch 8100fl(config)#snmp contact IT dept
Switch 8100fl(config)#snmp location building 1 closet
Switch 8100fl(config)#snmp chassis-id s/n12345
Switch 8100fl(config)#snmp mib <mib name>
Switch 8100fl(config)#snmp trap-source loopback0
```

The example sets the MIB objects sysContact to *IT dept*, sysLocation to *building 1 closet*, and hp-switch-fl-series-inventory-mib ChassisId to *s/n12345*.

Configuring SNMP Notifications

The ProCurve Switch 8100fl sends notifications to pre-defined targets. The targets are the SNMP management stations that receive the notifications. Notifications inform the SNMP managers about conditions on the network, such as an error condition or an authentication failure.

The tasks for configuring SNMP notifications are as follows:

Specifying the targets. This is required.

Configuring the notification's source address.

Specifying the Notification Targets

To send SNMP notifications, you need to specify the following:

The targets that will receive the notifications

A community string

Targets are defined by their IP addresses. Each target that is defined receives a copy of the notifications generated and sent by the ProCurve 8100fl agent.

In addition, you need to specify a community string for the notifications. For security reasons, the community strings in notifications should be different from the read/write community strings. So when the ProCurve Switch 8100fl sends notifications, unauthorized users who capture the notifications will not be able to use the community string to access the MIB modules.

In the following example, the notifications will be sent to the target with address 10.10.10.1 (and a community string of "western"):

```
Switch 8100fl(config)#snmp community community1 ro
Switch 8100fl(config)#snmp host 10.10.10.1 western
```

Note

If the IP address of the target is more than one hop away from the switch, configure the switch with a static route to the target. If the switch is rebooted, the static route allows a cold-start notification to be sent to the target. Without a static route, the cold-start notification is lost while the routing protocols are converging.

Enabling/Disabling SNMP

By default, SNMP is active and monitoring UDP port 161. You can turn off SNMP by entering:

```
Switch 8100fl(config)#no snmp active
```

To restart SNMP, enter:

```
Switch 8100fl(config)#snmp active
```

SNMP will issue a warmStart notification and resume monitoring port 161.

Note

The **snmp active** command runs independently of your SNMP configuration. You must still configure SNMP. The **no snmp active** command is a useful alternative to negating lines in your configuration file when you need to disable SNMP.

MIB Modules

The ProCurve Switch 8100fl supports the following MIB modules. You can use these modules with any SNMP version.

Table 16-1. Release 1.0 Supported MIBs

MIB Name	RFC Standard
SNMPv2-MIB	RFC 1907
IP-MIB	RFC 2011
TCP-MIB	RFC 2012
UDP-MIB	RFC 2013
IP-FORWARD-MIB	RFC 2096
IF-MIB	RFC 2863
ENTITY-MIB	RFC 2737
HP-SWITCH-FL-SERIES-INVENTORY-MIB	n/a

Loading MIBs

The order in which you load MIBs into an application is important. The following MIB modules should already be loaded (so you do not have to):

SNMPv2-SMI

SNMPv2-TC

SNMPv2-CONF

SNMP-FRAMEWORK-MIB

Some of the following list of IETF standard MIB modules may already be loaded, so you do not need to load them again (unless they are newer versions). Load them in the order shown—with the HP Switch proprietary MIB module at the end.

Caution

Do not inadvertently overwrite an older version of a MIB module with a newer one without first backing up that module. Devices from other vendors that support older versions, may not work with the newer version of the module.

SNMPv2-MIB

IF-MIB

IP-MIB

TCP-MIB

UDP-MIB

IP-FORWARD-MIB

ENTITY-MIB

HP-SWITCH-FL-SERIES-INVENTORY-MIB

Enabling/Disabling MIB Modules

All MIB modules are enabled (or online) by default. Use the **snmp mib ?** command to display supported MIBs. For this release, the ProCurve Switch 8100fl supports the following MIBs:

```
Switch 8100fl(config)#snmp mib ?
 entity-mib                 - rfc2737 specific system details
 if-mib                   - Interface status and generic counters per RFC 2863
 ip-forward-mib             - IPv4 CIDR forwarding database per RFC 2096
 ip-mib                     - Counters for IP and ICMP version 4 per RFC 2011
 hp-switch-fl-series-inventory-mib  - HP switch inventory details
 snmpv2-mib               - System detail, SNMPv1/v2c/v3 counters per RFC 1907
 tcp-mib                    - Counters for Transmission Control Protocol, IP
version 4 per RFC 2012
 udp-mib                  - Counters for User Datagram Protocol, IP version 4
per RFC 2013
```

All MIB modules can be accessed by SNMP management stations that provide the correct community strings. You may want to provide access to a smaller set of MIB modules. To do so, you can "disable" MIB modules by using the **no snmp mib** command as shown in the following example:

```
Switch 8100fl(config)#no snmp mib hp-switch-fl-series-
inventory-mib
```

You can then view the MIB modules, including their status, as shown in "Displaying SNMP Information" on page 16-9.

Displaying SNMP Information

The **show snmp** command is used to display SNMP configuration information. The status of the notifications are listed at the bottom of the output.

```
Switch 8100fl(config)#show snmp
agent operational 343 seconds
In/out packets: 0/0 last:
last error occurred on:
Bad version : 0
Bad community name: 0
Bad community uses: 0
ASN Parse Errors : 0
Too bigs : 0/0
No such name : 0/0
Bad value : 0/0
Read Only : 0
General Error : 0/0
command distribution
Get requests : 0/0 (0.00%)
GetNext requests : 0/0 (0.00%)
Set requests : 0/0 (0.00%)
Get responses : 0/0 (0.00% of in packets)
Variables had/set : 0/0
Silent/Proxy drops : 0/0
Traps sent/received : 2/0 last: Fri Mar 7 00:49:27 2005
Switch 8100fl(config)#
```

For information on snmp location, contact and host, use the commands that are shown below:

```
Switch 8100fl(config)#show snmp location
location: Bunker Hill Lane
Switch 8100fl(config)#
Switch 8100fl(config)#show snmp contact
contact: Umesh
Switch 8100fl(config)#
Switch 8100fl(config)#show snmp host
Address community port
10.200.118.245 private 162
Switch 8100fl(config)#
```

For information on MIB versions and status, use the show snmp mib-modules command as shown below:

```
Switch 8100fl(config)#show snmp mib-modules
SNMP AGENT MIB Registry
Last Modified: 0 days 0 hours 0 min 4 secs
Index  Name                                  Version    Status
-----  ----                                  -------    ------
1      SNMPv2-MIB                            1907       online
2      IF-MIB                                2863       online
3      IP-MIB                                2011       online
4      TCP-MIB                               2012       online
5      UDP-MIB                               2013       online
6      IP-FORWARD-MIB                        2096       online
7      ENTITY-MIB                            2737       online
8      HP-SWITCH-FL-SERIES-INVENTORY-MIB     --         online
```

Troubleshooting SNMP

SNMP misconfigurations typically generate the following error when you enter the **show snmp** command:

```
Switch 8100fl(config)#show snmp
%SNMP agent not enabled
Switch 8100fl(config)#
```

If you receive this error:

Make sure you have configured a community string (see "Configuring Community Strings" on page 16-4).

Make sure SNMP is enabled (see "Enabling/Disabling SNMP" on page 16-6).

Use the **show snmp** command and examine the output to verify your configuration.

SNMP Notifications

Table 16-2 lists the notifications that SNMP generates for this release.

Table 16-2. ProCurve 8100fl SNMP Notifications

N	Notification Type (OID with name)	MIB	VB List
Notifications from standard MIB modules			
1	coldStart OID: 1.3.6.1.6.3.1.1.5.1	SNMPv2-MIB	No List
2	warmStart OID: 1.3.6.1.6.3.1.1.5.2	SNMPv2-MIB	No list
3	linkDown OID: 1.3.6.1.6.3.1.1.5.3	IF-MIB	1: ifIndex 2: ifAdminStatus 3: ifOperStatus 4: ifDescr
4	linkUP OID: 1.3.6.1.6.3.1.1.5.4	IF-MIB	1: ifIndex 2: ifAdminStatus 3: ifOperStatus 4: ifDescr
5	authenticationFailure OID: 1.3.6.1.6.3.1.1.5.5	SNMPv2-MIB	No List
Notifications from HP switch proprietary MIB module			
6	hotSwapOut OID: 1.3.6.1.4.1.11.2.14.11.8.1.33.1.2.0.8	HP-SWITCH-FL-SERIES-INVENTORY-MIB	1: entPhysicalDescr 2: entPhysicalIndex 3: Trapdescription
7	hotSwapIn OID: 1.3.6.1.4.1.11.2.14.11.8.1.33.1.2.0.7		1: entPhysicalDescr 2: entPhysicalIndex 3: Trapdescription
8	TemperatureExceeded OID: 1.3.6.1.4.1.11.2.14.11.8.1.33.1.2.0.5		1: entPhysicalDescr 2: entPhysicalIndex 3: Trapdescription
9	fanFailed OID: 1.3.6.1.4.1.11.2.14.11.8.1.33.1.2.0.3		1: entPhysicalDescr 2: entPhysicalIndex 3: Trapdescription
10	PowerSupplyFailed OID: 1.3.6.1.4.1.11.2.14.11.8.1.33.1.2.0.1		1: entPhysicalDescr 2: entPhysicalIndex 3: Trapdescription

— This page is intentionally unused. —

17

Performance Monitoring

Contents

Overview

The ProCurve Switch 8100fl performs as a full wire-speed Layer 2, Layer 3, and layer-4 switching router. As packets enter the switch, Layer 2, 3, and 4 flow tables are populated on each interface module. The flow tables contain information on performance statistics and traffic forwarding. Thus the ProCurve Switch 8100fl provides the capability to monitor performance at layers 2, 3, and 4.

This chapter discusses the following topics dealing with monitoring performance and traffic:

- Show commands
- Debug commands
- Clear commands
- Logging Messages

Show Commands

Layer 2 performance information is accessible to SNMP through MIB-II and can be displayed by using the **show snmp** command in the CLI. Layer 3 and 4 performance statistics can be displayed by using the **show statistics** command in the CLI.

To display configuration and system information on the ProCurve Switch 8100fl, enter the following commands in Privileged Exec mode:

Command	Function
show aaa method-lists	Show authentication, authorization, and accounting (aaa) method-lists
show aaa servers	Show security server information for RADIUS or TACACS+ servers.
show access-lists	Show access-list entries
show active	Show current operating configuration
show arp	Show ARP entries

show bootvar	Show boot related information
show bridge fib	Show bridging information
show clock	Show information about the system clock
show configuration	Show configuration data in flash
show device-logging	Show how the terminal, host, and buffer are configured for logging
show environment	Show environmental conditions of the chassis
show errors	Show configuration commands that have failed
show hardware	Show information on installed hardware components
show history	Show the command history buffer
show images	Show the system files stored on the system
show interfaces	Show information on all configured interfaces
show ip	Show IP entries
show ip arp	Show IP ARP entries
show ip interface	Show current status of IP interfaces
show ip ospf	Show Open Shortest Path First (OSPF) protocol information
show ip prefix-list	Show specified prefix lists
show ip protocols	Show all IP routing protocol information
show ip route	Show current status of IP routes
show ip rxstats	Show L3 receive statistics
show ip traffic	Show IP traffic statistics for specified interface modules
show l2-vlan-translate	Show L2 VLAN translation statistics
show l2acl	Show L2 access-list entries and resource usage
show lacp	Show LACP related info
show lag	Show LAG parameters
show logging	Show log buffer
show memory	Show process/system memory consumption
show module	Show interface module information

show ntp	Show Network Time Protocol (NTP) information
show parse-errors	Show errors from last parsed configuration
show pinger	Show pinger gateway information
show policy	Show IP policies
show port	Show Layer 2 port related information
show process	Show resource usage per process
show radius servers	Show Remote Access Dial-in User Service (RADIUS) server information
show redundancy	Show the status of redundant modules
show reload	Show pending reload information for the entire chassis or for specified interface modules
show route-map	Show route-map entries
show running-config	Show current operating configuration
show scratchpad	Show configuration data not applied in temporary storage
show snmp	Show Simple Network Management Protocol (SNMP) agent information
show spolicy-input-map	Show all input spolicy maps and their matching criteria
show spolicy-output-map	Show all output spolicy maps and their matching criteria
show startup-config	Show contents of startup configuration
show statistics	Show interface statistics for configured ports
show stp	Show default Spanning Tree Protocol (STP) information
show system-mac	Show system MAC address
show tacacs servers	Show Terminal Access Controller Access Control System (TACACS) server information
show tech-support	Shows current configuration and process status that is of interest to technical support personnel
show terminal	Show terminal line parameters
show upgrade-status	Show the software upgrade status
show users	Show active user sessions
show version	Show system software version

show vlan	Show VLAN parameters
show vrrp	Show status of VRRP groups

Note All the show commands are accessible at the Privileged Exec mode. Many show commands are accessible from various configuration modes, and a limited number of show commands are available at the Exec mode level.

Debug Commands

To gather information on selected processes and to control tracing, enter the following commands from the Privileged Exec mode:

Command	Function
debug aaa	Gather debugging information and statistics on Authentication, Authorization, and Accounting (aaa)
debug radius	Gather Remote Access Dial In User Service (RADIUS) protocol information
debug stp	Gather Spanning Tree Protocol (STP) information
debug tacacs	Gather Terminal Access Controller Access Control System (TACACS+) protocol information
debug vrrp	Gather Virtual Redundant Router Protocol (VRRP) debug information

Clear Commands

To delete data from the system, use the following commands from the Privileged Exec mode:

Command	Function
clear access-list	Clear access-list counters for a specified Access Control List.
clear arp	Clear the Address Resolution Protocol (arp) entry IP address.
clear arp-cache	Clear the Address Resolution Protocol (arp) cache.
clear bridge mac-table	Delete MAC entries from the MAC address table.
clear errors	Empty the error-config file of all the commands that failed to run.
clear history	Clear the EXEC level history list.
clear ip ospf	Clear the IP OSPF counters, processes, and commands.
clear ip traffic	Clear IP traffic statistics.
clear l2acl	Clear layer 2 acl counters for a specified Access Control List and port.
clear lacp	Clears Link Access Control Protocol (LACP) data unit statistics.
clear logging	Empty the log buffer.
clear parse-errors	Reset the CLI parser error log.
clear screen	Wipe the terminal screen clean and reposition the cursor at the top left corner.
clear statistics	Delete Ethernet statistics for the specified port.

Error Reporting and Message Logging

Individual file system commands will report application specific errors as part of their normal output. ERRLOG messages will be generated on the following events:

- A physical file system becomes full
- A user attempts to overwrite or remove a read-only system file
- File system operation fails
- Any other catastrophic failure

The ProCurve Switch 8100fl logs system error messages by default. This section discusses how you can control where these messages are stored, how many messages are kept, what level of severity the messages will be, and how to turn off logging.

Disabling/Enabling Message Logging

To disable message logging, enter:

```
Switch 8100fl#no logging terminal
```

To re-enable message logging if it has been disabled, enter:

```
Switch 8100fl#logging terminal [all]
```

To modify the logging settings, instead of specifying **all**, specify the name of the process or device from the supported list of logging terminal options.

Displaying Logging Messages

To display log messages from the buffer (instead of the log buffer), use the following command in Privileged Exec mode:

```
Switch 8100fl#show logging [all]
```

To modify the logging display, instead of specifying **all**, specify the name of the process or device from the supported list of logging terminal options.

To display information about the log buffer, enter from Privileged Exec mode:

```
Switch 8100fl#show logging history
```

When logging is buffered, the following Privileged Exec command is useful to display logged messages:

```
Switch 8100fl#show logging
```

The following Privileged Exec command can be used to clear the log buffer:

```
Switch 8100fl#clear logging
```

Displaying Crash Log Files

To display a log file after a crash occurs, enter the following command:

```
Switch 8100fl#dir flash:
```

This will list the files located in the flash directory. The crash log file ends with the suffix "`core`". Depending on which process has crashed, the appropriate process name will be the prefix of that core file name.

Once you locate the crash log *<process_name>*`.core` file, you can use the standard ftp command from the CLI to access the file and troubleshoot the cause of the problem.

Specifying Logging Locations

You can specify that system logs be sent to the system console, local logs, or in a Unix-style syslog format.

When you configure messages to be sent to the console and the logging process is enabled, the messages are displayed on the console after the process that generated them has finished. When the logging process is disabled, messages are sent only to the console.

The syslog format is compatible with 4.3 Berkeley Standard Distribution (BSD) Unix.

To set the locations that receive messages (buffer, console, or syslog respectively), use the following commands in Configuration mode.

Command	Function
logging buffered	Set buffered logging.
logging host	Set host logging. Requires an <ip address>.
logging terminal	Set terminal logging.

Configuring the Syslog Host

Use the following Configuration command to configure the syslog host:

```
Switch 8100fl(config)#logging host <ipaddr> [all]
```

Setting Source Interface for Syslog Messages

To set the source interface for syslog messages, enter the following command:

```
Switch 8100fl(config)#logging source <ipaddr>
```

Note

This command instructs logging to send syslog only on the specified interface, and enables the network management to see a valid source ip address in a syslog message. If you do not specify a host source ip address, the default value would be 0.0.0.0 giving no indication where the message comes from.

Setting the Severity Level of Messages

You can set the message severity level to control the type of messages displayed for the console and each destination.

You can limit the number of messages displayed to the selected device by specifying the severity level of the error messages. To do so, use the following commands in Configuration mode, as needed:

```
Switch 8100fl(config)#logging terminal all
```

You can limit the logging of messages by specifying the alert level as follows:

Emergency	System is unusable (severity = 0)
Alert	Immediate action needed (severity = 1)
Critical	Critical conditions (severity = 2)
Error	Error conditions (severity = 3)

Warning	Warning conditions (severity = 4)
Notification	Normal but significant conditions (severity = 5)
Informational	Informational messages (severity = 6)
Debugging	Debugging messages (severity = 7)

Note The default alert level for the buffered messages is informational, and the default alert level for terminal messages is warning. The default alert level for syslog messages is informational.

Controlling the Size of the Log and Messages

You can set the number of messages that get stored in the history table. By default, messages of the level **warning** and above are stored in the history table even if syslog traps are not enabled. To change level and table size defaults, use the following commands in Configuration mode:

```
Switch 8100fl(config)#logging buffered size <size of buffer>
```

Time-Stamping Messages

By default, the switch time-stamps log messages to enhance real-time debugging and management.

Setting Temperature Thresholds

The management module contains a temperature sensor that monitors temperature on the switch. The sensor generates a Syslog message and an SNMP trap if the temperature exceeds the specified threshold. The default warning and critical temperatures are 72.0 C degrees and 74.0 C degrees respectively. The default shutdown temperature is 78.0 C degrees.

You can use the CLI to specify the warning or critical temperature levels using the following command:

```
Switch 8100fl(config)#set-temperature {warning | critical}
<temperature>
```

For example:

To set a warning level threshold of 66 degrees, enter:

```
Switch 8100fl(config)#set-temperature warning 66
```

The system will generate a Syslog message and an SNMP trap if the temperature exceeds 66 degrees.

Configuring Port Mirroring

The ProCurve Switch 8100fl allows you to monitor performance and activities of ports on the switch using port mirroring.

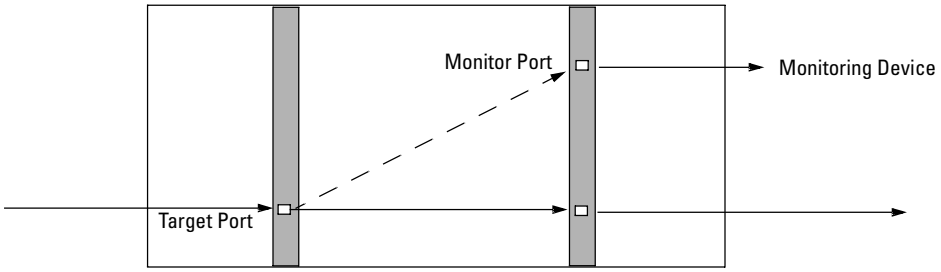


Figure 17-1. Port Mirroring

In Figure 17-1, the target port is mirrored to a port on another interface module. The port to which traffic is mirrored must be activated with the no shutdown command and have no other configuration.

The following example shows you how to configure a ten gigabit ethernet port 1 in slot 5 to be a mirror for the traffic on a ten gigabit ethernet port 1 in slot 6.

```
Switch 8100fl(config)#interface ten 5/1
Switch 8100fl(config-if)#mirror monitor-port int ten 6/1
Switch 8100fl(config-if)#no shutdown

Switch 8100fl(config)#interface TenGigabitEthernet6/1
Switch 8100fl(config-if)#no shutdown
```

Port Mirroring Limitations

As you can see from Figure 17-1, port mirroring adds traffic to the switch. To control the impact on performance, consider the following:

- The port to which traffic is mirrored must not be configured.
- Port mirroring can only be done on incoming traffic.
- Incoming traffic that is dropped at the ingress port is not mirrored.
- Only one port per switch can be mirrored at any one time.
- You can mirror a 10G target port to a 1G monitor port, but traffic volume greater than the monitor port can handle is dropped.
- Increase in traffic cuts the bandwidth of the port being mirrored in half.

— This page is intentionally unused. —

Command Line Index

This index provides an alphabetical listing of all the commands in the CLI that are referenced in this guide.

H

I

K

L

M

N

P

R

S

T

U

V

Index

A

B

C

M

N

O

P

Q

R

S

T

U

V

W

— This page is intentionally unused. —

September 2005

Manual Part Number
5990-8867