ProCurve
Networking by HP

**8200zl**
**6200yl**
**5400zl**
**3500yl**
**2900**

# IPv6 Configuration Guide

## ProCurve Switches

K.13.01
T.13.01

www.procurve.com

hp invent

ProCurve

8212zl Switch

6200yl Switch

Series 5400zl Switches

Series 3500yl Switches

Series 2900 Switches

**January 2008**

K.13.01

T.13.01

# IPv6 Configuration Guide

### Publication Number

5992-3067
January 2008

### Applicable Products

| | |
|---|---|
| ProCurve Switch 2900-24G | (J9049A) |
| ProCurve Switch 2900-48G | (J9050A) |
| ProCurve Switch 3500yl-24G-PWR | (J8692A) |
| ProCurve Switch 3500yl-48G-PWR | (J8693A) |
| ProCurve Switch 5406zl | (J8697A) |
| ProCurve Switch 5412zl | (J8698A) |
| ProCurve Switch 6200yl-24G | (J8992A) |
| ProCurve Switch 8212zl | (J8715A) |

### Trademark Credits

Microsoft, Windows, and Microsoft Windows NT are US registered trademarks of Microsoft Corporation. Java™ is a US trademark of Sun Microsystems, Inc.

### 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

# 3 IPv6 Addressing

## 4 IPv6 Addressing Configuration

## 5 IPv6 Management Features

# Product Publications and IPv6 Command Index

## About Your Switch Manual Set

**Note**

For the latest version of all ProCurve switch documentation, including Release Notes covering recently added features, please visit the ProCurve Networking Web site at **www.procurve.com,** click on **Technical support**, and then click on **Product manuals (all)**.

### Printed Publications

The two publications listed below are printed and shipped with your switch. The latest version of each is also available in PDF format on the ProCurve Web site, as described in the above Note.

- *Read Me First*—Provides software update information, product notes, and other information.
- *Installation and Getting Started Guide*—Explains how to prepare for and perform the physical installation and connect the switch to your network.

### Electronic Publications

The latest version of each publication listed in this section (including the above printed publications) is available in PDF format on the ProCurve Web site, as described in the Note at the top of this page.

The six publications listed below cover all of the switches supported by this manual.

- *Management and Configuration Guide*—Describes how to configure, manage, and monitor basic switch operation.
- *Advanced Traffic Management Guide*—Explains how to configure traffic management features such as VLANs, MSTP, QoS, and Meshing.
- *Multicast and Routing Guide*—Explains how to configure IGMP, PIM, IP routing, and VRRP features.
- *Access Security Guide*—Explains how to configure access security features and user authentication on the switch.
- *IPv6 Configuration Guide*—Describes the IPv6 protocol operations that are supported on the switch.
- *Release Notes*—Describe new features, fixes, and enhancements that become available between revisions of the main product guide.

The two publications listed below support all of the switches covered by this manual *except* the ProCurve Series 2900 switches:

- *Command Line Interface Reference Guide*—Provides a comprehensive description of CLI commands, syntax, and operations.
- *Event Log Message Reference Guide*—Provides a comprehensive description of event log messages.

# IPv6 Command Index

This index provides a tool for locating descriptions of individual IPv6 commands covered in this guide.

**Note**

A link-local address must include **%vlan< *vid* >** without spaces as a suffix. For example:

```
fe80::110:252%vlan20
```

The index begins on the next page.

*A link-local address in these commands must include **%vlan< *vid* >** as a suffix. For example, `fe80::110:252%vlan20`.

# 1

# Getting Started

## Contents

# Introduction

This guide is intended for use with the following switches:

- ProCurve Switch 8200zl series
- ProCurve Switch 5400zl series
- ProCurve Switch 3500yl and 6200yl series
- ProCurve Switch 2900 series

It describes how to use the command line interface (CLI) to configure, manage, monitor, and troubleshoot switch operation. For an overview of other product documentation for the above switches, refer to *"Product Documentation"* on page ix. You can download documentation from the ProCurve Networking web site, **www.procurve.com**.

# Conventions

This guide uses the following conventions for command syntax and displayed information.

## Command Syntax Statements

***Syntax:*** ip < default-gateway < *ip-addr* >> | routing >

***Syntax:*** show interfaces [*port-list*]

- Vertical bars ( | ) separate alternative, mutually exclusive elements.
- Square brackets ( [ ] ) indicate optional elements.
- Braces ( < > ) enclose required elements.
- Braces within square brackets ( [ < > ] ) indicate a required element within an optional choice.
- Boldface indicates use of a CLI command, part of a CLI command syntax, or other displayed element in general text. For example:

  "Use the **copy tftp** command to download the key from a TFTP server."
- Italics indicate variables for which you must supply a value when executing the command. For example, in this command syntax, you must provide one or more port numbers:

  ***Syntax:*** telnet < *ipv6-address* >

## Command Prompts

In the default configuration, your switch displays a CLI prompt similar to the following example:

```
ProCurve 8212zl#
```

To simplify recognition, this guide uses **ProCurve** to represent command prompts for all switch models. For example:

```
ProCurve#
```

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

## Screen Simulations

**Displayed Text.** Figures containing simulated screen text and command output look like this:

```
ProCurve> show version
Image stamp:   /sw/code/build/info
               January 14, 2008 13:43:13
               K.13.01
               243

ProCurve>
```

**Figure 1-1. Example of a Figure Showing a Simulated Screen**

In some cases, brief command-output sequences appear without figure identification. For example:

```
ProCurve(config)# clear public-key
ProCurve(config)# show ip client-public-key
show_client_public_key: cannot stat keyfile
```

## Configuration and Operation Examples

Unless otherwise noted, examples using a particular switch model apply to all switch models covered by this guide.

## Keys

Simulations of actual keys use a bold, sans-serif typeface with square brackets. For example, the Tab key appears as **[Tab]** and the "Y" key appears as **[Y]**.

# Sources for More Information

This guide covers features related to IPv6 operation in software release K.13.01, and includes an IPv6 command index on page xi.

For information about switch operation and features not covered in this guide, refer to the switch publications listed in this section.

**Note**

For the latest version of all ProCurve switch documentation referred to below, including Release Notes covering recently added features, visit the ProCurve Networking web site at **www.procurve.com**, click on **Technical support**, and then click on **Product Manuals (all)**.

- Software Release Notes—*Release Notes* are posted on the ProCurve Networking 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
- Product Notes and Software Update Information—The printed *Read Me First* shipped with your switch provides software update information, product notes, and other information.
- *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 also 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.
- *Management and Configuration Guide*—Use this guide for information on topics such as:
  - various interfaces available on the switch
  - memory and configuration operation
  - interface access
  - IP addressing
  - time protocols
  - port configuration, trunking, traffic control, and PoE operation
  - Redundant management
  - SNMP, LLDP, and other network management topics
  - file transfers, switch monitoring, troubleshooting, and MAC address management

- *Advanced Traffic Management Guide*—Use this guide for information on topics such as:
  - VLANs: Static port-based and protocol VLANs, and dynamic GVRP VLANs
  - spanning-Tree: 802.1D (STP), 802.1w (RSTP), and 802.1s (MSTP)
  - meshing
  - Quality-of-Service (QoS)
  - Access Control Lists (ACLs)
- *Multicast and Routing Guide*—Use this guide for information on topics such as:
  - IGMP
  - PIM (SM and DM)
  - IP routing
  - VRRP
- *Access Security Guide*—Use this guide for information on topics such as:
  - Local username and password security
  - Web-Based and MAC-based authentication
  - RADIUS and TACACS+ authentication
  - SSH (Secure Shell) and SSL (Secure Socket Layer) operation
  - 802.1X access control
  - Port security operation with MAC-based control
  - Authorized IP Manager security
  - Key Management System (KMS)
- *IPv6 Configuration Guide*—Use this guide for information on topics such as:
  - Overview of IPv6 operation and features supported in software release K.13.01
  - Configuring IPv6 addressing
  - Using IPv6 management, security, and troubleshooting features
- Feature Index—The following software guides for your switch include an index of non-IPv6 features (and where to find them). This index immediately preceeds the first chapter in each guide listed.
  - *Management and Configuration Guide*
  - *Advanced Traffic Management Guide*
  - *Access Security Guide*
  - *Multicast and Routing Guide*

## Getting Documentation From the Web

To obtain the latest versions of documentation and release notes for your switch:

1. Go to the ProCurve Networking web site at

   **www.procurve.com**

2. Click on **Technical support**.
3. Click on **Product manuals**.
4. Click on the product for which you want to view or download a manual.

If you need further information on ProCurve switch technology, visit the ProCurve Networking web site at:

**www.procurve.com**

## Online Help

### Menu Interface

If you need information on specific parameters in the menu interface, refer to the online help provided in the interface. For example:

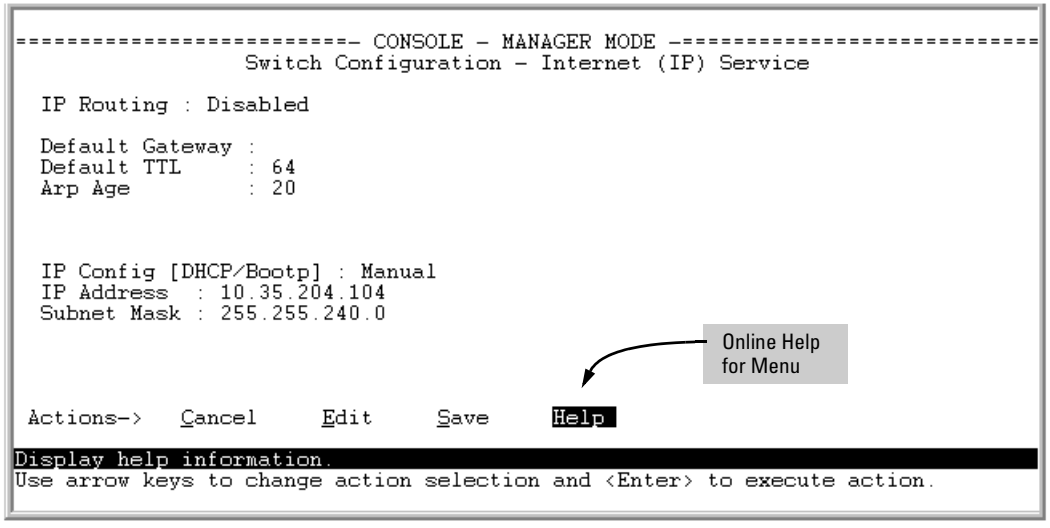



**Figure 1-2. Online Help for Menu Interface**

## Command Line Interface

If you need information on a specific command in the CLI, type the command name followed by **help**. For example:

```
ProCurve# write help
Usage: write <memory|terminal>

Description: View or save the running configuration of the switch.
             write terminal - displays the running configuration of the
                              switch on the terminal
             write memory   - saves the running configuration of the
                              switch to flash. The saved configuration
                              becomes the boot-up configuration of the switch
                              the next time it is booted.
```

**Figure 1-3. Example of CLI Help**

## Web Browser Interface

If you need information on specific features in the ProCurve Web Browser Interface, use the online Help. You can access the Help by clicking on the question mark button in the upper right corner of any of the web browser interface screens.


**Figure 1-4. Button for Web Browser Interface Online Help**

**Note**

To access the online Help for the ProCurve web browser interface, you need either ProCurve Manager (version 1.5 or greater) installed on your network or an active connection to the World Wide Web. Otherwise, Online help for the web browser interface will not be available.

# To Set Up and Install the Switch in Your Network

Use the ProCurve *Installation and Getting Started Guide* (shipped with the switch) for the following:

- Notes, cautions, and warnings related to installing and using the switch and its related modules
- Instructions for physically installing the switch in your network
- Quickly assigning an IP address and subnet mask, set a Manager password, and (optionally) configure other basic features.
- Interpreting LED behavior.

For the latest version of the *Installation and Getting Started Guide* for your switch, refer to “Getting Documentation From the Web” on page 1-6.

# 2

# Introduction to IPv6

## Contents

# Migrating to IPv6

To successfully migrate to IPv6 involves maintaining compatibility with the large installed base of IPv4 hosts and routers for the immediate future. To achieve this purpose, software release K.13.01 supports dual-stack (IPv4/IPv6) operation and connectons to IPv6-aware routers for routing IPv6 traffic between VLANs and across IPv4 networks.

**Note**

Software release K.13.01 supports traffic connections with IPv6-aware routers, but does not support IPv6 routing operation in the switches covered by this guide.

Beginning with software release K.13.01, the switches covered by this guide support the following IPv6 protocol operations:

- receiving IPv6 traffic addressed to the switch
- transmitting IPv6 traffic originating on the switch
- switching IPv6 traffic between IPv6 devices connected to the switch on the same VLAN
- concurrent (dual-stack) operation with IPv4 traffic and devices on the same VLAN
- using a connection to an external, IPv6-configured router, forward IPv6 traffic intended for devices on other VLANs and for traffic that must traverse an IPv4 network to reach an IPv6 destination

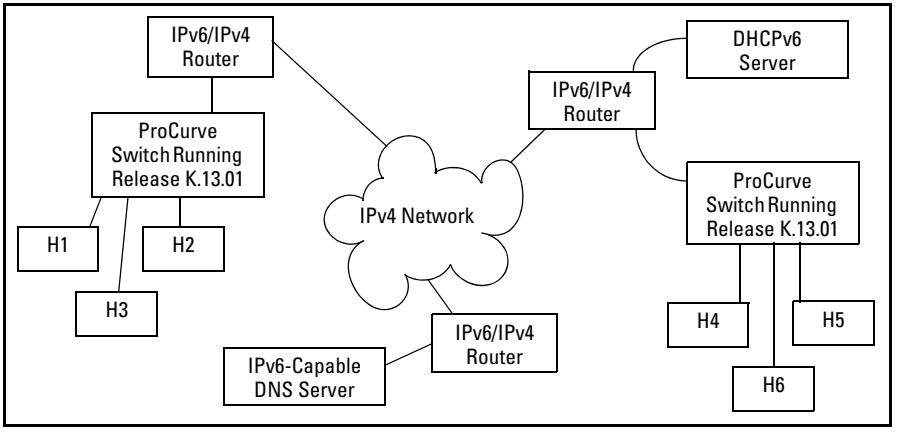



**Figure 2-1. Dual-Stack ProCurve Switches Employed in an IPv4/IPv6 Network**

## IPv6 Propagation

IPv6 is currently in the early stages of deployment worldwide, involving a phased-in migration led by the application of basic IPv6 functionality. In these applications, IPv6 traffic is switched among IPv6-capable devices on a given LAN, and routed between LANs using IPv6-capable routers. Using the IPv6 features in this software release, the switch can operate in an IPv6 network, be managed using an IPv6 management station, and interact with DHCPv6 and IPv6-enabled DNS servers in the same network or accessible through a connection to an IPv6 router.

## Dual-Stack Operation

Since most initial IPv6 deployments are in networks having a mixture of IPv6 and IPv4 hosts software release K.13.01 supports dual- stack IPv4/IPv6 operation. This enables the switch to communicate individually with IPv4 and IPv6 devices with their respective protocols. Thus, IPv4 and IPv6 traffic is supported simultaneously on the same VLAN interface. This means that both IPv4 and IPv6 devices can operate at the same time on a given VLAN.

**Note**

Software release K.13.01 does not include gateways for translation between IPv6 and IPv4 traffic. While IPv4 and IPv6 traffic coexists on the same VLAN, the individual IPv4 and IPv6 devices ignore each other's traffic.

To forward IPv6 traffic from the switch to an IPv6-capable device on a different VLAN, a link to an external IPv6-capable router is needed. Also, IPv6 traffic movement from the switch over IPv4 paths requires routers capable of IPv6 over IPv4 tunneling.

# Connecting to Devices Supporting IPv6 Over IPv4 Tunneling

The switches covered by this guide can interoperate with IPv6/IPv4 devices capable of tunneling IPv6 traffic across an IPv4 infrastructure. Some examples include:

- traffic between IPv6/IPv4 routers(router/router)
- traffic between an IPv6/IPv4 router and an IPv6/IPv4 host capable of tunneling (router/host)

**Note**

Tunneling requires an IPv6-capable router. A switch running software release K.13.01 does not route or tunnel IPv6 traffic. To enable IPv6 traffic from the switch to be routed or to be tunneled across an IPv4 network, it is necessary to connect the switch to an appropriate IPv6-capable router. For more information, refer to the documentation provided with the dual- stack (IPv4/IPv6) routers you plan to use for this purpose.

IPv6 tunneling eases IPv6 deployment by maintaining compatibility with the large existing base of IPv4 hosts and routers. Generally, the various IPv6 tunneling methods enable IPv6 hosts and routers to connect with other IPv6 hosts and routers over the existing IPv4 Internet.

## Information Sources for Tunneling IPv6 Over IPv4

For more information on IPv6 routing and tunneling, refer to the documentation provided with the IPv6/IPv4 routing and tunneling-capable devices in your network. Some other sources of information are:

- RFC 2893: “Transition Mechanisms for IPv6 Hosts and Routers”
- RFC 2401: “Security Architecture for the Internet Protocol”
- RFC 2473: “Generic Packet Tunneling in IPv6 Specification”
- RFC 2529: “Transmission of IPv6 via IPv4 Domains without Explicit Tunnels”
- RFC 3056: “Connection of IPv6 Domains Over IPv4 Clouds”

# Use Model

## Adding IPv6 Capability

IPv6 was designed by the Internet Engineering Task Force (IETF) to improve on the scalability, security, ease of configuration, and network management capabilities of IPv4.

IPv6 provides increased flexibility and connectivity for existing networked devices, addresses the limited address availability inherent in IPv4, and the infrastructure for the next wave of Internet devices, such as PDAs, mobile phones and appliances.

Where IPv4 networks exist today, IPv6 will be phased in over a period of years, requiring an interoperability among the devices using the two protocols. Beginning with software release K.13.01, the switches covered by this guide offer IPv4/IPv6 dual stack operation. This allows full ethernet link support for both IPv4 and IPv6 traffic to move on the same interface (VLAN) without modifying current IPv4 network topologies. This enables you to use IPv6 devices on existing VLANs, manage the switch and other devices from IPv6 management stations, and create "islands" of IPv6 devices as needed to accomodate the need for the IPv6 network growth anticipated for the future.

## Supported IPv6 Operation in Release K.13.01

Software release K.13.01 provides IPv6 protocol and addressing to support host-mode (endpoint) IPv6 operation, including basic layer-2 functionality. IPv6 routing features are not available in this release. However, using a dual-stack (IPv4/IPv6-capable) router, IPv6 traffic can be routed between VLANs and sent across an IPv4 network to another IPv6 device.

(For general information on sending IPv6 traffic across an IPv4 network, refer to “Connecting to Devices Supporting IPv6 Over IPv4 Tunneling” on page 2-5.)

The IPv6 features available in release K.13.01 belong to these general categories:

- switch configuration and management
- security
- IPv6 multicast traffic
- diagnostic and troubleshooting

The next three sections outline the IPv6 features supported in software release K.13.01.

# Configuration and Management

This section outlines the configurable management features supporting IPv6 operation on your ProCurve IPv6-ready switch.

## Management Features

Software release K.13.01 provides host-based IPv6 features that enable the switches covered in this guide to be managed from an IPv6 management station and to operate in both IPv6 and IPv4/IPv6 network environments.

**Note**

Software release K.13.01 does not include IPv6 routing, but interoperates with routers that support IPv6 and IPv4/IPv6 router applications.

## IPv6 Addressing

The switch offers these IPv6 address configuration features:

- SLAAC (stateless automatic address configuration)
- DHCPv6 (stateful automatic address configuration)
- static address configuration

### SLAAC (Stateless Automatic Address Configuration)

Enabling IPv6 on a VLAN automatically enables configuration of a link-local unicast IPv6 address on the VLAN. (No DHCPv6 server is needed.) This address begins with the hexadecimal prefix **fe80**, which is prepended to the interface identifier part of the address. (The interface identifier is generated from the MAC address of the VLAN itself, using the 64-bit extended unique identifier (EUI) method.) This enables the IPv6 nodes on the VLAN to configure and manage the switch.

Enabling IPv6 address autoconfiguration on a VLAN automatically enables automatic configuration of global unicast addresses on the VLAN. After enabling autoconfiguration, a router advertisement (RA) containing an assigned global address prefix must be received on the VLAN from an IPv6 router on the same VLAN. The resulting address is a combination of the prefix

and the interface identifier currently in use in the link-local address. Having a global unicast address and a connection to an IPv6- aware router enables IPv6 traffic on a VLAN to be routed to other VLANs supporting IPv6-aware devices. (Using software release K.13.01, an external, IPv6- aware router is required to forward traffic between VLANs.)

Multiple, global unicast addresses can be configured on a VLAN that receives RAs specifying different prefixes.

### DHCPv6 (Stateful) Address Configuration

The IPv6 counterpart to DHCP client for IPv4 operation is DHCPv6. Global unicast addresses of any scope can be assigned, along with NTP (timep) server addressing when DHCPv6 server support is available through either of the following modes:

- accessible on a VLAN configured on the switch
- accessible through a connection to a router configured with DHCP relay

IPv6 also allows the option of using stateless autoconfiguration or static configuration to assign unicast addresses to a VLAN, while using a DHCPv6 server for time server addressing.

### Static Address Configuration

Statically configuring IPv6 addresses provides flexibility and control over the actual address values used on an interface. Also, if a statically configured link-local address is configured on a static VLAN, the global addresses configured on the VLAN as the result of router advertisements uses the device identifier included in the link-local address. Statically configuring an IPv6 address on a VLAN enables IPv6 on the VLAN if it has not already been enabled.

### Default IPv6 Gateway

Instead of using static or DHCPv6 configuration, a default IPv6 gateway for an interface (VLAN) is determined from the default router list of reachable or probably reachable routers the switch detects from periodic multicast router advertisements (RAs) received on the interface. For a given interface, there can be multiple default gateways, with different nodes on the link using different gateways. If the switch does not detect any IPv6 routers that are reachable from a given interface, it assumes (for that interface) that it can reach only the other devices connected to the interface.

**Note**

In IPv6 for the switches covered in this guide, the default route cannot be statically configured. Also, DHCPv6 does not include default route configuration.)

Refer to “Default IPv6 Router” on page 4-28 and “View IPv6 Gateway, Route, and Router Neighbors ” on page 4-29.

## Neighbor Discovery (ND) in IPv6

The IPv6 Neighbor Discovery protocol operates in a manner similar to the IPv4 ARP protocol to provide for discovery of IPv6 devices such as other switches, routers, management stations, and servers on the same interface. Neighbor Discovery runs automatically in the default configuration and provides services in addition to those provided in IPv4 by ARP. For example:

- Run Duplicate Address Detection (DAD) to detect duplicate unicast address assignments on an interface. An address found to be a duplicate is not used, and the **show ipv6** command displays the address as a **duplicate**.
- Quickly identify routers on an interface by sending router solicitations requesting an immediate router advertisement (RA) from reachable routers.
- If a default router becomes unreachable, locate an alternate (if available on the interface).
- Learn from reachable routers on the interface whether to use DHCPv6 or stateless address autoconfiguration. In the latter case, this also includes the address prefixes to use with stateless address autoconfiguration for routed destinations. (A DHCPv6 server can also be used for "stateless" service; that is, for configuring the interface for access to other network services, but not configuring a global IPv6 unicast address on the interface. Refer to “Neighbor Discovery (ND)” on page 4-17.)
- Use multicast neighbor solicitations to learn the link-layer addresses of destinations on the same interface and to verify that neighbors to which traffic is being sent are still reachable.
- Send a multicast neighbor advertisement in response to a solicitation from another device on the same interface or to notify neighbors of a change in the link- layer address.
- Advertise anycast addresses that may be configured on the device.
- Determine the MTU (Maximum Transmission Unit) for the interface from router advertisements.

For more on IPv6 neighbor discovery applications, refer to “Neighbor Discovery (ND)” on page 4-17.

## IPv6 Management Features

The switch's IPv6 management features support operation in an environment employing IPv6 servers and management stations.With a link to a properly configured IPv6 router, switch management extends to routed traffic solutions. (Refer to the documentation provided for the IPv6 router.) Otherwise, IPv6 management for the switches covered by this guide are dependent on switched management traffic solutions.

### TFTPv6 Transfers

The switch supports these downloads from an IPv6 TFTP server:

- automatic OS download
- manual OS download
- command script download and execution
- configuration file downloads
- public key file downloads
- startup configuration file downloads

The switch supports these uploads to an IPv6 TFTP server

- startup or running configuration upload
- OS upload from flash in current use (primary or secondary)
- event log content upload
- crash log content upload
- output of a specified command

Refer to “TFTP File Transfers Over IPv6” on page 5-15.

### IPv6 Time Configuration

The switch supports both Timepv6 and SNTPv6 time services. Refer to “SNTP and Timep” on page 5-9.

### Telnet6

The switch supports both of the following Telnet6 operations:

- Enable (the default setting) or disable Telnet6 access to the switch from remote IPv6 nodes.
- Initiate an outbound telnet session to another IPv6 networked device.

Refer to “Telnet6 Operation” on page 5-6

### IP Preserve

IP Preserve operation preserves both the IPv4 and IPv6 addresses configured on VLAN 1 (the default VLAN) when a configuration file is downloaded to the switch using TFTP. Refer to “IP Preserve for IPv6” on page 5-23.

### Multicast Listener Discovery (MLD)

MLD operates in a manner similar to IGMP in IPv4 networks. In the factory default state (MLD disabled), the switch floods all IPv6 multicast traffic it receives on a given VLAN through all ports on that VLAN except the port receiving the inbound multicast traffic. Enabling MLD imposes management controls on IPv6 multicast traffic to reduce unnecessary bandwidth usage. MLD is configured per- VLAN. For information on MLD, refer to the chapter titled “Multicast Listener Discovery (MLD) Snooping”.

### Web Browser Interface

For the web browser interface, software release K.13.01 adds the following IPv6 functionality:

- configure and display IPv6 addressing
- ping6 diagnostic operation

# Configurable IPv6 Security

This section outlines the configurable IPv6 security features supported in software release K.13.01. For further information on these features, refer to the indicated pages.

## SSHv2 on IPv6

SSHv2 provides for the authentication between clients and servers, and protection of data integrity, and privacy. It is used most often to provide a secure alternative to Telnet and is also used for secure file transfers (SFTP and SCP). Software release K.13.01 with SSHv2 on IPv6 extends to IPv6 devices the SSH functionality that has been previously available on ProCurve switches running IPv4. This means that SSH version 2 connections are

supported between the switch and IPv6 management stations when SSH on the switch is also configured for IPv6 operation. The switch now offers these SSHv2 connection types:

- IPv6 only
- IPv4 only
- IPv4 or IPv6

The switch supports up to six inbound sessions of the following types in any combination at any given time:

- SSHv2
- SSHv2 IPv6
- Telnet-server
- Telnet6-server
- SFTP/SCP
- Console (serial RS-232 connection)

For more information, refer to “Secure Shell for IPv6” on page 6-15.

## IP Authorized Managers

The IPv6 Authorized IP Managers feature, like the IPv4 version, uses IP addresses and masks to determine which stations (PCs and workstations) can access the switch through the network, and includes these access methods:

- Telnet, SSH, and other terminal emulation applications
- the switch's web browser interface
- SNMP (with a correct community name)

Also, when configured in the switch, the access control imposed by the Authorized IP Manager feature takes precedence over the other forms of access control configurable on the switch, such as local passwords, RADIUS, and both Port-Based and Client-Based Access Control (802.1X). This means that the IP address of a networked management device must be authorized before the switch will attempt to authenticate the device by invoking any other access security features. Thus, with Authorized IP Managers configured, having the correct passwords or MAC address is not sufficient for accessing the switch through the network unless an IPv6 address configured on the station attempting the access is also included in the switch's Authorized IP Managers configuration. This presents the opportunity to combine the Authorized IP Managers feature with other access control features to enhance the security fabric protecting the switch.

**Caution**

The Authorized IP Managers feature does not protect against unauthorized station access through a modem or direct connection to the Console (RS-232) port. Also, if an unauthorized station "spoofs" an authorized IP address, then the unauthorized station cannot be blocked by the Authorized IP Managers feature, even if a duplicate IP address condition exists.

To configure authorized IPv6 managers, refer to "Authorized IP Managers for IPv6" on page 6-3.

For related information, refer to:

- RFC 4864, "Local Network Protection for IPv6".

# Diagnostic and Troubleshooting

Software release K.13.01 includes the IPv6 diagnostic and troubleshooting features listed in this section.

## ICMP Rate-Limiting

Controlling the frequency of ICMPv6 error messages can help to prevent DoS (Denial- of- Service) attacks. With IPv6 enabled on the switch, you can control the allowable frequency of these messages with ICMPv6 rate-limiting. Refer to "ICMP Rate-Limiting" on page 8-2.

## Ping6

Implements the Ping protocol for IPv6 destinations, and includes the same options as are available for IPv4 Ping, including DNS hostnames. Refer to "Ping for IPv6 (Ping6)" on page 8-4.

## Traceroute6

Implements Traceroute for IPv6 destinations, and includes the same same options as are available for the IPv4 Traceroute, including DNS hostnames. Refer to "Traceroute for IPv6" on page 8-6.

## Domain Name System (DNS) Resolution

This feature enables resolving a host name to an IPv6 address and the reverse, and takes on added importance over its IPv4 counterpart due to the extended length of IPv6 addresses. With DNS-compatible commands, CLI command entry becomes easier for reaching a device whose IPv6 address is configured with a host name counterpart on a DNS server.

Software release K.13.01 includes the following DNS-compatible commands:

- **ping6**
- **traceroute6**

The switches covered by this guide now support a prioritized list of up to three DNS server addresses. (Earlier software releases supported only one DNS server address.) Also, the server address list can include both IPv4 and IPv6 DNS server addresses. (An IPv6 DNS server can respond to IPv4 queries, and the reverse.)

**Note**

If an IPv6 DNS server address is configured on the switch, at least one VLAN on the switch (and in the path to the DNS server) must be configured with an IPv6 address.

For information on configuring DNS resolution on the switch, refer to "DNS Resolver for IPv6" on page 8-9.

## IPv6 Neighbor Discovery (ND) Controls

The neighbor discovery feature includes commands for:

- increasing or decreasing the frequency of Duplicate Address Detection searches
- displaying the IPv6 neighbor cache
- clearing dynamic entries from the neighbor cache

Refer to "Neighbor Discovery (ND) in IPv6" on page 2-9.

## Event Log

Messages returning IP addresses now include IPv6 addresses where applicable.

### SNMP

When IPv6 is enabled on a VLAN interface, you can manage the switch from a network management station configured with an IPv6 address. Refer to "SNMP Management for IPv6" on page 5-20.

### Loopback Address

Like the IPv4 loopback address, the IPv6 loopback address (::1) can be used by the switch to send an IPv6 packet to itself. However, the IPv6 loopback address is implicit on a VLAN and cannot be statically configured on any VLAN. Refer to "Loopback Address" on page 3-24.

### Debug/Syslog Enhancements

Includes new options for IPv6. Refer to "Debug/Syslog for IPv6" on page 8-12.

## IPv6 Scalability

As of software release K.13.01, the switches covered by this guide support the following:

- Dual stack operation (IPv4 and IPv6 addresses on the same VLAN).
- Maximum of 512 VLANs with IPv4 and IPv6 addresses in any combination.
- Up to 2048 VLANs configured on the switch.
- Maximum of 2048 active IPv6 addresses on the switch, in addition to a maximum of 2048 IPv4 addresses. ("Active IPv6 addresses" includes the total of all preferred and non-preferred addresses configured statically, through DHCPv6, and through stateless autoconfiguration. Excluded from "Active IPv6 Addresses" is the link-local address assigned to each VLAN, and "on- link" prefixes received as part of a router advertisement.)
- Maximum of 32 IPv6 addresses on a VLAN.
- Maximum of 10,000 IPv6 routes.

For more information on VLAN and route scalability on the switches covered by this guide, refer to the appendix titled "Scalability: IP Address, VLAN, and Routing Maximum Values" in the *Management and Configuration Guide* for your switch.

# Path MTU (PMTU) Discovery

IPv6 PMTU operation is managed automatically by the IPv6 nodes between the source and destination of a transmission. For Ethernet frames, the default MTU is 1500 bytes. If a router on the path cannot forward the default MTU size, it sends an ICMPv6 message (PKT_TOO_BIG) with the recommended MTU to the sender of the frame. If the sender of the frame is an IPv6 node that supports PMTU discovery, it will then use the MTU specified by the router and cache it for future reference.

For related information, refer to:

- RFC 1981: “Path MTU Discovery for IP version 6”

# 3

# IPv6 Addressing

## Contents

# Introduction

IPv6 supports multiple addresses on an interface, and uses them in a manner comparable to subnetting an IPv4 VLAN. For example, where the switch is configured with multiple VLANs and each is connected to an IPv6 router, each VLAN will have a single link-local address and one or more global unicast addresses. This section describes IPv6 addressing and outlines the options for configuring IPv6 addressing on the switch. The configuration process includes automatically or statically creating an IPv6 address and automatically verifying the uniqueness of each.

# IPv6 Address Structure and Format

## Address Format

An IPv6 address is composed of 128 bits divided into eight 2-byte fields of hexadecimal values. The full format is:

XXXX : XXXX : XXXX : XXXX : XXXX : XXXX : XXXX : XXXX

where each field delimited by a colon (:) is a set of four hexadecimal digits.

For example:

2001:0db8:0000:00A9:0215:60ff:fe7a:adc0

2001:0db8:0260:0212:0000:0000:0000:01b4

The hexadecimal characters in IPv6 addresses are not case-sensitive.

## Address Notation

Leading zeros in each field can be omitted as long as each field is represented by at least one value. The exception to this rule is when there is an uninterrupted series of zeros in one or more contiguous fields. In this case, the series of zeros can be replaced by "::", with the restriction that "::" can be used only once in a given address. Applying this convention to the above examples results in the following address notations:

2001:db8::a9:215:60ff:fe7a:adc0

2001:db8:260:0212::01b4

An IPv6 address includes a network prefix and an interface identifier.

## Network Prefix

The network prefix (high-order bits) in an IPv6 address begins with a well-known, fixed prefix for defining the address type. Some examples of well-known, fixed prefixes are:

2000::/3global (routable) unicast address

fd08::/8 unique local unicast address

fe80::/8link-local unicast address

ff00::/8multicast address

The remainder of the network prefix depends on the prefix type, and includes information such as the subnet destination of unicast addresses or the flags and scope of multicast addresses.

In a given address, CIDR-type notation (Classless Inter-Domain Routing) is used to define the network prefix. In the following address example, the 64 bits comprising 2001:0db8:0260:0201 form the network prefix:

2001:0db8:0260:0212:0215:60ff:fe7a:adc0/64

A shorter way to show this address is to remove the leading zeros:

2001:db8:260:212:215:60ff:fe7a:adc0/64

## Interface (Device) Identifier

The remaining (low-order) bits in the address comprise a unique interface identifier in an IPv6 address. In the above example, the rightmost 64 bits (215:60ff:fe7a:adc0) comprise the interface identifier. Unlike IPv4, an IPv6 identifier for a unicast or anycast address can be automatically generated from the switch's MAC address using EUI-64 (Extended Unique Identifier) format. Other methods include DHCPv6 assignments and static configuration. Interface identifiers are covered in more detail in the later sections of this chapter describing different address types.

# IPv6 Addressing Options

## IPv6 Address Sources

IPv6 addressing sources provide a flexible methodology for assigning addresses to VLAN interfaces on the switch. Options include:

- stateless IPv6 autoconfiguration on VLAN interfaces includes:
  - link-local unicast addresses
  - global unicast addresses
- stateful, global unicast IPv6 address configuration using DHCPv6
- static IPv6 address configuration

You can combine stateless, stateful, and static IP addressing methods on the switch as needed, according to the needs in your network. For example, if your network includes only one VLAN, you may need only stateless autoconfiguration of link-local addresses, although you could also use the static IPv6 method. (DHCPv6 does not configure link-local addresses.) Where routed traffic is used, you will also need global unicast addressing, either through stateless autoconfiguration or the other listed methods.

## General IPv6 Address Types

IPv6 supports stateless and stateful address autoconfiguration, as well as static address configuration.This enables IPv6 to automatically address a device so that it can be placed in a network with or without static or DHCPv6 addressing intervention. All three of these methods can be used exclusively or in conjunction with each other, and a given IPv6 device can have multiple addresses assigned to the same interface in a manner similar to subnetting in IPv4.

**Stateless Address Autoconfiguration .** This method does not require the use of servers. Instead, in the default operation, the host uses its MAC address to automatically generate a link-local IPv6 address using the EUI-64 method to generate the device identifier. (Refer to “Autoconfiguring Link-Local Unicast Addresses” on page 3-13.) The scope of the link-local address enables communication with other IPv6 devices on the same VLAN. If an IPv6 router is present, an IPv6 address supporting routing is automatically generated, as well. (The switch merges a router-generated prefix received in router advertisements with the last 64 bits of thelink-local address on an interface to create the global address.) Refer to page 3-7.

**Stateful Address Autoconfiguration.** This method allows use of a DHCPv6 server to automatically configure IPv6 addressing on a host in a manner similar to stateful IP addressing with a DHCPv4 server. For software release K.13.01, a DHCPv6 server can provide routable IPv6 addressing and NTP (timep) server addresses. Also, if the host acquires its IPv6 addressing through stateless or static methods, the DHCPv6 server can still be used to automatically provide other configuration information to the host. Refer to page 3-8.

**Static Address Configuration.** Static configuration is used instead of or in addition to stateless and stateful autoconfiguration where use of the host MAC address does not provide the desired level of address control and distribution. Refer to page 3-9.

**Duplicate Address Detection (DAD).** IPv6 verifies both the link-local and the global unicast address(es) on each interface for uniqueness, regardless of the method used to configure the address. If an address fails this test, it is identified as a **duplicate**, and a replacement must be configured using the static method. (To view address status, use the **show ipv6** command.) For more information on DAD, refer to "Neighbor Discovery (ND)" on page 4-17.

**Developing an Addressing Plan.** For small, flat networks and any environment where control of address assignments need not be restricted or tightly controlled, stateless addressing is adequate for network management and control. Where systematic and controlled addressing is needed, stateful and static addressing methods should be used. Where dual-stack operation is used in a VLAN, incorporating the local IPv4 addressing scheme into the IPv6 addresses you use can help to provide consistency and correspondence among the IPv6 and IPv4 addresses in use on the VLAN.

**Related Information.**

- RFC 4291: "IP Version 6 Addressing Architecture"
- RFC 2462: "IPv6 Stateless Address Autoconfiguration"
- RFC 3315: "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)"

# IPv6 Address Sources

IPv6 addressing sources provide a flexible methodology for assigning addresses to VLAN interfaces on the switch. Options include:

- stateless IPv6 autoconfiguration on VLAN interfaces includes:
  - link-local unicast addresses
  - global unicast addresses
- stateful IPv6 address configuration using DHCPv6
- static IPv6 address configuration

You can combine stateless, stateful, and static IP addressing methods on the switch as needed, according to the needs in your network. For example, if your network includes only one VLAN, you may need only stateless autoconfiguration of link-local addresses, although you could also use the static IPv6 method. (DHCPv6 does not configure link-local addresses.) Where routed traffic is used, you will also need global unicast addressing, either through stateless autoconfiguration or the other listed methods.

## Stateless Address Autoconfiguration (SLAAC)

On the switches covered by this guide, stateless address autoconfiguration (SLAAC) generates link-local unicast and global unicast IPv6 addresses on a VLAN interface. In all cases, the prefix is 64 bits.

### Applications

Stateless autoconfiguration is suitable where a link-local or global unicast IPv6 address (if a router is present) must be unique, but the actual address used is not significant. Where a specific unicast address or a unicast address from a specific range of choices is needed on an interface, DHCPv6 or static IPv6 address configuration should be used. (Refer to pages 3-8 and 3-9.)

### Preferred and Valid Lifetimes of Stateless Autoconfigured Addresses

The preferred and valid lifetimes of an autoconfigured global unicast address are set by the router advertisements (RA) used to generate the address, and are the autoconfiguration counterpart to the lease time assigned by DHCPv6

servers. These lifetimes cannot be reset using control from the switch console or SNMP methods. Refer to "Preferred and Valid Address Lifetimes" on page 3-25.

## Stateful (DHCPv6) Address Configuration

Stateful addresses are defined by a system administrator or other authority, and automatically assigned to the switch and other devices through the Dynamic Host Configuration Protocol (DHCPv6). Generally, DHCPv6 should be applied when you want specific, non-default addressing to be assigned automatically. For IPv6, DHCP use is indicated for conditions such as the following:

- address conventions used in your network require defined control
- static addressing is not feasible due to the number of nodes in the network
- automatic assignment of multiple IPv6 addresses per interfaces is needed
- automatic configuration of IPv6 access to DNS, SNTP, or TimeP servers

To implement stateful address configuration:

- The DHCPv6 server must be configured and accessible to the switch, either on the same VLAN or through an IPv6 router configured with DHCP Relay to support service requests from the switch.

**Note**

DHCPv6 relay may not currently be available in some IPv6 routers.

- DHCPv6 addressing must be enabled per-VLAN on the switch.

Note that IPv6 router advertisements (RAs) can also include instructions to clients to use DHCPv6 resources. Refer to the documentation for your IPv6 router.

If you want to use DHCPv6 in a dual-stack environment, you will need both DHCPv4 and DHCPv6 server access. Also, further developments in DHCP services are likely to mean new capabilities affecting DHCPv6 deployments.

For related information, refer to:

- RFC 3315: "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)"
- RFC 3041: "Privacy Extensions for Stateless Address Autoconfiguration in IPv6"

## Static Address Configuration

Generally, static address configuration should be used when you want specific, non-default addressing to be assigned to a VLAN interface. For IPv6, DHCP use is indicated for conditions such as the following:

- address conventions used in your network require defined control
- the task of static addressing is not so extensive as to be impractical due to the number of addresses and/or interfaces needing configuration

If IPv6 is not already enabled on a VLAN interface, the following is true:

- Statically configuring a link-local address on the interface also enables IPv6.
- Statically configuring a global unicast or anycast address also enables IPv6 and generates a link-local address.

Statically configured global unicast addresses can be used in addition to stateless addresses on the same interface. However, because only one link-local address is allowed on a VLAN interface (fe80::), static configuration of a link-local address automatically replaces an existing link-local address.

**Note**

For a statically configured global unicast address to be routable, a gateway router must be transmitting router advertisements on the VLAN that include the prefix used in the statically configured address. If the VLAN is not receiving an RA with this prefix, the address is listed as “preferred”, but is not used.

Statically configured IPv6 addresses saved to the startup-config file (by using **write memory**) remain across a reboot and are permanent, unless statically removed by **no ipv6 address < *ipv6-addr* >**.

For more information and the CLI command for static address configuration, refer to “Configuring a Static IPv6 Address on a VLAN” on page 4-11.

# Address Types and Scope

## Address Types

IPv6 uses these IP address types:

- **Unicast:** Identifies a specific IPv6 interface. Traffic having a unicast destination address is intended for a single interface. Like IPv4 addresses, unicast addresses can be assigned to a specific VLAN on the switch and to other IPv6 devices connected to the switch. At a minimum, a given interface must have at least a link-local address. To send or receive traffic off of a VLAN, an interface must also have one or more global unicast addresses.
- **Multicast:** Provides a single destination address for traffic intended for all members of a group, and provides a means for reducing unnecessary traffic to interfaces that do not belong to a given multicast group. Membership in a group can be determined by request or by a characteristic, such as all nodes, all routers, or all routers of a given type. Multicast traffic can be generated by a single source or multiple sources, but in either case is intended for multiple destinations.Common types of multicast traffic include streaming video and audio to multiple receivers who have joined a specific group from diverse locations.

**Note**

Unlike IPv4, broadcast addresses are not used in IPv6. Multicast addresses are used instead. For more on this topic, refer to “Multicast Application to IPv6 Addressing” on page 3-21.

- **Anycast:** A single address of this type can be assigned to multiple interfaces, possibly on separate devices within a defined address scope, where any of the interfaces having the anycast address can provide the desired service or response. A packet sent to a given anycast address is delivered only to the nearest interface having an instance of the address. This option is useful where multiple servers provide the same service, and it does not matter to the client which source it uses to acquire the service. Anycast usage can be of value, for example, in a network supporting multiple DNS servers. Refer to “Anycast Addresses” on page 3-20.

A given interface can have only one link-local address, but can have multiple unicast and anycast addresses.

## Address Scope

The address scope determines the area (topology) in which a given IPv6 address is used. This section provides an overview of IPv6 address types. For more information, refer to the chapter titled “IPv6 Addressing”.

**Link-Local Address.** Limited to a given interface (VLAN). Enabling IPv6 on a given VLAN automatically generates a link-local address used for switched traffic on the VLAN.

**Global Unicast Address.** Applies to a unique IPv6 routable address on the internet. A unique global address has a routing prefix and a unique device identifier.When autoconfiguration is enabled on a VLAN receiving an IPv6 router advertisement (RA), the prefix specified in the RA and the device identifier specified in the link-local address are combined to create a unique, global unicast address. A global unicast address can also be statically configured to either replace or complement an automatically configured address of the same type.

**Unique Local Unicast.** Applies to a routable, globally unique address intended for use within an entity defined by the system adminstrator, such as a specific site or a group of related sites defined by IPv6 border routers. These addresses are intended to be routable on a local site or an organization's intranet, but are not intended to be routed on the global internet. A unique local unicast address has the same format as a global unicast address. In this guide, unless otherwise stated, information on global unicast addresses also applies to unique local unicast addresses. For more on this topic, refer to “Unique Local Unicast IPv6 Address” on page 3-19.

## Unicast Address Prefixes

Traffic having a unicast destination address is intended for a single interface identified by that address. While IPv6 unicast addresses can have prefixes of varying length, a 64-bit prefix is generally adequate.

**Link-Local Unicast Prefix (fe80):** This well-known 64-bit fixed prefix is for a non- routable address used to identify a device on a single VLAN interface, and requires the high-order ten bits to be set to fe80 (fe80::/10). The remaining 54 bits in the prefix are set to zeros, followed by an interface ID of 64 bits.

fe80:0000:0000:0000:0215:60ff:fe7a:adc0/64

or

fe80::215:60ff:fe7a:asc0/64

In binary notation, the fixed prefix for link-local prefixes is:

1111 1110 10 = fe80/10

For more on link-local addresses, refer to “Link-Local Unicast Address” on page 3-13.

**Routable Global Unicast Prefix.** This well-known 3-bit fixed-prefix indicates a routable address used to identify a device on a VLAN interface that is accessible by routing from multiple networks. The complete prefix is 64 bits, followed by a 64-bit interface identifier. For example, the leading 2 in the first octet of the following address illustrates a global unicast address:

2001:db8:260:212:215:60ff:fe7a:adc0/64

In binary notation, the fixed prefix in this example appears as follows:

0010 0000 = 20/3

**Unique Local Unicast Prefix (fd).** This well-known fixed prefix is defined as FC00/7. However, the eighth high-order bit must also be set to 1, resulting in a fixed prefix of fd00/8. (In the future, setting the eighth high-order bit to zero may become an option.) This prefix signifies a routable address intended for use within the boundaries of a site or organization. For example, the leading fd in the first octet of this address illustrates a unique local unicast address intended to be used in a privately defined network.

fd00:00ff:0C00:000a:215:60ff:fe7a:adc0

Unique local unicast addresses are described in more detail under "Unique Local Unicast IPv6 Address" on page 3-19.

**Multicast Prefix (ff).** This well-known 8-bit fixed prefix signifies a permanent or temporary multicast address. The second 8 high-order bits are used for flags and scope for the multicast address. The remaining 112 bits define the multicast group identifier. For example:

ff02::1:ffc7:b5b9

For more information, refer to “Multicast Application to IPv6 Addressing” on page 3-21.

**Other Prefix Types.** There are other designated global unicast prefixes such as those for the following address types:

- RFC 4380: “Teredo: Tunneling IPv6 over UDP”
- RFC 3056: “Connection of IPv6 Domains via IPv4 Clouds”
- RFC 4214: “Intra-Site Automatic Tunnel Addressing Protocol (ISATAP)”

For related information, refer also to:

- RFC 4291: "IP Version 6 Addressing Architecture

# Link-Local Unicast Address

A link-local unicast address is a non-routable address for use on a single VLAN interface, and provides basic connectivity to an IPv6 network. Because the scope of a link-local address is restricted to the VLAN on which the address is used, a link-local address must be unique only for the VLAN on which it is configured. (Traffic with a link-local source or destination address cannot be routed between VLANs.)

## Autoconfiguring Link-Local Unicast Addresses

Enabling IPv6 on a given VLAN automatically generates a link-local address. This address is limited in scope to that VLAN, and is usable only for switched traffic. This address has a well- known, 64-bit prefix of fe80:0000:0000:0000 (hexadecimal), or fe80::, and a 64-bit device identifier derived from the VLAN's MAC address using the Extended Unique Identifier format (EUI-64, page 3-14). For example, if the MAC address of VLAN 10 is 021560-7aadc0, the automatically generated link-local address for VLAN 10 is:

fe80:0000:0000:0000:0215:60ff:fe7a:adc0

or, in standard IPv6 notation,

fe80::215:60ff:fe7a:adc0

Note that only one link-local address is allowed on an interface. Thus, on a given interface, statically configuring a link-local address type replaces the existing link-local address.

Because all VLANs configured on the switch use the same MAC address, all automatically generated link-local addresses on the switch will have the same link-local address. However, since the scope of a link-local address includes only the VLAN on which it was generated, this should not be a problem.

For example, executing **ipv6 address dhcp full** on a VLAN for which IPv6 was not previously configured does all of the following:

- enables IPv6 on the VLAN
- causes the switch to generate a stateless link-local unicast address on the VLAN
- configures the VLAN to send DHCPv6 requests

**Note**

Only one link-local unicast address can exist on a VLAN interface at any time. Configuring a new address of this type on an interface on which IPv6 is already enabled replaces the previously existing link-local address with the new one.

Any link-local address must include the well-known link-local prefix fe80::/64 plus a 64-bit device identifier.

Any of the following commands enable IPv6 on a VLAN and automatically generate a link-local address:

- **ipv6 enable** (page 4-6)
- **ipv6 address autoconfig** (page 4-7)
- **ipv6 address dhcp full [rapid-commit]** (page 4-9)
- **ipv6 address < *network-prefix*><*device-id* >/< *prefix-length* >** (page 4-13)

## Extended Unique Identifier (EUI)

When the link-local address is automatically generated, the device identifier is derived from the switch's 48- bit (hexadecimal) MAC address to create a 64-bit Extended Unique Identifier (EUI) to be appended to the fe80 link-local prefix, as follows:

- ff-fe is inserted between third and fourth bytes of MAC address
- The second low-order bit (the Universal/Local bit) in the first byte of the MAC address is complemented, which usually means the bit is originally set to 0 and is changed to 1. This indicates a globally unique IPv6 interface identifier. For example:

| MAC Address | IPv6 I/F Identifier | Full Link-Local Unicast Address |
|---|---|---|
| 00-15-60-7a-ad-c0 | 215:60ff:fe7a:adc0 | fe80::215:60ff:fe7a:adc0/64 |
| 09-c1-8a-44-b4-9d | 11c1:8aff:fe44:b49d | fe80::11c1:8aff:fe44:b49d/64 |
| 00-1a-73-5a-7e-57 | 21a:73ff:fe5a:7e57 | fe80::21a:73ff:fe5a:7e57/64 |

The EUI method of generating a link-local address is automatically implemented on the switches covered by this guide when IPv6 is enabled on a VLAN interface.

If automatically generated link-local addresses are not suitable for the addressing scheme you want to use, statically assigned link-local addresses can be used instead. (Refer to “Static Address Configuration” on page 3-9.)

For related information, refer to:

- RFC 2373: “IP Version 6 Addressing Architecture”
- RFC 2464: “Transmission of IPv6 Packets Over Ethernet Networks”

**Note**

While only one link-local IPv6 address is allowed on an interface, multiples of other address types can exist on the same interface. Thus, an interface can have one link-local unicast address, but multiple global unicast, anycast, and unique local addresses.

## Statically Configuring Link-Local Addresses

A link-local unicast address can be configured statically on a VLAN interface. If IPv6 is not already enabled on the VLAN, this action also enables IPv6 on the VLAN. Only one link-local address can exist on a VLAN at any time. If a link-local address (static or autoconfigured) already exists on the VLAN, then statically configuring a new one replaces the previously existing one. To statically configure a link-local address, refer to “Statically Configuring a Link-Local Unicast Address ” on page 4-12.

# Global Unicast Address

A global unicast address is required for unicast traffic to be routed across VLANs within an organization as well as across the public internet. To support subnetting, a VLAN can be configured with multiple global unicast addresses. Any of the following methods can be used to configure this kind of address on a VLAN:

- stateless address autoconfiguration using a prefix received in an advertisement received from a router on the VLAN (page 3-7)
- stateful address configuration using DHCPv6 (page 3-8)
- static address configuration (page 3-9)

## Stateless Autoconfiguration of a Global Unicast Address

If there is an IPv6-enabled router transmitting router advertisements on a VLAN interface, enabling this method generates a global, routable unicast address for the VLAN. The prefix for this address type is typically 64 bits with the three highest-order bits set to 2.

**Router Advertisements.** With autoconfiguration enabled, if the switch receives the same prefix from router advertisements (RAs) from multiple IPv6 routers on the same VLAN, then one global unicast address is configured with that prefix. If different prefixes are received from different routers on the same VLAN, then there will be one address configured on the VLAN for each unique prefix received. Where there are multiple routers on the VLAN, the default route for the VLAN is determined by the relative router priorities included in the RAs the VLAN receives. If the highest priority is duplicated on multiple routers, then the first RA detected on the VLAN determines the default route.

If the RA used to define the prefix for an autoconfigured address ceases to be received on the VLAN, then the address becomes deprecated. (Refer to "IPv6 Address Deprecation" on page 3-25.)

If IPv6 is not already enabled on a VLAN when you enable autoconfiguration on the VLAN, then the switch automatically generates a link-local address for the VLAN as well.

**If IPv6 Is Not Already Enabled.** Enabling address autoconfiguration on a VLAN when IPv6 is not already enabled on the VLAN causes the switch to:

- generate a link-local address on the VLAN as described in the preceeding section (page 3-13).
- transmit a router solicitation on the VLAN, and to listen for advertisements from any IPv6 routers on the VLAN.

For each unique router advertisement (RA) the switch receives from any router(s), the switch configures a unique, global unicast address. This address type is composed of a 64-bit network prefix specified by the router advertisement, plus a device identifier generated in the same way as described in the preceeding section for link-local addresses (using the EUI algorithm). For example, suppose the following is true:

- IPv6 is not enabled on VLAN 1.
- The MAC address for VLAN 1 is 00-15-60-7a-ad-c0.
- A router on the same VLAN transmits router advertisements that assign the prefix 2001:0:260:212/64, plus a 64-bit interface identifier generated using the EUI format.

In this case, enabling IPv6 address autoconfiguration on VLAN 1 generates the following address assignments on VLAN 1:

- link-local unicast: fe80::215:60ff:fe7a:adc0/64
- global unicast:2001:0:260:212:215:60ff:fe7a:adc0/64

**IPv6 Already Enabled.** Enabling address autoconfiguration on a VLAN when IPv6 is already enabled on the VLAN creates a global unicast address in the same way as described above, except that the device identifier applied to the new global address is a duplicate of the 64-bit identifier in the current link-local address.

**Note**

After a global unicast address has been configured, its device identifier will not be changed by any later changes to the link-local address.

## Static Configuration of a Global Unicast Address

A global unicast address can be configured statically on a VLAN interface. If IPv6 is not already enabled on a VLAN, then statically configuring a global unicast address automatically generates a link-local unicast address on the VLAN, as described in the pdreceeding section. To statically configure a global unicast address, refer to “Statically Configuring A Global Unicast Address” on page 4-13.

## Prefixes in Routable IPv6 Addresses

In routable IPv6 addresses, the prefix uniquely identifies an entity and a unicast subnet within that entity, and is defined by a length value specifying the number of leftmost contiguous (high-order) bits comprising the prefix. For an automatically generated global unicast address, the default prefix length is 64 bits. (Pratically speaking, the entire prefix in a /64 address defines the subnet.) Prefixes configured through stateful or static methods can be any length compatible with the local network application.

In the following example, the leftmost 64 bits of the address comprise the prefix:

2001:0db8:0000:0212:0215:60ff:fe7a:adc0/64

or

2001:db8::212:215:60ff:fe7a:adc0/64

In this case, the prefix is read as:

2001:0db8:0000:0212::

or

2001:db8::212::

All bits to the right of 0212 comprise the device identifier in the unicast address.

For related information, refer to:

- RFC 3177: “IAB/IESG Recommendations on IPv6 Address Allocations to Sites”
- RFC 4291: “IP Version 6 Addressing Architecture”

# Unique Local Unicast IPv6 Address

A unique local unicast address is an address that falls within a specific range, but is used only as a global unicast address within an organization. Traffic having a source address within the defined range should not be allowed beyond the borders of the intended domain or onto the public internet.

The current prefix for specifically identifying unique local unicast addresses is fd00/8. The leftmost 64 bits of a unique local unicast address include:

- the well-known prefix “fd”
- a 40-bit global identifier
- a 16-bit subnet identifier

For example:

fd73:110:255:23:215:60ff:fe7a:adc0/64

In the above case, the following values are used with the well-known prefix and L-bit setting:

- global identifier: 0073:110:255
- subnet identifier: 23
- interface identifier: 215:60ff:fe7a:adc0

Unique local unicast addresses can be assigned by router advertisements, DHCPv6 servers, or static configuration. The boundaries for unique local unicast address are set by border routers. Unique local unicast addresses can be assigned in DNS servers supporting an internal network, but should not be included in global DNS assignments.

For related information, refer to:

- RFC 4193: “Unique Local IPv6 Unicast Addresses”

# Anycast Addresses

Network size, traffic loads and the potential for network changes make it desirable to build in redundancy for some network services to provide increased service reliability. Anycast addressing provides this capability for applications where it does not matter which source is actually used to provide a service that is offered on multiple sources. Some applications that can benefit from anycast addressing include:

- DNS (UDP)
- time servers
- multicast rendezvous
- syslog devices
- gateways to a common network area.

Similarly, it is also useful in some cases to economically provide redundant paths to a given entity, such as a specific service provider. With IPv6 this can be done efficiently using the anycast address capability to assign the same address to multiple devices providing access to the desired services. An added benefit of utilizing anycast addresses is to reduce the need to configure clients with the addresses of multiple devices offering the same service.

An anycast address is an identifier for a set of interfaces typically belonging to different nodes. Packets sent to an anycast address are delivered to one of the interfaces identified as the “nearest” address, according to the routing protocol's measure of distance.

**Note** Equal-Cost paths between a host and multiple instances of the same anycast address can result in different packets in the same communication session to be sent to different destinations, and should be avoided.

An anycast address is formatted the same as a unicast address. For this reason, configuring an anycast address on the switch includes using an **anycast** keyword as part of the command. The prefix for an anycast address should include all areas of the network in which the address is used. For information on configuring an anycast address on the switches covered by this guide, refer to “Statically Configuring An Anycast Address” on page 4-14.

**Note** Duplicate Address Detection (DAD) does not apply to anycast addresses.

For related information, refer to:

- RFC 4291: “IP Version 6 Addressing Archetecture”
- RFC 2526: “Reserved IPv6 Subnet Anycast Addresses”

# Multicast Application to IPv6 Addressing

Multicast is used to reduce traffic for applications that have more than one recipient for the same data. IPv6 also uses multicast for purposes such as providing a more defined control of administrative traffic on a VLAN interface than can be achieved with the broadcast method used by IPv4. This approach improves traffic control for such purposes as neighbor and router solicitations, router advertisements, and responses to DAD messages. It also avoids the bandwidth consumption used for broadcasts by narrowing the scope of possibly interested destinations for various types of messages.

## Overview of the Multicast Operation in IPv6

When IPv6 is enabled on a VLAN interface on the switch, the interface automatically joins the *All-Nodes* and *Solicited-Node* multicast address groups for each of its configured unicast and anycast addresses. The interface also attempts to learn of other devices by sending solicitations to additional, well-known multicast groups, such as the following:

- all routers
- all MLDv2-capable routers, if multicast listener discovery (MLD) is enabled on the interface
- all DHCP agents (if DHCP is enabled on the interface)

There is a separate, *solicited node multicast group* for each IPv6 unicast and anycast address configured on a given interface. These automatically generated groups are limited in scope to the VLANs on which the node resides. Where multiple IPv6 unicast or anycast addresses on the same node differ only in their prefixes, they join the same solicited-node multicast group. Solicited-Node multicast groups are used, for example, in autoconfiguration. In this case, a node attempting to autoconfigure a link-local address computes the solicited-node multicast address for the proposed link-local address, then sends a Neighbor solicitation to this solicited-node multicast address. If there is no response from another node, the proposed address is available for use.

For more on Neighbor Discovery, refer to “Neighbor Discovery (ND)” on page 4-17.

For information on Multicast Listener Discovery (MLD) refer to the chapter titled “Multicast Listener Discovery (MLD) Snooping”.

When MLD is enabled on an interface, you can use **show ipv6 mld [ vlan < *vid* >]** to list the active multicast group activity the switch has detected per interface from other devices.

## IPv6 Multicast Address Format

The multicast address format has three principal sections in the leading 16 bits:

- identifier: ff (bits 1-8)
- flags: 0xxx (bits 9-12)
- scope: 0001 - 1110 (bits 13-16)

For related information, refer to RFC 4291.

### Multicast Group Identification

| Multicast ID, Flags and Scope (16 bits) | Group Identifier (112 bits) |
|---|---|
| 1111 1111 0*xxx xxxx*: | *x...x* : *x...x* : *x...x* : *x...x* : *x...x* : *x...x* : *x...x* : *x...x* |

- **multicast identifier:** The first eight high-order bits, set to ff, identify the address as multicast.
- **multicast flags:** Bits 9-12 are multicast flags that provide additional information about the multicast address, as follows:

| Bit ID | Options | Use |
|---|---|---|
| 9 | 0 | reserved |
| 10 (R) | 0 | multicast address without PIM-SM rendezvous point |
| | 1 | multicast address with PIM-SM rendezvous point |
| 11 (P) | 0 | multicast address without prefix information from the originating network |
| | 1 | multicast address with prefix information from the originating network |
| 12 (T) | 0 | multicast address is permanent (well-known, and not restricted by scope value) |
| | 1 | multicast address is temporary (and used only within an identified scope) |

- **multicast scope:** Bits 13-16 set boundaries on multicast traffic distribution, such as the interface defined by the link-local unicast address of an area, or the network boundaries of an organization. Because IPv6 uses multicast technology in place of the broadcast technology used in IPv4, the multicast scope field also controls the boundaries for broadcast-type traffic sent in multicast packets.

| Bit | Use |
|---|---|
| 0 | reserved |
| 1 | interface-local (loopback) |
| 2 | link-local (same topology as the corresponding link-local unicast scope) |
| 3 | reserved |
| 4 | admin-local (smallest administratively configured scope) |
| 5 | site-local (single site) |
| 6 | *unassigned* |
| 7 | *unassigned* |
| 8 | organization-local (multiple sites within the same organization) |
| 9 | *unassigned* |
| A | *unassigned* |
| B | *unassigned* |
| C | *unassigned* |
| D | *unassigned* |
| E | global |
| F | reserved |

  For example, the following prefix indicates multicast traffic with a temporary multicast address and a link-local scope:

  ff12 or (binary) 1111 1111 0001 0010

- **group identifier:** This field includes the last 112 bits of the multicast address and contains the actual multicast group identity. (Refer to RFCs 3306, 4291, and 2375.)

## Solicited-Node Multicast Address Format

The solicited-node multicast address the switch generates for a configured unicast or anycast address is composed of a unique, 104-bit multicast prefix (ff02:0:0:0:0:1:ff) and the last 24 bits of the subject address. For example, if a VLAN interface is configured with a link-local address of

fe90::215:60ff:fe7a:adc0

then the corresponding solicited-node multicast address is

ff02:0:0:0:0:1:ff7a:adc0

For related information, refer to:

- RFC 2375: IPv6 Multicast Address Assignments
- RFC 3306: Unicast-Prefix-based IPv6 Multicast Addresses
- RFC 3956: Embedding the Rendezvous Point (RP) Address in an IPv6 Multicast Address
- RFC 3177: IAB/IESG Recommendations on IPv6 Address Allocations to Sites
- RFC 4007: IPv6 Scoped Address Architecture
- RFC 4291: IP Version 6 Addressing Architecture
- "Internet Protocol Version 6 Multicast Addresses" (at **www.iana.org**)
- RFC 2710: Multicast Listener Discovery (MLD) for IPv6
- RFC 3810: Multicast Listener Discovery Version 2 (MLDv2) for IPv6 (Updates RFC 2710.)

# Loopback Address

The IPv6 loopback address is a link-local unicast address that enables a device to send traffic to itself for self-testing purposes. The loopback address does not have a physical interface assignment. If an IPv6 packet destined for the loopback address is received on a switch interface, it must be dropped. The IPv6 loopback address is never used as the source IPv6 address for any packet that is sent out of a device, and the switch drops any traffic it receives with a loopback address destination. An example use case is:

```
ProCurve# ping6 ::1

0000:0000:0000:0000:0000:0000:0000:0001 is alive, time = 1 ms
```

# The Unspecified Address

The “unspecified” address is defined as 0.0.0.0.0.0.0.0 (::/128, or just ::). It can be used, for example, as a temporary source address in multicast traffic sent by an interface that has not yet acquired its own address. The unspecified address cannot be statically configured on the switch, or used as a destination address.

# IPv6 Address Deprecation

## Preferred and Valid Address Lifetimes

Autoconfigured IPv6 global unicast addresses acquire their valid and preferred lifetime assignments from router advertisements. A *valid* lifetime is the time period during which an address is allowed to remain available and usable on an interface. A *preferred* lifetime is the length of time an address is intended for full use on an interface, and must be less than or equal to the address's valid lifetime.

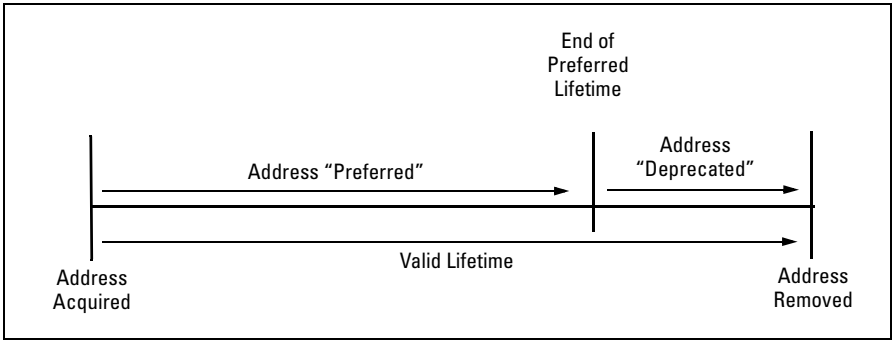



**Figure 3-1. Valid and Preferred Lifetimes**

When the preferred lifetime expires, the address becomes *deprecated*, meaning that the address should no longer be used as a source address (except for existing exchanges that began before the timeout occurred), but can still be used as a destination. When the timeout arrives for the valid lifetime, the address becomes unusable.

**Notes**

Preferred and valid lifetimes on a VLAN interface are determined by the router advertisements received on the interface. These values are not affected by the lease time assigned to an address by a DHCPv6 server. That is, lease expiration on a DHCPv6-assigned address terminates use of the address, regardless of the status of the RA-assigned lifetime, and router-assigned lifetime expiration of a leased address terminates the switch's use of the address. (The router-assigned lifetime can be extended by receipt of a new router advertisement.)

Statically configured IPv6 addresses are regarded as permanent addresses, and do not expire.

Related Information

- RFC 2462: "IPv6 Stateless Address Autoconfiguration"
- RFC 4291: "IP Version 6 Addressing Architecture"

# 4

# IPv6 Addressing Configuration

## Contents

# Introduction

| Feature | Default | CLI |
|---|---|---|
| Enable IPv6 with a Link-Local Address | disabled | 4-6 |
| Configure Global Unicast Autoconfig | disabled | 4-7 |
| Configure DHCPv6 Addressing | disabled | 4-9 |
| Configure a Static Link-Local Address | None | 4-12 |
| Configure a Static Global Unicast Address | None | 4-13 |
| Configure an Anycast Address | None | 4-14 |
| Change DAD Attempts | 3 | 4-18 |
| View Current IPv6 Addressing | *n/a* | 4-21 |

In the default configuration, IPv6 operation is disabled on the switch. This section describes the general steps and individual commands for enabling IPv6 operation.

This chapter provides the following:

- general steps for IPv6 configuration
- IPv6 command syntax descriptions, including **show** commands

Most IPv6 configuration commands are applied per-VLAN. The exceptions are ICMP, ND (neighbor discovery), and the (optional) authorized-managers feature, which are configured at the global configuration level. (ICMP and ND for IPv6 are enabled with default values when IPv6 is first enabled, and can either be left in their default settings or reconfigured, as needed.) For more informaton on ICMP, refer to “ICMP Rate-Limiting” on page 8-2. For more on ND, refer to “Neighbor Discovery (ND) in IPv6” on page 2-9.

For a quick reference to all IPv6 commands available on the switch, refer to the “IPv6 Command Index” on page xi at the front of this guide.

**Note**

Beginning with software release K.13.01, the switch is capable of operating in dual-stack mode, where IPv4 and IPv6 run concurrently on a given VLAN.

# General Configuration Steps

The IPv6 configuration on switches running software release K.13.01 includes global and per-VLAN settings. This section provides an overview of the general configuration steps for enabling IPv6 on a given VLAN and can be enabled by any one of several commands. The following steps provide a suggested progression for getting started.

**Note** The ICMP and Neighbor Discovery (ND) parameters are set to default values at the global configuration level are satisfactory for many applications and generally do not need adjustment when you are first configuring IPv6 on the switch.

In the default configuration, IPv6 is disabled on all VLANs.

1. If IPv6 DHCP service is available, enable IPv6 DHCP on the VLAN. If IPv6 is not already enabled on the VLAN, enabling DHCPv6 also enables IPv6 and automatically configures a link-local address using the EUI-64 format.

**Note** If IPv6 is not already enabled on the VLAN, enabling DHCPv6 causes the switch to automatically generate a link-local address. DHCPv6 does not assign a link-local address.

   A DHCPv6 server can provide other services, such as the addresses of time servers. For this reason you may want to enable DHCP even if you are using another method to configure IPv6 addressing on the VLAN.

2. If IPv6 DHCP service is not enabled on the VLAN, then do either of the following:
   - Enable IPv6 on the VLAN. This automatically configures a link-local address with an EUI- 64 interface identifier.
   - Statically configure a unicast IPv6 address on the VLAN. This enables IPv6 on the VLAN and, if you configure anything other than a link-local address, the link-local address will be automatically configured as well, with an EUI-64 interface identifier.
3. If an IPv6 router is connected on the VLAN, then enable IPv6 address autoconfiguration to automatically configure global unicast addresses with prefixes included in advertisements received from the router. The device identifier used in addresses configured by this method will be the same as the device identifier in the current link-local address.

4. If needed, statically configure IPv6 unicast addressing on the VLAN interface as needed. This can include any of the following:
   - statically replacing the automatically generated link-local address
   - statically adding global unicast, unique local unicast, and/or anycast addresses

## Configuring IPv6 Addressing

In the default configuration on a VLAN, any one of the following commands enables IPv6 and creates a link-local address. Thus, while any one of these methods is configured on a VLAN, IPv6 remains enabled and a link-local address is present:

ipv6 enable (page 4-6)

ipv6 address autoconfig (page 4-7)

ipv6 address dhcp full [rapid-commit] (page 4-9)

ipv6 address fe80:0:0:0:< *device-identifier* > link-local (page 4-12)

ipv6 address < *prefix:device-identifier* > (page 4-13)

**Note** Addresses created by any of these methods remain tentative until verified as unique by Duplicate Address Detection. (Refer to “Duplicate Address Detection (DAD)” on page 4-18.)

# Enabling IPv6 with an Automatically Configured Link-Local Address

This command enables automatical configuration of a link-local address .

***Syntax:*** [no] ipv6 enable

*If IPv6 has not already been enabled on a VLAN by another IPv6 command option described in this chapter, this command enables IPv6 on the VLAN and automatically configures the VLAN's link-local unicast address with a 64-bit EUI-64 interface identifier generated from the VLAN MAC address. (Refer to "Extended Unique Identifier (EUI)" on page 3-14.).*

***Note:*** *Only one link-local IPv6 address is allowed on the VLAN interface. Subsequent static or DHCP configuration of another link-local address overwrites the existing link-local address.*

*A link-local address always uses the prefix fe80:0:0:0.*

*With IPv6 enabled, the VLAN uses received router advertisements to designate the default IPv6 router. (Refer to "Default IPv6 Router" on page 4-28.)*

*After verification of uniqueness by DAD, a link-local IPv6 address assigned automatically is set to the* **preferred** *status, with a "permanent" lifetime. (Refer to "IPv6 Address Deprecation" on page 3-25.)*

***Default:*** *Disabled*

*The* **no** *form of the command disables IPv6 on the VLAN if no other IPv6-enabling command is configured on the VLAN. (Refer to "Disabling IPv6 on a VLAN" on page 4-16.)*

To view the current IPv6 Enable setting and any statically configured IPv6 addresses per-VLAN, use **show run**.

To view all currently configured IPv6 unicast addresses, use the following:

- **show ipv6** (Lists IPv6 addresses for all VLANs configured on the switch.)
- **show ipv6 vlan < *vid* >** (Lists IPv6 addresses configured on the VLAN.)

For more information, refer to "View the Current IPv6 Addressing Configuration" on page 4-21.

# Enabling Automatic Configuration of a Global Unicast Address and a Default Router Identity on a VLAN

Enabling autoconfig or rebooting the switch with autoconfig enabled on a VLAN causes the switch to configure IPv6 addressing on the VLAN using router advertisements and an EUI-64 interface identifier (page 3-14).

***Syntax:*** [no] ipv6 address autoconfig

*Implements unicast address autoconfiguration as follows:*

- *If IPv6 is not already enabled on the VLAN, this command enables IPv6 and generates a link-local (EUI- 64) address.*
- *Generates router solicitations (RS) on the VLAN.*
- *If a router advertisement (RA) is received on the VLAN, the switch uses the route prefix in the RA to configure a global unicast address. The device identifier for this address will be the same as the device identifier used in the current link-local address at the time the RA is received. (This can be either a statically configured or the (automatic) EUI-64 device identifier, depending on how the link-local address was configured.) For information on EUI- 64, refer to "Extended Unique Identifier (EUI)" on page 3-14.) If an RA is not received on the VLAN after autoconfig is enabled, a link-local address will be present, but no global unicast addresses will be autoconfigured.*

***Notes:*** *If a link-local address is already configured on the VLAN, a later, autoconfigured global unicast address uses the same device identifier as the link-local address.*

*Autoconfigured and DHCPv6-assigned global unicast addresses with the same prefix are mutually exclusive on a VLAN. On a given switch, if both options are configured on the same VLAN, then only the first to acquire a global unicast address will be used.*

*— Continued on the next page. —*

*— Continued from the previous page. —*

*After verification of uniqueness by DAD, an IPv6 address assigned to a VLAN by autoconfiguration is set to the preferred and valid lifetimes specified by the RA used to generate the address, and is configured as a preferred address. (Refer to "IPv6 Address Deprecation" on page 3-25.)*

***Default:*** *Disabled.*

*The* **no** *form of the command produces different results, depending on how IPv6 is configured on the VLAN:*

*If IPv6 was enabled only by the* **autoconfig** *command, then deleting this command disables IPv6 on the VLAN. (Refer to "Disabling IPv6 on a VLAN" on page 4-16.)*

To view the current IPv6 autoconfiguration settings per-VLAN, use **show run**.

To view all currently configured IPv6 unicast addresses, use the following:

- **show ipv6** (Lists IPv6 addresses for all VLANs configured on the switch.)
- **show ipv6 vlan < *vid* >** (Lists IPv6 addresses configured onthe VLAN.)

For more information, refer to "View the Current IPv6 Addressing Configuration" on page 4-21.

## Operating Notes

With IPv6 enabled, the VLAN uses received router advertisements to designate the default IPv6 router. (Refer to "Router Access and Default Router Selection" on page 4-27.)

# Enabling DHCPv6

Enabling the DHCPv6 option on a VLAN allows the switch to obtain a global unicast address and an NTP (network time protocol) server assignment for a Timep server. (If a DHCPv6 server is not needed to provide a global unicast address to a switch interface, the server can still be configured to provide the NTP server assignment. This is sometimes referred to as "stateless DHCPv6".)

***Syntax:*** [no] ipv6 address dhcp full [rapid-commit]

*This option configures DHCPv6 on a VLAN, which initiates transmission of DHCPv6 requests for service. If IPv6 is not already enabled on the VLAN by the* **ipv6 enable** *command, this option also enables IPv6 and causes the switch to autoconfigure a link-local unicast address with an EUI-64 interface identifier.*

> ***Notes:*** *A DHCPv6 server does not assign link-local addresses, and enabling DHCPv6 on a VLAN does not affect a pre-existing link-local address configured on the VLAN.*
>
> *A DHCPv6-assigned address can be configured on a VLAN when the following is true:*
>
> - *The assigned address is not on the same subnet as a previously configured autoconfig address.*
> - *The maximum IPv6 address limit on the VLAN or the switch has not been reached.*

*If a DHCPv6 server responds with an IPv6 address assignment, this address is assigned to the VLAN. (The DHCPv6-assigned address will be dropped if it has the same subnet as another address already assigned to the VLAN by an earlier autoconfig command.)*

*— Continued on the next page. —*

*— Continued from the previous page. —*

*After verification of uniqueness by DAD, an IPv6 address assigned to the VLAN by an DHCPv6 server is set to the preferred and valid lifetimes specified in a router advertisement received on the VLAN for the prefix used in the assigned address, and is configured as a preferred address. (Refer to the section titled "Address Lifetimes" on page 4-32.)*

**[rapid-commit]:** *Expedites DHCP configuration by using a two-message exchange with the server (solicit-reply) instead of the default four-message exchange (solicit-advertise- request-reply).*

**Default:** *Disabled*

*The* **no** *form of the command removes the DHCPv6 option from the configuration and, if no other IPv6-enabling command is configured on the VLAN, disables IPv6 on the VLAN. (Refer to "Disabling IPv6 on a VLAN" on page 4-16.)*

To view the current IPv6 DHCPv6 settings per-VLAN, use **show run**.

To view all currently configured IPv6 unicast addresses, use the following:

- **show ipv6** (Lists IPv6 addresses for all VLANs configured on the switch.)
- **show ipv6 vlan < *vid* >** (Lists IPv6 addresses configured on the VLAN.)

For more information, refer to "View the Current IPv6 Addressing Configuration" on page 4-21.

## Operating Notes

- If multiple DHCPv6 servers are available, the switch selects a server based on the preference value sent in DHCPv6 messages from the servers.
- The switch supports both DHCPv4 and DHCPv6 client operation on the same VLAN.
- DHCPv6 authentication and stateless DHCPv6 are not supported in software release K.13.01.
- With IPv6 enabled, the switch determines the default IPv6 router for the VLAN from the router advertisements it receives. (Refer to "Default IPv6 Router" on page 4-28.)

- DHCPv6 and statically configured global unicast or anycast addresses are mutually exclusive on a given VLAN. That is, configuring DHCPv6 on a VLAN erases any static global unicast or anycast addresses previously configured on that VLAN, and the reverse. (A statically configured link-local address will not be affected by configuring DHCPv6 on the VLAN.)
- For the same subnet on the switch, a DHCPv6 global unicast address assignment takes precedence over an autoconfigured address assignment, regardless of which address type was the first to be configured. If DHCPv6 is subsequently removed from the configuration, then an autoconfigured address assignment will replace it after the next router advertisement is received on the VLAN. DHCPv6 and autoconfigured addresses co-exist on the same VLAN if they belong to different subnets.

For related information refer to:

- RFC 3315: “Dynamic Host Configuration Protocol for IPv6 (DHCPv6)”
- RFC 3633: “IPv6 Prefix Options for Dynamic Host Configuration Protocol (DHCP) version 6”
- RFC 3736: “Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6”

# Configuring a Static IPv6 Address on a VLAN

This option enables configuring of unique, static unicast and anycast IPv6 addresses for global and link-local applications, including:

- link-local unicast (including EUI and non-EUI device identifiers)
- global unicast (and unique local unicast)
- anycast

## Statically Configuring a Link-Local Unicast Address

***Syntax:*** [no] ipv6 address fe80::< *device-identifier* > link-local

- *If IPv6 is not already enabled on the VLAN, this command enables IPv6 and configures a static link-local address.*
- *If IPv6 is already enabled on the VLAN, then this command overwrites the current, link- local address with the specified static address. (One link-local address is allowed per VLAN interface.)*

**< *device-identifier* >:** The low-order 64 bits, in 16-bit blocks, comprise this value in a link-local address:

xxxx xxxx : xxxx xxxx : xxxx xxxx : xxxx xxxx

*Where a static link-local address is already configured, a new, autoconfigured global unicast addresses assignment uses the same device identifier as the link-local address.*

> ***Notes:*** *An existing link-local address is replaced, and is not deprecated, when a static replacement is configured.*
>
> *The prefix for a statically configured link-local address is always 64 bits, with all blocks after fe80 set to zero. That is: fe80:0:0:0.*

*After verification of uniqueness by DAD, a statically configured link-local address status is set to* **preferred**, *with a* **permanent** *lifetime. (Refer to "IPv6 Address Deprecation" on page 3-25.)*

*For link-local addressing, the* ***no*** *form of the static IPv6 address command produces different results, depending on how IPv6 is configured on the VLAN:*

- *If IPv6 was enabled only by a statically configured link-local address, then deleting the link-local address disables IPv6 on the VLAN.*
- *If other IPv6-enabling commands have been configured on the VLAN, then deleting the statically configured link-local address causes the switch to replace it with the default (EUI-64) link-local address for the VLAN, and IPv6 remains enabled. (For more on the EUI-64 address format, refer to "Extended Unique Identifier (EUI)" on page 3-14.)*

*Refer also to "Disabling IPv6 on a VLAN" on page 4-16.*

## Statically Configuring A Global Unicast Address

***Syntax:.*** [no] ipv6 address < network-prefix><device-id >/< prefix-length >
[no] ipv6 address < network-prefix>::/< prefix-length > eui-64

*If IPv6 is not already enabled on a VLAN, either of these command options do the following:*

- *enable IPv6 on the VLAN*
- *configure a link-local address using the EUI-64 format*
- *statically configure a global unicast address*

*If IPv6 is already enabled on the VLAN, then the above commands statically configure a global unicast address, but have no effect on the current link-local address.*

**< *network-prefix* >:** *This includes the global routing prefix and the subnet ID for the address. For more on this topic, refer to "Prefixes in Routable IPv6 Addresses" on page 3-18.*

**< *device-id* >:** Enters a user-defined device identity.

**< *prefix-length* >:** Specifies the number of bits in the network prefix. If you are using the **eui-64** option, this value must be 64.

**eui-64:** *Specifies using the Extended Unique Identifier format to create a device identifier based on the VLAN MAC address. Refer to "Extended Unique Identifier (EUI)" on page 3-14.*

*After verification of uniqueness by DAD, the lifetime of a statically configured IPv6 address assigned to a VLAN is set to permanent, and is configured as a preferred address. (Refer to "IPv6 Address Deprecation" on page 3-25.)*

*The* **no** *form of the command erases the specified address and, if no other IPv6-enabling command is configured on the VLAN, disables IPv6 on the VLAN. (Refer to "Disabling IPv6 on a VLAN" on page 4-16.)*

To view the currently configured static IPv6 addresses per-VLAN, use **show run**.

To view all currently configured IPv6 unicast addresses, use the following:

- **show ipv6** (Lists IPv6 addresses for all VLANs configured on the switch.)
- **show ipv6 vlan < *vid* >** (Lists IPv6 addresses configured on **VLAN < *vid* >**.)

For more information, refer to "View the Current IPv6 Addressing Configuration" on page 4-21.

### Operating Notes

- With IPv6 enabled, the switch determines the default IPv6 router for the VLAN from the router advertisements it receives. (Refer to "Router Access and Default Router Selection" on page 4-27.)
- If DHCPv6 is configured on a VLAN, then configuring a static global unicast address on the VLAN removes DHCPv6 from the VLAN's configuration and deletes the DHCPv6-assigned global unicast address.
- Note that for a statically configured global unicast address to be routable, a gateway router must be transmitting router advertisements on the VLAN.
- If an autoconfigured global unicast address already exists for the same subnet as a new, statically configured global unicast address, the statically configured address is denied. In the reverse case, you can add an autoconfig command to the VLAN configuration, but it will not be implemented unless the static address is removed from the configuration.

## Statically Configuring An Anycast Address

Anycast addresses on the switch appear the same as global unicast addresses. To configure an anycast address on a VLAN, append the **anycast** keyword to the same command that is used to statically configure a global unicast address. (Link-Local unicast addresses cannot be configured as anycast addresses on the switch.)

Anycast addresses are allocated from the unicast address space, and cannot be distinguished from other IPv6 global unicast addresses configured on the switch, except by viewing the address configurations listed per-VLAN in the **show run** output. For more information on using anycast addresses, refer to "Anycast Addresses" on page 3-20.

***Syntax:.*** [no] ipv6 address < network-prefix >< device-identifier >/< prefix-length > anycast

*If IPv6 is not already enabled on a VLAN, this command option does the following:*

- *enables IPv6 on the VLAN*
- *configures a link-local address using the EUI-64 format*
- *statically configures an anycast address*

*If IPv6 is already enabled on the VLAN, then the above commandss statically configure an anycast address, but has no effect on the current link-local address.*

**anycast:** *Identifies the specified address as an anycast address. This allows the address to be duplicated (as an anycast address) on other devices on the same network.*

**Default:** *None.*

*The* **no** *form of the command erases the specified anycast address and, if no other IPv6- enabling command is configured on the VLAN, disables IPv6 on the VLAN. (Refer to "Disabling IPv6 on a VLAN" on page 4-16.)*

To verify the identity of anycast addresses configured for VLANs to which the switch belongs, use the **show run** command.

To view all currently configured IPv6 unicast addresses, use the following:

- **show ipv6** (Lists IPv6 addresses for all VLANs configured on the switch.)
- **show ipv6 vlan < *vid* >** (Lists IPv6 addresses configured on **VLAN < *vid* >**.)

For more information, refer to "View the Current IPv6 Addressing Configuration" on page 4-21.

### Duplicate Address Detection (DAD) for Statically Configured Addresses

Statically configured IPv6 addresses are designated as permanent. If DAD determines that a statically configured address duplicates a previously configured and reachable address on another device belonging to the VLAN, then the more recent, duplicate address is designated as **duplicate**. For more on this topic, refer to:

- "Duplicate Address Detection (DAD)" on page 4-18.
- "View the Current IPv6 Addressing Configuration" on page 4-21

**Note**

Multiple, duplicate addresses configured as Anycast on different devices are special cases of unicast addresses, and are not identified as duplicates by DAD. Refer to "Anycast Addresses" on page 3-20.

## Disabling IPv6 on a VLAN

While one IPv6-enabling command is configured on a VLAN, IPv6 remains enabled on that VLAN. In this case, removing the only IPv6-enabling command from the configuration disables IPv6 operation on the VLAN. That is, to disable IPv6 on a VLAN, all of the following commands must be removed from the VLAN's configuration:

```
ipv6 enable

ipv6 address dhcp full [rapid-commit]

ipv6 address autoconfig

ipv6 address fe80::< device-identifier > link-local

ipv6 address < prefix > : < device-identifier >
```

If any of the above remain enabled, then IPv6 remains enabled on the VLAN and, at a minimum, a link-local unicast address will be present.

# Neighbor Discovery (ND)

Neighbor Discovery (ND) is the IPv6 equivalent of the IPv4 ARP for layer 2 address resolution, and uses IPv6 ICMP messages to do the following:

- Determine the link-layer address of neighbors on the same VLAN interface.
- Verify that a neighbor is reachable.
- Track neighbor (local) routers.

Neighbor Discovery enables functions such as the following:

- router and neighbor solicitation and discovery
- detecting address changes for devices on a VLAN
- identifying a replacement for a router or router path that has become unavailable
- duplicate address detection (DAD)
- router advertisement processing
- neighbor reachability
- autoconfiguration of unicast addresses
- resolution of destination addresses
- changes to link-layer addresses
- anycast address operation

An instance of Neighbor Discovery is triggered on a device when a new (tentative) or changed IPv6 address is detected. (This includes stateless, stateful, and static address configuration.) ND operates in a per-VLAN scope; that is, within the VLAN on which the the device running the ND instance is a member. Neighbor discovery actually occurs when there is communication between devices on a VLAN. That is, a device needing to determine the link-layer address of another device on the VLAN initiates a (multicast) neighbor solicitation message (containing a solicited-node multicast address that corresponds to the IPv6 address of the destination device) on the VLAN. When the destination device receives the neighbor solicitation, it responds with a neighbor advertisement message identifying its link-layer address. When the initiating device receives this advertisement, the two devices are ready to exchange traffic on the VLAN interface. Also, when an IPv6 interface becomes operational, it transmits a router solicitation on the interface and listens for a router advertisement.

**Note:** Neighbor and router solicitations must originate on the same VLAN as the receiving device. To support this operation, IPv6 is designed to discard any incoming neighbor or router solicitation that does not have a value of 255 in the IP Hop Limit field. For a complete list of requirements, refer to RFC 246.

When a pair of IPv6 devices in a VLAN exchange communication, they enter each other's IPv6 and corresponding MAC addresses in their respective neighbor caches. These entries are maintained for a period of time after communication ceases, and then dropped.

To view or clear the content of the neighbor cache, refer to “Viewing and Clearing the IPv6 Neighbors Cache” on page 5-2.

For related information, refer to:

- RFC 2461: “Neighbor Discovery for IP Version 6 (IPv6)”

## Duplicate Address Detection (DAD)

Duplicate Address Detection verifies that a configured unicast IPv6 address is unique before it is assigned to a VLAN interface on the switch. DAD is enabled in the default IPv6 configuration, and can be reconfigured, disabled, or re-enabled at the global config command level. DAD can be useful in helping to troubleshoot erroneous replies to DAD requests, or where the neighbor cache contains a large number of invalid entries due to an unauthorized station sending false replies to the switch's neighbor discovery queries. If DAD verifies that a unicast IPv6 address is a duplicate, the address is not used. If the link-local address of the VLAN interface is found to be a duplicate of an address for another device on the interface, then the interface stops processing IPv6 traffic.

### DAD Operation

On a given VLAN interface, when a new unicast address is configured, the switch runs DAD for this address by sending a neighbor solicitation to the All-Nodes multicast address (ff02::1). This operation discovers other devices on the VLAN and verifies whether the proposed unicast address assignment is unique on the VLAN. (During this time, the address being checked for uniqueness is held in a tentative state, and cannot be used to receive traffic other than neighbor solicitations and neighbor advertisements.) A device that receives the neighbor solicitation responds with a Neighbor Advertisement

that includes its link-local address. If the newly configured address is from a static or DHCPv6 source and is found to be a duplicate, it is labelled as duplicate in the "Address Status" field of the **show ipv6** command, and is not used. If an autoconfigured address is found to be a duplicate, it is dropped and the following message appears in the Event Log:

```
W < date > < time > 00019 ip: ip address < IPv6-address >
removed from vlan id < vid >
```

DAD does not perform periodic checks of existing addresses. However, when a VLAN comes up with IPv6 unicast addresses configured (as can occur during a reboot) the switch runs DAD for each address on the interface by sending neighbor solicitations to the All-Nodes multicast address as described above.

If an address is configured while DAD is disabled, the address is assumed to be unique and is assigned to the interface. If you want to verify the uniqueness of an address configured while DAD was disabled, re-enable DAD and then either delete and reconfigure the address, or reboot the switch.

## Configuring DAD

***Syntax:*** ipv6 nd dad-attempts < 0 - 600 >

*This command is executed at the global config level, and configures the number of neighbor solicitations to send when performing duplicate address detection for a unicast address configured on a VLAN interface.*

**< 0 - 600 >:** *The number of consecutive neighbor solicitation messages sent for DAD inquiries on an interface. Setting this value to 0 disables DAD on the interface. Disabling DAD bypasses checks for uniqueness on newly configured addresses. If a reboot is performed while DAD is disabled, the duplicate address check is not performed on any IPv6 addresses configured on the switch.*

***Default:*** *3 (enabled); Range: 0 - 600 (0 = disabled)*

The **no** form of the command restores the default setting (3).

## Operating Notes

- A verified link-local unicast address must exist on a VLAN interface before the switch can run DAD on other addresses associated with the interface.
- If a previously configured unicast address is changed, a neighbor advertisement (an all-nodes multicast message–ff02::1) is sent to notify other devices on the VLAN and to perform duplicate address detection.
- IPv6 addresses on a VLAN interface are assigned to multicast address groups identified with well- known prefixes. For more on this topic, refer to “Multicast Application to IPv6 Addressing” on page 3-21.
- DAD is performed on all stateful, stateless, and statically configured unicast addresses, but not on Anycast addresses.
- Neighbor solicitations for DAD do not cause the neighbor cache of neighboring switches to be updated.
- If a previously configured unicast address is changed, a neighbor advertisement is sent on the VLAN to notify other devices, and also for duplicate address detection.
- If DAD is disabled when an address is configured, the address is assumed to be unique and is assigned to the interface.

# View the Current IPv6 Addressing Configuration

Use these commands to view the current status of the IPv6 configuration on the switch.

***Syntax:*** show ipv6

*Lists the current, global IPv6 settings and per-VLAN IPv6 addressing on the switch.*

**IPv6 Routing:** *For software release K.13.01, this setting is always* **Disabled**. *This is a global setting, and is not configured per-VLAN. (Refer to "Router Access and Default Router Selection" on page 4-27.)*

**Default Gateway:** *Lists the IPv4 default gateway, if any, configured on the switch. This is a globally configured router gateway address, and is not configured per-VLAN.*

**ND DAD:** *Indicates whether DAD is enabled (the default) or disabled. Using* **ipv6 nd dad-attempts 0** *disables neighbor discovery. (Refer to "Duplicate Address Detection (DAD)" on page 4-18.)*

**DAD Attempts:** *Indicates the number of neighbor solicitations the switch transmits per-address for duplicate (IPv6) address detection. Implemented when a new address is configured or when an interface with configured addresses comes up (such as after a reboot). The default setting is 3, and the range is 0 - 600. A setting of "0" disables duplicate address detection. (Refer to "Duplicate Address Detection (DAD)" on page 4-18.)*

**VLAN Name:** *Lists the name of a VLAN statically configured on the switch.*

**IPv6 Status:** *For the indicated VLAN, indicates whether IPv6 is disabled (the default) or enabled. (Refer to "Configuring IPv6 Addressing" on page 4-5.)*

**Address Origin:**

- **Autoconfig:** *The address was configured using stateless address autoconfiguration (SLAAC). In this case, the device identifier for global unicast addresses copied from the current link-local unicast address.*
- **DHCP:** *The address was assigned by a DHCPv6 server. Note that addresses having a DHCP origin are listed with a 128-bit prefix length.*
- **Manua:l:** *The address was statically configred on the VLAN.*
- **IPv6 Address/Prefix Length:** *Lists each IPv6 address and prefix length configured on the indicated VLAN.*

**Address Status:**

- **Tentative:** *DAD has not yet confirmed the address as unique, and is not usable for sending and receiving traffic.*
- **Preferred:** *The address has been confirmed as unique by DAD, and usable for sending and receiving traffic. The Expiry time shown for this address by the* **show ipv6 vlan < *vid* >** *command output is the preferred lifetime assigned to the address. (Refer to "Address Lifetimes" on page xxx.)*
- **Deprecated:** *The preferred lifetime for the address has been exceeded, but there is time remaining in the valid lifetime.*
- **Duplicate:** *Indicates a statically configured IPv6 address that is a duplicate of another IPv6 address that already exists on another device belonging to the same VLAN interface. A duplicate address is not used.*

For example, figure 4-1 shows the output on a switch having IPv6 enabled on one VLAN.

```
ProCurve(config)# show ipv6

 Internet (IPv6) Service

  IPv6 Routing    : Disabled
  Default Gateway : 10.0.9.80
  ND DAD          : Enabled
  DAD Attempts    : 3

  Vlan Name       : DEFAULT_VLAN
  IPv6 Status     : Disabled

  Vlan Name       : VLAN10
  IPv6 Status     : Enabled

  Address     |                                              Address
  Origin      | IPv6 Address/Prefix Length                   Status
  ----------- + -------------------------------------------- -----------
  autoconfig  | 2620:0:a03:e102::127/64                      preferred
  dhcp        | 2620:0:a03:e102:212:79ff:fe88:a100/64        preferred
  manual      | fe80::127/64                                 preferred
```

**Figure 4-1. Example of Show IPv6 Command Output**

***Syntax:*** show ipv6 vlan < *vid* >

*Displays IP and IPv6 global configuration settings, the IPv6 status for the specified VLAN, the IPv6 addresses (with prefix lengths) configured on the specified VLAN, and the expiration data (Expiry) for each address.:*

- **IPv6 Routing:** *For software release K.13.01, this setting is always* **Disabled**. *(Refer to "Router Access and Default Router Selection" on page 4-27.).*
- **Default Gateway:** *Lists the IPv4 default gateway, if any, configured on the switch. This is a globally configured router gateway address, and is not configured per-VLAN.*
- **ND DAD:** *Shows whether Neighbor Discovery (ND) is enabled. The default setting is Enabled. Using* **ipv6 nd dad-attempts 0** *disables neighbor discovery.*

- **DAD Attempts:** *Indicates the number of neighbor solicitations the switch transmits per-address for duplicate (IPv6) address detection. Implemented when a new address is configured or when an interface with configured addresses comes up (such as after a reboot). The default setting is 3, and the range is 0 - 600. A setting of "0" disables duplicate address detection. (Refer to "Duplicate Address Detection (DAD)" on page 4-18.)*
- **VLAN Name:** *Lists the name of a VLAN statically configured on the switch.*
- **IPv6 Status:** *For the indicated VLAN, indicates whether IPv6 is disabled (the default) or enabled. (Refer to "Configuring IPv6 Addressing" on page 4-5.)*
- **IPv6 Address/Prefix Length:** *Lists each IPv6 address and prefix length configured on the indicated VLAN.*
- **Expiry:** *Lists the lifetime status of each IPv6 address listed for a VLAN:*
  - **Permanent:** *The address will not time out and need renewal or replacement.*
  - **date/time:** *The date and time that the address expires. Expiration date and time is specified in the router advertisement used to create the prefix for automatically configured, global unicast addresses. The* **Address Status** *field in the* **show ipv6** *command output indicates whether this date/time is for the "preferred" or "valid" lifetime assigned to the corresponding address. (Refer to "Preferred and Valid Address Lifetimes" on page 3-25.)*

```
ProCurve(config)# show ipv6 vlan 10

 Internet (IPv6) Service

  IPv6 Routing    : Disabled
  Default Gateway : 10.0.9.80
  ND DAD          : Enabled
  DAD Attempts    : 3

  Vlan Name       : VLAN10
  IPv6 Status     : Enabled

  IPv6 Address/Prefixlength                   Expiry
  ------------------------------------------- -------------------------
  2620:0:a03:e102::127/64                     Wed Jan 23 14:16:17 2008

  2620:0:a03:e102:212:79ff:fe88:a100/64       Sat Jan  5 05:02:22 2008

  fe80::127/64                                permanent
```

**Figure 4-2. Example of Show IPv6 VLAN < *vid* > Output**

***Syntax:*** show run

*In addition to the other elements of the current configuration, this command lists the statically configured, global unicast and anycast IPv6 addressing, and the current IPv6 configuration per-VLAN. The listing may include one or more of the following, depending on what other IPv6 options are configured on the VLAN. Any stateless address autoconfiguration (SLAAC) commands in the configuration are also listed in the output, but the actual addresses resulting from these commands are not included in the output.*

- ipv6 enable
- ipv6 address fe80::< *device-id* > link-local
- ipv6 address < *prefix* >:< *device-id* >/< *prefix-length* >
- ipv6 address autoconfig
- ipv6 address dhcp full [rapid-commit]
- ipv6 < *global-unicast-address* >/< *prefix* > anycast

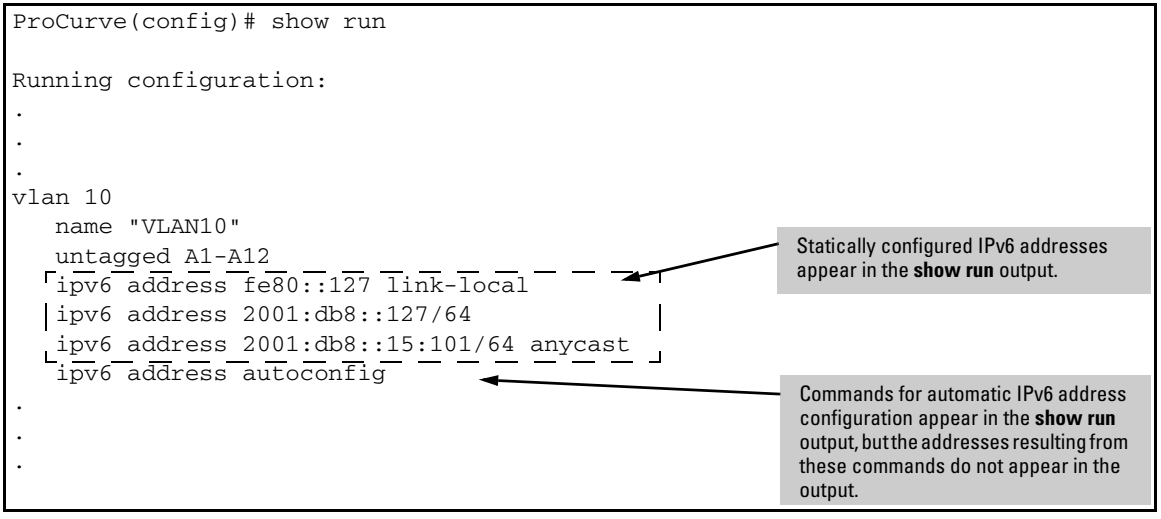



**Figure 4-3. Example of Show Run Output Listing the Current IPv6 Addressing Commands**

# Router Access and Default Router Selection

Routing traffic between destinations on different VLANs configured on the switch or to a destination on an off-switch VLAN is done by placing the switch on the same VLAN interface or subnet as an IPv6-capable router configured to route traffic to other IPv6 interfaces or to tunnel IPv6 traffic across an IPv4 network.

## Router Advertisements

An IPv6 router periodically transmits router advertisements (RAs) on the VLANs to which it belongs to notify other devices of its presence. The switch uses these advertisements for purposes such as:

- learning the MAC and link-local addresses of IPv6 routers on the VLAN (For devices other than routers, the switch must use neighbor discovery to learn these addresses.)
- building a list of default (reachable) routers, along with router lifetime and prefix lifetime data
- learning the prefixes and the valid and preferred lifetimes to use for stateless (autoconfigured) global unicast addresses (This is required for autoconfiguration of global unicast IPv6 addresses.)
- learning the hop limit for traffic leaving the VLAN interface
- learning the MTU (Maximum Transmission Unit) to apply to frames intended to be routed

## Router Solicitations

When an IPv6 interface becomes operational on the switch, a router solicitation is automatically sent to trigger a router advertisement (RA) from any IPv6 routers reachable on the VLAN. (Router solicitations are sent to the All-Routers multicast address; ff02::2. Refer to “Multicast Application to IPv6 Addressing” on page 3-21.) If an RA is not received within one second of sending the initial router solicitation, the switch sends up to three additional solicitations at intervals of four seconds. If an RA is received, the sending router is added to the switch's default router list and the switch stops sending router solicitations. If an RA is not received, then IPv6 traffic on that VLAN cannot be routed, and the only usable unicast IPv6 address on the VLAN is the link-local address.

**Note** If the switch does not receive a router advertisement after sending the router solicitations, as described above, then no further router solicitations are sent on that VLAN unless a new IPv6 setting is configured, IPv6 on the VLAN is disabled, then re-enabled, or the VLAN itself is disconnected, then reconnected.

## Default IPv6 Router

If IPv6 is enabled on a VLAN where there is at least one accessible IPv6 router, the switch selects a default IPv6 router. (Refer to “Enabling Automatic Configuration of a Global Unicast Address and a Default Router Identity on a VLAN” on page 4-7.)

- If the switch receives router advertisements (RAs) from a single IPv6 router on the same VLAN or subnet, the switch configures a global unicast address and selects the advertising router as the default IPv6 router.
- If multiple IPv6 routers on a VLAN send RAs advertising the same network, the switch configures one global unicast address and selects one router as the default router, based on the router's relative reachability, using factors such as router priority and route cost.
- If multiple IPv6 routers on a VLAN send RAs advertising different subnets, the switch configures a corresponding global unicast address for each RA and selects one of the routers as the default IPv6 router, based on route cost. When multiple RAs are received on a VLAN, the switch uses the router priority and route cost information included in the RAs to identify the default router for the VLAN.

## Router Redirection

With multiple routers on a VLAN, if the default (first-hop) router for an IPv6-enabled VLAN on the switch determines that there is a better first-hop router for reaching a given, remote destination, the default router can redirect the switch to use that other router as the default router. For further information on routing IPv6 traffic, refer to the documentation provided for the IPv6 router.

For related information:

- RFC 2461: “Neighbor Discovery for IP Version 6”

# View IPv6 Gateway, Route, and Router Neighbors

Use these commands to view the switch's current routing table content and connectivity to routers per VLAN. This includes information received in router advertisements from IPv6 routers on VLANs enabled with IPv6 on the switch.

## Viewing Gateway and IPv6 Route Information

***Syntax:*** show ipv6 route [ *ipv6-addr* ] [connected

*This command displays the routes in the switch's IPv6 routing table.*

***ipv6-addr****. Optional. Limits the output to show the gateway to the specified IPv6 address.*

**connected:** *Optional. Limits the output to show only the gateways to IPv6 addresses connected to VLAN interfaces configured on the switch, including the loopback (::1/128) address.*

**Dest:** *The destination address for a detected route.*

**Gateway:** *The IPv6 address or VLAN interface used to reach the destination. (Includes the loopback address.)*

**Type:** *Indicates route type (static, connected, RIP, or OSPF).*

**Distance:** *The route's administrative distance, used to determine the best path to the destination.*

**Metric:** *Indicates the route cost for the selected destination.*

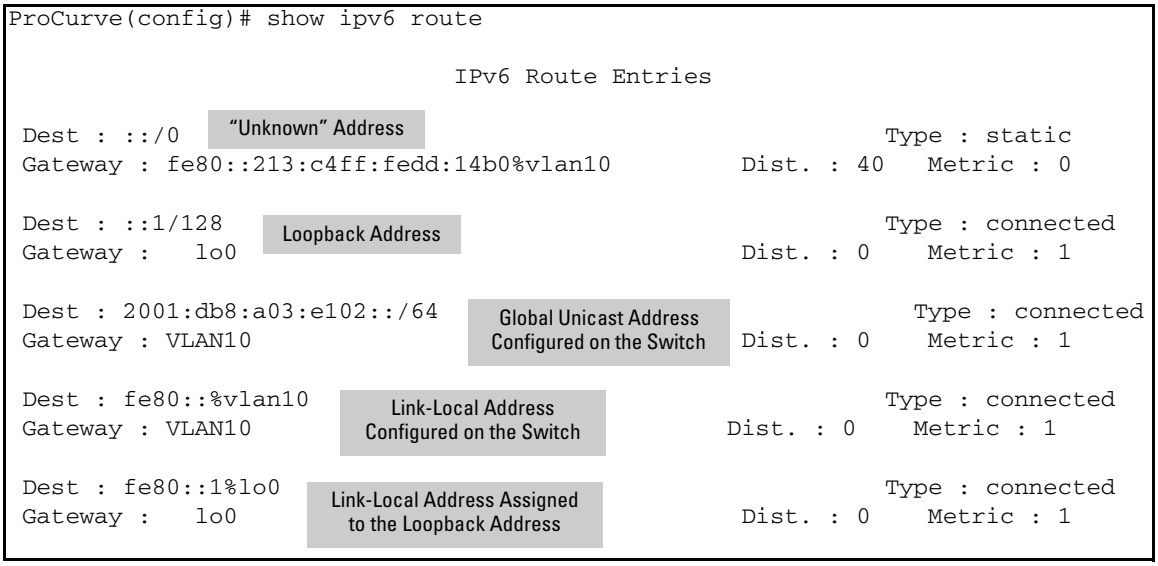



**Figure 4-4. Example of Show IPv6 Route Output**

## Viewing IPv6 Router Information

***Syntax:*** show ipv6 routers [ vlan < *vid* > ]

*This command lists the switch's IPv6 router table entries for all VLANs configured on the switch or for a single VLAN. This output provides information about the IPv6 routers from which routing advertisements (RAs) have been received on the switch.*

**vlan < *vid* >:** Optional. Specifies only the information on IPv6 routers on the indicated VLAN.

**Router Address:** *The IPv6 address of the router interface.*

**Preference:** *The relative priority of prefix assignments received from the router when prefix assignments are also received on the same switch VLAN interface from other IPv6 routers.*

**Interface:** *The VLAN interface on which the path to the router exists.*

**MTU:** *This is the Maximum Transmission Unit (in bytes) allowed for frames on the path to the indicated router.*

**Hop Limit:** *The maximum number of router hops allowed.*

**Prefix Advertised:** *Lists the prefix and prefix size (number of leftmost bits in an address) originating with the indicated router.*

**Valid Lifetime:** *The total time the address is available, including the preferred lifetime and the additional time (if any) allowed for the address to exist in the deprecated state. Refer to "Address Lifetimes" on page 4-32.*

**Preferred Lifetime:** *The length of time during which the address can be used freely as both a source and a destination address for traffic exchanges with other devices. Refer to "Address Lifetimes" on page 4-32.*

**On/Off Link:** Indicates whether the entry source is on the same VLAN as is indicated in the **Interface** field.

For example, figure 4-5 indicates that the switch is receiving router advertisements from a single router that exists on VLAN 10.

```
ProCurve(config)# show ipv6 routers

 IPv6 Router Table Entries

  Router Address : fe80::213:c4ff:fedd:14b0
  Preference     : Medium
  Interface      : VLAN10
  MTU            : 1500
  Hop Limit      : 64

                                                Valid        Preferred    On/Off
  Prefix Advertised                             Lifetime(s)  Lifetime(s)  Link
  --------------------------------------------- ------------ ------------ -------
  2001:db8:a03:e102::/64                       864000       604800       Onlink
```

**Figure 4-5. Example of Show IPv6 Routers Output**

# Address Lifetimes

Every configured IPv6 unicast and anycast address has a lifetime setting that determines how long the address can be used before it must be refreshed or replaced. Some addresses are set as “permanent” and do not expire. Others have both a “preferred” and a “valid” lifetime that specify the duration of their use and availability.

## Preferred Lifetime

This is the length of time during which the address can be used freely as both a source and a destination address for traffic exchanges with other devices. This time span is equal to or less than the valid lifetime also assigned to the address. If this time expires without the address being refreshed, the address becomes deprecated and should be replaced with a new, preferred address. In the deprecated state, an address can continue to be used as a destination for existing communication exchanges, but is not used for new exchanges or as a source for traffic sent from the interface. A new, preferred address and its deprecated counterpart will both appear in the **show ipv6 vlan < *vid* >** output as long as the deprecated address is within its valid lifetime.

## Valid Lifetime

This is the total time the address is available, and is equal to or greater than the preferred lifetime. The valid lifetime enables communication to continue for transactions that began before the address became deprecated. However, in this timeframe, the address should no longer be used for new communications. If this time expires without the deprecated address being refreshed, the address becomes invalid and may be assigned to another interface.

## Sources of IPv6 Address Lifetimes

Manually configured addresses have permanent lifetimes. The prefixes received from router advertisements for global unicast addresses include finite valid and preferred lifetime assignments. Refer to “Unicast Address Prefixes” on page 3-11.

**Table 4-1. IPv6 Unicast Addresses Lifetimes**

| Address Source | Lifetime Criteria |
|---|---|
| Link-Local | Permanent |
| Statically Configured Unicast or Anycast | Permanent |
| Autoconfigured Global | Finite Preferred and Valid Lifetimes |
| DHCPv6-Configured | Finite Preferred and Valid Lifetimes |

A new, preferred address used as a replacement for a deprecated address can be acquired from a manual, DHCPv6, or autoconfiguration source.

# 5

# IPv6 Management Features

## Contents

# Introduction

| Feature | Default | CLI |
|---|---|---|
| Neighbor Cache | n/a | 5-3, 5-5 |
| Telnet6 | Enabled | 5-6, 5-7, 5-8 |
| SNTP Address | None | 5-10 |
| Timep Address | None | 5-13 |
| TFTP | n/a | 5-15 |
| SNMP Trap Receivers | None | 5-21 |

This chapter focuses on the IPv6 application of management features in software release K.13.01 that support both IPv6 and IPv4 operation. For additional information on these features, refer to the current *Management and Configuration Guide* for your switch.

# Viewing and Clearing the IPv6 Neighbors Cache

Neighbor discovery occurs when there is communication between the switch and another, reachable IPv6 device on the same VLAN. A neighbor destination is reachable from a given source address if a confirmation (neighbor solicitation) has been received at the source verifying that traffic has been received at the destination.

The switch maintains an IPv6 neighbor cache that is populated as a result of communication with other devices on the same VLAN. You can view and clear the contents of the neighbor cache using the commands described in this section.

**Anycast Addresses.** Multiple, duplicate addresses configured as Anycast on different devices are special cases of unicast addresses and are not identified as duplicates by the Neighbor Discovery process. Refer to “Anycast Addresses” on page 3-20.

## Viewing the Neighbor Cache

Neighbor discovery occurs when there is communication between IPv6 devices on a VLAN. The Neighbor Cache retains data for a given neighbor until the entry times out. For more on this topic, refer to "Neighbor Discovery (ND)" on page 4-17.

***Syntax:*** show ipv6 neighbors [vlan < *vid* >]

*Displays IPv6 neighbor information currently held in the neighbor cache. After a period without communication with a given neighbor, the switch drops that neighbor's data from the cache. The command lists neighbors for all VLAN interfaces on the switch or for only the specified VLAN. The following fields are included for each entry in the cache:*

**IPv6 Address:** *Lists the 128-bit addresses for the local host and any neighbors (on the same VLAN) with whom there has been recent communication.*

**MAC Address:** *The MAC Address corresponding to each of the listed IPv6 addresses.*

**VLAN < *vid* >:** *Optional. Causes the switch to list only the IPv6 neighbors on a specific VLAN configured on the switch.*

**Type:** *Appears only when VLAN is not specified, and indicates whether the corresponding address is* ***local*** *(configured on the switch) or* **dynamic** *(configured on a neighbor device).*

**Age:** *Appears only when VLAN is specified, and indicates the length of time the entry has remained unused.*

**Port**: *Identifies the switch port on which the entry was learned. If this field is empty for a given address, then the address is configured on the switch itself.*

**State**: *A neighbor destination is reachable from a given source address if confirmation has been received at the source verifying that traffic has been received at the destination. This field shows the reachability status of each listed address:*

- **INCOM** (*Incomplete*)**:** *Neighbor address resolution is in progress, but has not yet been determined.*
- **REACH** (*Reachable*)**:** *The neighbor is known to have been reachable recently.*

*— Continued on the next page. —*

*— Continued from previous page. —*

- **STALE:** *A timeout has occurred for reachability of the neighbor, and an unsolicited discovery packet has been received from the neighbor address. If the path to the neighbor is then used successfully, this state is restored to* **REACH**.
- **DELAY:** *Indicates waiting for a response to traffic sent recently to the neighbor address. The time period for determining the neighbor's reachability has been extended.*
- **PROBE:** *The neighbor may not be reachable. Periodic, unicast neighbor solicitations are being sent to verify reachability.*

```
ProCurve(config)# show ipv6 neighbor

 IPv6 ND Cache Entries

 IPv6 Address                            MAC Address   State Type    Port
 --------------------------------------- ------------- ----- ------- ----
 2001:db8:260:212::101                   0013c4-dd14b0 STALE dynamic A1
 2001:db8:260:214::1:15                  001279-88a100 REACH local
 fe80::1:1                               001279-88a100 REACH local
 fe80::10:27                             001560-7aadc0 REACH dynamic A3
 fe80::213:c4ff:fedd:14b0                0013c4-dd14b0 REACH dynamic A1
```

**Figure 5-1. Example of Neighbor Cache Without Specifying a VLAN**

```
ProCurve(config)# show ipv6 neighbor vlan 10

 IPv6 ND Cache Entries

 IPv6 Address                            MAC Address   State Age           Port
 --------------------------------------- ------------- ----- ------------- ----
 2001:db8:260:212::101                   0013c4-dd14b0 STALE 5h:13m:44s    A1
 2001:db8:260:214::1:15                  001279-88a100 REACH 11h:15m:23s   B17
 fe80:1a3::1:1                           001279-88a100 REACH 9h:35m:11s    B12
 fe80:::10:27                            001560-7aadc0 REACH 22h:26m:12s   A3
 fe80::213:c4ff:fedd:14b0                0013c4-dd14b0 REACH 23 0h:32m:36s A1
```

**Figure 5-2. Example of Neighbor Cache Content for a Specific VLAN**

## Clearing the Neighbor Cache

When there is an event such as a topology change or an address change, the neighbor cache may have too many entries to allow efficient use. Also, if an unauthorized client is answering DAD or normal neighbor solicitations with invalid replies, the neighbor cache may contain a large number of invalid entries and communication with some valid hosts may fail and/or the **show ipv6 neighbors** command output may become too cluttered to efficiently read. In such cases, the fastest way to restore optimum traffic movement on a VLAN may be to statically clear the neighbor table instead of waiting for the unwanted entries to time-out.

***Syntax:*** clear ipv6 neighbors

*Executed at the global config level, this command removes all nonlocal IPv6 neighbor addresses and corresponding MAC addresses from the neighbor cache. (Local IPv6 addresses, that is, IPv6 addresses configured on the VLAN interface for the switch on which the command is executed, are not removed.) Removed addresses are listed in the command output.*

```
ProCurve(config)# clear ipv6 neighbors

2001:db8:260:212::1%vlan10 deleted
fe80:::10:27%vlan10 deleted
fe80::213:c4ff:fedd:14b0%vlan10 deleted
```

**Figure 5-3. Example of Clearing the IPv6 Neighbors Cache**

# Telnet6 Operation

This section describes Telnet operation for IPv6 on the switch. For IPv4 Telnet operation, refer to the *Management and Configuration Guide* for your switch.

## Outbound Telnet6 to Another Device

***Syntax:*** telnet < *link-local-addr* >%vlan< *vid* >
telnet < *global-unicast-addr* >

*Outbound Telnet6 establishes a Telnet session from the switch CLI to another IPv6 device, and includes these options.*

- *Telnet for Link-Local Addresses on the same VLAN requires the link-local address and and interface scope:*

  **< *link-local-addr* >:** *Specifies the link-local IPv6 address of the destination device.*

  **%vlan< *vid* >:** *Suffix specifying the interface on which the destination device is located. No spaces are allowed in the suffix.*

- *Telnet for Global Unicast Addresses requires a global unicast address for the destination. Also, the switch must be receiving router advertisements from an IPv6 gateway router.*

  **< *global-unicast-addr* >:** *Specifies the global IPv6 address of the destination device.*

For example, to Telnet to another IPv6 device having a link-local address of fe80::215:60ff:fe79:8980 and on the same VLAN interface (VLAN 10), you would use the following command:

```
ProCurve(config)# telnet fe80::215:60ff:fe79:980%vlan10
```

If the switch is receiving router advertisements from an IPv6 default gateway router, you can Telnet to a device on the same VLAN or another VLAN or subnet by using its global unicast address. For example, to Telnet to a device having an IPv6 global unicast address of 2001:db8::215:60ff:fe79:980, you would enter the following command:

```
ProCurve(config)# telnet 2001:db8::215:60ff:fe79:980
```

## Viewing the Current Telnet Activity on a Switch

***Syntax:*** show telnet

*This command shows the active incoming and outgoing telnet sessions on the switch (for both IPv4 and IPv6). Command output includes the following:*

**Session:** *The session number. The switch allows one outbound session and up to five inbound sessions.*

**Privilege:** *Manager or Operator.*

**From:** *Console (for outbound sessions) or the source IP address of the inbound session.*

**To:** *The destination of the outbound session, if in use.*

For example, the following figure shows that the switch is running one outbound, IPv4 session and is being accessed by two inbound sessions.

```
ProCurve# show telnet

 Telnet Activity

 ------------------------------------------------------------
 Session  :     1
 Privilege: Manager
 From     : Console
 To       : 10.0.10.140
 ------------------------------------------------------------
 Session  :     2
 Privilege: Manager
 From     : 2620:0:260:212::2:219
 To       :
 ------------------------------------------------------------
 Session  : **  3
 Privilege: Manager
 From     : fe80::2:101
 To       :
```

The * * in the "Session: indicates the session through which **show telnet** was run.

**Figure 5-4. Example of Show Telnet Output with Three Sessions Active**

## Enabling or Disabling Inbound Telnet6 Access

***Syntax:*** [ no ] telnet6-server

*This command is used at the global config level to enable (the default) or disable inbound Telnet6 access to the switch.*

*The* **no** *form of the command disables inbound telnet6.*

***Note:*** *To disable inbound Telnet access completely, you must disable Telnet access for both IPv6 and IPv4. (The command for disabling Telnet4 access is* **no telnet-server**.*)*

For example, to disable Telnet6 access to the switch, you would use this command:

```
ProCurve(config)# no telnet6-server
```

## Viewing the Current Inbound Telnet6 Configuration

***Syntax:*** show console

*This command shows the current configuration of IPv4 and IPv6 inbound telnet permissions, as well as other information. For both protocols, the default setting allows inbound sessions.*

```
LPE-5400-a100(config)# show console

 Console/Serial Link

  Inbound Telnet Enabled [Yes] : Yes
  Inbound Telnet6 Enabled [Yes] : Yes
  Web Agent Enabled [Yes] : Yes
  Terminal Type [VT100] : VT100
  Screen Refresh Interval (sec) [3] : 3
  Displayed Events [All] : All

 Baud Rate [Speed Sense] : speed-sense
  Flow Control [XON/XOFF] : XON/XOFF
  Session Inactivity Time (min) [0] : 0
```

Telnet6 Setting

**Figure 5-5. Show Console Output Showing Default Console Configuration**

# SNTP and Timep

## Configuring (Enabling or Disabling) the SNTP Mode

Software release K.13.01 enables configuration of a global unicast address for IPv6 SNTP time server.

This section lists the SNTP and related commands, including an example of using an IPv6 address. For the details of configuring SNTP on the switch, refer to the chapter titled "Time Protocols" in the *Management and Configuration Guide* for your switch.

The following commands are available at the global config level for SNTP operation.

| Commands Affecting SNTP | Function |
|---|---|
| show sntp | Display the current SNTP configuration. |
| timesync < sntp \| timep > | Enable either SNTP or Timep as the time synchronization method on the switch without affecting the configuration of either. |
| [no] timesync | Enable time synchronization. (Requires a timesync method to also be enabled.) The no version disable time synchronization without affecting the configuration of the current time synchronization method.) |
| [ no ]sntp | Enables SNTP with the current SNTP configuration. The no version disables SNTP without changing the current SNTP configuration. |
| sntp < unicast \| broadcast > | Configures the SNTP mode. (Default: Broadcast) |
| sntp < 30 - 720 > | Changes the interval between time requests. (Default: 720 seconds) |

## Configuring an IPv6 Address for an SNTP Server

**Note**

To use a global unicast IPv6 address to configure an IPv6 SNTP time server on the switch, the switch must be receiving advertisements from an IPv6 router on a VLAN configured on the switch.

To use a link-local IPv6 address to configure an IPv6 SNTP time server on the switch, it is necessary to append **%vlan** followed immediately (without spaces) by the VLAN ID of the VLAN on which the server address is available. (The VLAN must be configured on the switch.) For example:

```
fe80::11:215%vlan10
```

***Syntax:.*** [no ] sntp server priority < 1 - 3 > < *link-local-addr* >%vlan< *vid* > [1 - 7]
[no ] sntp server priority < 1 - 3 > < *global-unicast-addr* > [1 - 7]

*Configures an IPv6 address for an SNTP server.*

**server priority < 1 - 3 >:** *Specifies the priority of the server addressing being configured. When the SNTP mode is set to unicast and more than one server is configured, this value determines the order in which the configured servers will be accessed for a time value. The switch polls multiple servers in order until a response is received or all servers on the list have been tried without success. Up to three server addresses (IPv6 and/or IPv4) can be configured.*

***< link-local-addr >:*** *Specifies the link-local IPv6 address of the destination device.*

**%vlan< *vid* >:** *Suffix specifying the interface on which the destination device is located. No spaces are allowed in the suffix.*

***< global-unicast-addr >:*** *Specifies the global IPv6 address of the destination device.*

**[ 1 - 7 ]:** *This optional setting specifies the SNTP server version expected for the specified server. (Default: 3)*

For example, to configure link-local and global unicast SNTP server addresses of:

- fe80::215:60ff:fe7a:adc0 (on VLAN 10, configured on the switch)
- 2001:db8::215:60ff:fe79:8980

as the priority "1" and "2" SNTP servers, respectively, using version 7, you would enter these commands at the global config level, as shown below.

```
ProCurve(config)# sntp server priority 1
fe80::215:60ff:fe7a:adc0%vlan10 7

ProCurve(config)# sntp server priority 2
2001:db8::215:60ff:fe79:8980 7
```

**Note**

In the preceeding example, using a link-local address requires that you specify the local scope for the address; VLAN 10 in this case. This is always indicated by **%vlan** followed immediately (without spaces) by the VLAN identifier.

***Syntax:.*** show sntp

*Displays the current SNTP configuration, including the following:*

**Time Sync Mode:** *Indicates whether timesync is disabled or set to either SNTP or Timep. (Default:* **timep***)*

**SNTP Mode:** *Indicates whether SNTP uses the broadcast or unicast method of contacting a time server. The broadcast option does not require you to configure a time server address. The unicast option does require configuration of a time server address.*

**Poll Interval:** *Indicates the interval between consecutive time requests to an SNTP server.*

**Priority:** *Indicates the configured priority for the corresponding SNTP server address.*

**SNTP Server Address:** *Lists the currently configured SNTP server addresses.*

**Protocol Version:** *Lists the SNTP server protocol version to expect from the server at the corresponding address.*

For example, the **show sntp** output for the preceeding **sntp server** command example would appear as follows:

```
ProCurve(config)# show sntp

 SNTP Configuration

  Time Sync Mode: Sntp
  SNTP Mode : Broadcast
  Poll Interval (sec) [720] : 719

  Priority SNTP Server Address                          Protocol Version
  -------- ---------------------------------------------- ----------------
  1        2001:db8::215:60ff:fe79:8980                  7
  2        10.255.5.24                                   3
```

This example illustrates the command output when both IPv6 and IPv4 server addresses are configured.

**Figure 5-6. Example of Show SNTP Output with Both an IPv6 and an IPv4 Server Address Configured**

Note that the **show management** command can also be used to display SNTP server information.

## Configuring (Enabling or Disabling) the Timep Mode

Software release K.13.01 enables configuration of a global unicast address for IPv6 Timep time server.

This section lists the Timep and related commands, including an example of using an IPv6 address. For the details of configuring Timep on the switch, refer to the chapter titled "Time Protocols" in the *Management and Configuration Guide* for your switch.

The following commands are available at the global config level for Timep operation.

| Commands Affecting Timep | Function |
|---|---|
| show timep | Display the current timep configuration. |
| timesync < sntp \| timep > | Enable either SNTP or Timep as the time synchronization method on the switch without affecting the configuration of either. |
| ip timep dhcp [ interval < 1 - 9999 >] | Enable Timep operation with a Timep server assignment configured from an IPv4 or IPv6 DHCP server. Optionally change the interval between time requests. |

| | |
|---|---|
| ip timep manual < *ipv6-addr* > [ interval < 1 - 9999 >] | Enable Timep operation with a statically configured IPv6 address for a Timep server. Optionally change the interval between time requests. |
| no ip timep | Disables Timep operation. To re-enable Timep, it is necessary to reconfigure either the DHCP or the static option. |

**Note**

To use a global unicast IPv6 address to configure an IPv6 Timep server on the switch, the switch must be receiving advertisements from an IPv6 router on a VLAN configured on the switch.

To use a link-local IPv6 address to configure an IPv6 Timep server on the switch, it is necessary to append **%vlan** followed (without spaces) by the VLAN ID of the VLAN on which the server address is available. The VLAN must be configured on the switch. For example: `fe80::11:215%vlan10`

***Syntax:.*** ip timep dhcp [ interval < 1 - 9999 >]
ip timep manual < *ipv6-addr* | *ipv4-addr* > [ interval < 1 - 9999 >]

*Used at the global config level to configure a Timep server address.*

***Note:*** *The switch allows one Timep server configuration.*

**timep dhcp:** *Configures the switch to obtain the address of a Timep server from an IPv4 or IPv6 DHCP server.*

**timep manual:** *Specifies static configuration of a Timep server address.*

**< *ipv6-addr* >:** *Specifies the IPv6 address of an SNTP server. Refer to preceeding* **Note**.

**[ Interval < 1 - 9999 > ]:** *This optional setting specifies the interval in minutes between Timep requests. (Default: 720)*

For example, to configure a link-local Timep server address of:

fe80::215:60ff:fe7a:adc0

where the address is on VLAN 10, configured on the switch, you would enter this command at the global config level, as shown below.

```
ProCurve(config)# ip timep manual
fe80::215:60ff:fe7a:adc0%vlan10
```

**Note**

In the preceeding example, using a link-local address requires that you specify the local scope for the address; VLAN 10 in this case. This is always indicated by **%vlan** followed immediately (without spaces) by the VLAN identifier. For a global unicast address, you would enter the address *without* the **%vlan** suffix.

***Syntax:.*** show timep

*Displays the current Timep configuration, including the following:*

**Time Sync Mode:** *Indicates whether timesync is disabled or set to either SNTP or Timep. (Default: Disabled)*

**Timep Mode:** *Indicates whether Timep is configured to use a DHCP server to acquire a Timep server address or to use a statically configured Timep server address.*

**Server Address:** *Lists the currently configured Timep server address.*

**Poll Interval (min) [720]:** *Indicates the interval between consecutive time requests to the configured Timep server.*

For example, the **show timep** output for the preceeding **ip timep manual** command example would appear as follows:

```
ProCurve(config)# sho timep

 Timep Configuration

  Time Sync Mode: Timep
  TimeP Mode [Disabled] : Manual
  Server Address : fe80::215:60ff:fe7a:adc0%vlan10
  Poll Interval (min) [720] : 720
```

**Figure 5-7. Example of Show Timep Output with an IPv6 Server Address Configured**

Note that the **show management** command can also be used to display Timep server information.

# TFTP File Transfers Over IPv6

## TFTP File Transfers over IPv6

You can use TFTP **copy** commands over IPv6 to upload, or download files to and from a physically connected device or a remote TFTP server, including:

- Switch software
- Software images
- Switch configurations
- ACL command files
- Diagnostic data (crash data, crash log, and event log)

For complete information on how to configure TFTP file transfers between the switch and a TFTP server or other host device on the network, refer to the "File Transfers" appendix in the *Management and Configuration Guide* for your switch.

To upload and/or download files to the switch using TFTP in an IPv6 network, you must:

1. Enable TFTP for IPv6 on the switch (see "Enabling TFTP for IPv6" on page 5-16).
2. Enter a TFTP **copy** command with the IPv6 address of a TFTP server in the command syntax (see "Using TFTP to Copy Files over IPv6" on page 5-17).
3. (Optional) To enable auto-TFTP operation, enter the **auto-tftp** command (see "Using Auto-TFTP for IPv6" on page 5-19).

## Enabling TFTP for IPv6

TFTP for IPv6 is enabled by default on the switch. However, if it is disabled, you can re-enable it by specifying TFTP client or server functionality with the **tftp6** <**client** | **server**> command. Enter the **tftp6** <**client** | **server**> command at the global configuration level.

***Syntax:*** tftp6 <client | server>

*Enables TFTP for IPv6 client or server functionality so that the switch can:*

- *Use TFTP client functionality to access IPv6-based TFTP servers in the network to receive downloaded files.*
- *Use TFTP server functionality to be accessed by other IPv6 hosts to upload files to an IPv6 host.*

### Usage Notes

To disable all TFTP client or server operation on the switch except for the auto-TFTP feature, enter the **no tftp6** <**client** | **server**> command. To re-enable TFTP client or server operation, re-enter the **tftp6** <**client** | **server**> command.

When TFTP is disabled, instances of TFTP in the CLI **copy** command and the Menu interface “Download OS” screen become unavailable.

The **no tftp6** <**client** | **server**> command does not disable auto-TFTP operation. For more information, see “Using Auto-TFTP for IPv6” on page 5-19.

## Using TFTP to Copy Files over IPv6

Use the TFTP **copy** commands described in this section to:

- Download specified files from a TFTP server to a switch on which TFTP client functionality is enabled.
- Upload specified files from a switch, on which TFTP server functionality is enabled, to a TFTP server.

***Syntax:*** copy tftp < *target* > < *ipv6-addr* > < *filename* >

*Copies (downloads) a data file from a TFTP server at the specified IPv6 address to a target file on a switch that is enabled with TFTP server functionality.*

**< *ipv6-addr* >:** *If this is a link-local address, use this IPv6 address format:*

fe80::< *device-id* >%vlan< *vid* >

*For example:* `fe80::123%vlan10`

*If this is a global unicast or anycast address, use this IPv6 format:*

< *ipv6-addr* >

*For example:* `2001:db8::123`

**< *target* >** *is one of the following values:*

- **autorun-cert-file**: Copies an autorun trusted certificate to the switch.
- **autorun-key-file**: Copies an autorun key file to the switch.
- **command-file**: Copies a file stored on a remote host and executes the ACL command script on the switch. Depending on the ACL commands stored in the file, one of the following actions is performed in the running-config file on the switch:
  - *A new ACL is created.*
  - *An existing ACL is replaced.*
  - **match**, **permit**, *or* **deny** *statements are added to an existing ACL.*

  *For more information on ACLs, refer to "Creating an ACL Offline" in the Access Control Lists (ACLs) chapter in the Access Security Guide.*
- **config < *filename* >:** *Copies the contents of a file on a remote host to a configuration file on the switch.*

- **flash** < **primary** | **secondary** >: *Copies a software file stored on a remote host to primary or secondary flash memory on the switch. To run a newly downloaded software image, enter the* **reload** *or* **boot system flash** *command.*
- **pub-key-file**: *Copies a public-key file to the switch.*
- **startup-config**: *Copies a configuration file on a remote host to the startup configuration file on the switch.*

.

***Syntax:*** copy <*source* > tftp < *ipv6-addr* > < *filename* > < pc | unix >

*Copies (uploads) a source data file on a switch that is enabled with TFTP server functionality to a file on the TFTP server at the specified IPv6 address, where <source> is one of the following values:*

- **command-output < cli-command >**: *Copies the output of a CLI command to the specified file on a remote host.*
- **config < *filename* >:** *Copies the specified configuration file to a remote file on a TFTP server.*
- **crash-data < slot-id | master >:** *Copies the contents of the crash data file to the specified file path on a remote host. The crash data is software-specific and used to determine the cause of a system crash. You can copy crash information from an individual slot or from the master crash file on the switch.*
- **crash-log < *slot-id* | master >:** *Copies the contents of the crash log to the specified file path on a remote host. The crash log contains processor-specific operational data that is used to determine the cause of a system crash. You can copy the contents of the crash log from an individual slot or from the master crash log on the switch.*
- **event-log**: *Copies the contents of the Event Log on the switch to the specified file path on a remote host.*
- **flash < primary | secondary >:** *Copies the software file used as the primary or secondary flash image on the switch to a file on a remote host.*
- **startup-config**: *Copies the startup configuration file in flash memory to a remote file on a TFTP server.*
- **running-config**: *Copies the running configuration file to a remote file on a TFTP server.*

**< *ipv6-addr* >:** *If this is a link-local address, use this IPv6 address format:*

fe80::< *device-id* >%vlan< *vid* >

*For example:* `fe80::123%vlan10`

*If this is a global unicast or anycast address, use this IPv6 format:*

< *ipv6-addr* >

*For example:* `2001:db8::123`

## Using Auto-TFTP for IPv6

The auto-TFTP for IPv6 feature automatically downloads a software image to a switch, on which TFTP client functionality is enabled, from a specified IPv6-based device at switch startup. You must reboot the switch to implement the downloaded software image by entering the **boot system flash primary** or **reload** command

***Syntax:*** auto-tftp <*ipv6-addr*> <*filename*>

*Configures the specified software file on the TFTP server at the specified IPv6 address to be automatically downloaded into primary flash memory at switch startup.*

> ***Note:*** *In order for the auto-TFTP feature to copy a software image to primary flash memory, the version number of the downloaded software file (for example, E.10.78) must be different from the version number of the primary flash image.*

*The* **no** *form of the command disables auto-TFTP operation. This command deletes the* **auto-tftp** *entry from the startup configuration, and prevents auto-tftp operation if the switch reboots.*

*The* **no auto-tftp** *command does not affect the current TFTP-enabled configuration on the switch.*

# SNMP Management for IPv6

As with SNMP for IPv4, you can manage a switch via SNMP from an IPv6-based network management station by using an application such as ProCurve Manager (PCM) or ProCurve Manager Plus (PCM+). (For more on PCM and PCM+, go to the ProCurve Networking web site at **www.procurve.com**.)

## SNMP Features Supported

The same SNMP for IPv4 features are supported over IPv6:

- access to a switch using SNMP version 1, version 2c, or version 3
- enhanced security with the configuration of SNMP communities and SNMPv3 user-specific authentication password and privacy (encryption) settings
- SNMP notifications, including:
  - SNMP version 1 or SNMP version 2c traps
  - SNMPv2c informs
  - SNMPv3 notification process, including traps
- Advanced RMON (Remote Monitoring) management
- ProCurve Manager or ProCurve Manager Plus management applications
- Flow sampling using sFlow
- Standard MIBs, such as the Bridge MIB (RFC 1493) and the Ethernet MAU MIB (RFC 1515)

## SNMP Configuration Commands Supported

IPv6 addressing is supported in the following SNMP configuration commands: For more information on each SNMP configuration procedure, refer to the "Configuring for Network Management Applications" chapter in the current *Management and Configuration Guide* for your switch.

### SNMPv1 and V2c

***Syntax:.*** snmp-server host < *ipv4-addr* | *ipv6-addr* > < *community-name* > [none | all | non-info | critical | debug] [inform [retries < *count* >] [timeout < *interval* >]]

*Executed at the global config level to configure an SNMP trap receiver to receive SNMPv1 and SNMPv2c traps, SNMPv2c informs, and (optionally) event log messages*

### SNMPv3

***Syntax:*** snmpv3 targetaddress < *name* > params < *parms_name* > <*ipv4-addr* | *ipv6-addr*> [addr-mask < *ip4-addr* >] [filter < none | debug | all | not-info | critical>] [max-msg-size < 484-65535 >] [port-mask < *tcp-udp port* >] [retries < 0 - 255 >] [taglist <*tag_name*> ] [timeout < 0 - 2147483647 >] [udp-port *port-number*]

*Executed at the global config level to configure an SNMPv3 management station to which notifications (traps and informs) are sent.*

**Note**

IPv6 is not supported in the configuration of an interface IPv6 address as the default source IP address used in the IP headers of SNMP notifications (traps and informs) or responses sent to SNMP requests. Only IPv4 addresses are supported in the following configuration commands:

snmp-server trap-source < *ipv4-addr* | loopback < 0-7 >>

snmp-server response-source [*dst-ip-of-request* | *ipv4-addr* | loopback < 0-7 >]

IPv6 addresses are supported in SNMP **show** command output as shown in Figure 5-8 and Figure 5-9.

The **show snmp-server** command displays the current SNMP policy configuration, including SNMP communities, network security notifications, link-change traps, trap receivers (including the IPv4 or IPv6 address) that can receive SNMPv1 and SNMPv2c traps, and the source IP (interface) address used in IP headers when sending SNMP notifications (traps and informs) or responses to SNMP requests.

```
ProCurve(config)# show snmp-server

 SNMP Communities

  Community Name       MIB View Write Access
  -------------------- -------- ------------
  public               Manager  Unrestricted
  marker               Manager  Unrestricted

 Trap Receivers

  Link-Change Traps Enabled on Ports [All] : All

  Traps Category                  Current Status
  --------------------------      --------------
  SNMP Authentication           : Extended
  Password change               : Enabled
  Login failures                : Enabled
  Port-Security                 : Enabled
  Authorization Server Contact  : Enabled
  DHCP-Snooping                 : Enabled
  Dynamic ARP Protection        : Enabled

  Address                 Community              Events   Type   Retry   Timeout
  ---------------------- ---------------------- -------- ------ ------- -------
  15.29.17.218            public                 All      trap   3       15
  15.29.17.219            public                 Critical trap   3       15
  2620:0000:0260:02I1
    :0217:a4ff:feff:1f70  marker                 Critical trap   3       15

 Excluded MIBs


 Snmp Response Pdu Source-IP Information

  Selection Policy   : rfc1517

 Trap Pdu Source-IP Information

  Selection Policy   : rfc1517
```

An IPv6 address is displayed on two lines.

**Figure 5-8. "show snmp-server" Command Output with IPv6 Address**

The **show snmpv3 targetaddress** command displays the configuration (including the IPv4 or IPv6 address) of the SNMPv3 management stations to which notification messages are sent.

```
ProCurve(config)# show snmpv3 targetaddress


 snmpTargetAddrTable [rfc2573]

  Target Name               IP Address              Parameter
  ------------------------- ----------------------- ---------------------------
  1                         15.29.17.218            1
  2                         15.29.17.219            2
  PP.217                    15.29.17.217            marker_p
  PP.218                    2620:0:260:211
                              :217:a4ff:feff:1f70   marker_p
```

An IPv6 address is displayed on two lines.

**Figure 5-9. "show snmpv3 targetaddress" Command Output with IPv6 Address**

# IP Preserve for IPv6

IPv6 supports the IP Preserve feature, which allows you to copy a configuration file from a TFTP server to multiple switches without overwriting the IPv6 address and subnet mask on VLAN 1 (default VLAN) in each switch, and the Gateway IPv6 address assigned to the switch.

To configure IP Preserve, enter the **ip preserve** statement at the end of the configuration file that will be downloaded from a TFTP server. (Note that you do not invoke IP Preserve by entering a command from the CLI).

```
; J8697A Configuration Editor; Created on release #K.13.01
hostname "ProCurve"
time daylight-time-rule None

   *
   *
   *
   *
   *
   *
password manager
password operator
ip preserve
```

Entering an **ip preserve** statement as the last line in a configuration file stored on a TFTP server allows you to download and execute the file as the startup-config file on an IPv6 switch. When the switch reboots, the configuration settings in the downloaded file are implemented without changing the IPv6 address and gateway assigned to the switch as shown in Figure 5-11.

**Figure 5-10. Example of How to Enter IP Preserve in a Configuration File**

To download an IP Preserve configuration file to an IPv6-based switch, enter the TFTP **copy** command as described in "TFTP File Transfers over IPv6" on page 5-15 to copy the file as the new startup-config file on a switch.

When you download an IP Preserve configuration file, the following rules apply:

- If the switch's current IPv6 address for VLAN 1 was statically configured and not dynamically assigned by a DHCP/Bootp server, the switch reboots and retains its current IPv6 address, subnet mask, and gateway address. All other configuration settings in the downloaded configuration file are applied.
- If the switch's current IPv6 address for VLAN 1 was assigned from a DHCP server and not statically configured, IP Preserve is suspended. The IPv6 addressing specified in the downloaded configuration file is implemented when the switch copies the file and reboots.
  - If the downloaded file specifies DHCP/Bootp as the source for the IPv6 address of VLAN 1, the switch uses the IPv6 address assigned by the DHCP/Bootp server.
  - If the file specifies a dedicated IPv6 address and subnet mask for VLAN 1 and a Gateway IPv6 address, the switch implements these settings in the startup-config file.

To verify how IP Preserve was implemented in a switch, after the switch reboots, enter the **show run** command. Figure 5-11 shows an example in which all configurations settings have been copied into the startup-config file except for the IPv6 address of VLAN 1 (2001:db8::214:c2ff:fe4c:e480) and the default IPv6 gateway (2001:db8:0:7::5), which were retained.

Note that if a switch received its IPv6 address from a DHCP server, the "ip address" field under "vlan 1" would display: **dhcp-bootp**.

```
ProCurve(config)# show run

Running configuration:

; J8715A Configuration Editor; Created on release #K.13.01

hostname "ProCurve"
module 1 type J8702A
module 2 type J8705A
trunk A11-A12 Trk1 Trunk
ip default-gateway 2001:db8:0:7::5
snmp-server community "public" Unrestricted
vlan 1
   name "DEFAULT_VLAN"
   untagged A1-A10,A13-A24,B1-B24,Trk1
   ip address 2001:db8::214:c2ff:fe4c:e480
   exit
spanning-tree Trk1 priority 4
password manager
password operator
```

Because the switch's IPv6 address and default gateway were statically configured (not assigned by a DHCP server), when the switch boots up with the IP Preserve startup configuration file (see Figure 5-10), its current IPv6 address, subnet mask, and default gateway are not changed.

If a switch's current IP address was acquired from a DHCP/Bootp server, the IP Preserve statement is ignored and the IP addresses in the downloaded configuration file are implemented.

**Figure 5-11. Configuration File with Dedicated IP Addressing After Startup with IP Preserve**

For more information on how to use the IP Preserve feature, refer to the "Configuring IP Addressing" chapter in the current *Management and Configuration Guide* for your ProCurve switch.

# 6

# IPv6 Management Security Features

## Contents

# IPv6 Management Security

This chapter describes management security features that are IPv6 counterparts of IPv4 management security features on the switches covered by this guide.

| Feature | Default | CLI |
|---|---|---|
| configure authorized IP managers for IPv6 | disabled | 6-5 |
| configuring secure shell for IPv6 | disabled | 6-15 |
| enabling secure copy and secure FTP for IPv6 | disabled | 6-18 |

This chapter describes the following IPv6-enabled management security features included in software release K.13.01:

- Authorized IP Managers for IPv6
- Secure Shell for IPv6
- Secure Copy and Secure FTP for IPv6

# Authorized IP Managers for IPv6

The Authorized IP Managers feature uses IP addresses and masks to determine which stations (PCs or workstations) can access the switch through the network. This feature supports switch access through:

- Telnet and other terminal emulation applications
- Web browser interface
- SNMP (with a correct community name)

As with the configuration of IPv4 management stations, the Authorized IP Managers for IPv6 feature allows you to specify the IPv6-based stations that can access the switch.

## Usage Notes

- You can configure up to ten authorized IPv4 and IPv6 manager *addresses* on a switch, where each address applies to either a single management station or a group of stations. Each authorized manager address consists of an IPv4 or IPv6 address and a mask that determines the individual management stations that are allowed access.
  - You configure authorized IPv4 manager addresses using the **ip authorized-managers** command. For more information, refer to the "Using Authorized IP Managers" chapter in the *Access Security Guide*.
  - You configure authorized IPv6 manager addresses using the **ipv6 authorized-managers** command. For more information, see "Configuring Authorized IP Managers for Switch Access" on page 6-5.
- You can block all IPv4-based or all IPv6-based management stations from accessing the switch by entering the following commands:
  - To block access to all IPv4 manager addresses while allowing access to IPv6 manager addresses, enter the **ip authorized-managers 0.0.0.0** command.
  - To block access to all IPv6 manager addresses while allowing access to IPv4 manager addresses, enter the **ipv6 authorized-managers ::** command. (The double colon represents an IPv6 address that consists of all zero's: **0:0:0:0:0:0:0:0**.)

- You configure each authorized manager address with Manager or Operator-level privilege to access the switch in a Telnet, SNMPv1, or SNMPv2c session. (Access privilege for SSH, SNMPv3, and web browser sessions are configured through the access application, not through the Authorized IP Managers feature.)
  - Manager privilege allows full access to all web browser and console interface screens for viewing, configuration, and all other operations available in these interfaces.
  - Operator privilege allows read-only access from the web browser and console interfaces.
- When you configure station access to the switch using the Authorized IP Managers feature, the settings take precedence over the access configured with local passwords, TACACS+ servers, RADIUS-assigned settings, port-based (802.1X) authentication, and port security settings.

  As a result, the IPv6 address of a networked management device must be configured with the Authorized IP Managers feature before the switch can authenticate the device using the configured settings from other access security features. If the Authorized IP Managers feature disallows access to the device, then access is denied. Therefore, with authorized IP managers configured, logging in with the correct passwords is not sufficient to access a switch through the network unless the station requesting access is also authorized in the switch's Authorized IP Managers configuration.

## Configuring Authorized IP Managers for Switch Access

To configure one or more IPv6-based management stations to access the switch using the Authorized IP Managers feature, enter the **ipv6 authorized-managers** command

***Syntax:*** ipv6 authorized-managers *<ipv6-addr>* [*ipv6-mask*] [access <operator | manager>]

*Configures one or more authorized IPv6 addresses to access the switch, where:*

***ipv6-mask*** *specifies the mask that is applied to an IPv6 address to determine authorized stations. For more information, see "Using a Mask to Configure Authorized Management Stations" on page 6-5. Default:* **FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF**.

**access <operator | manager>** *specifies the level of access privilege granted to authorized stations and applies only to Telnet, SNMPv1, and SNMPv2c access. Default:* **Manager.**

***Note:*** *The Authorized IP Manager feature does not support the configuration of access privileges on authorized stations that use an SSH, SNMPv3, or the web browser session to access the switch. For these sessions, access privilege is configured with the access application.*

## Using a Mask to Configure Authorized Management Stations

The *ipv6-mask* parameter controls how the switch uses an IPv6 address to determine the IPv6 addresses of authorized manager stations on your network. For example, you can specify a mask that authorizes:

- Single station access
- Multiple station access

**Note**

Mask configuration is a method for determining the valid IPv6 addresses that are authorized for management access to the switch. In the Authorized IP Managers feature, the mask serves a different purpose than an IPv6 subnet mask and is applied in a different manner.

### Configuring Single Station Access

To authorize only one IPv6-based station for access to the switch, enter the IPv6 address of the station and set the mask to **FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF**.

**Notes**

If you do not enter a value for the *ipv6-mask* parameter when you configure an authorized IPv6 address, the switch automatically uses **FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF** as the default mask (see "Configuring Authorized IP Managers for Switch Access" on page 6-5).

If you have ten or fewer management and/or operator stations for which you want to authorize access to the switch, it may be more efficient to configure them by entering each IPv6 address with the default mask in a separate **ipv6 authorized-managers** command.

When used in a mask, "**FFFF**" specifies that each bit in the corresponding 16-bit (hexadecimal) block of an authorized station's IPv6 address must be identical to the same "on" or "off" setting in the IPv6 address entered in the **ipv6 authorized-managers** command. (The binary equivalent of **FFFF** is 1111 1111 1111 1111, where **1** requires the same "on" or "off" setting in an authorized address.)

For example, as shown in Figure 6-1, if you configure a link-local IPv6 address of FE80::202:B3FF:FE1E:8329 with a mask of **FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF**, only a station having an IPv6 address of FE80::202:B3FF:FE1E:8329 has management access to the switch.

| | 1st Block | 2nd Block | 3rd Block | 4th Block | 5th Block | 6th Block | 7th Block | 8th Block | Manager- or Operator-Level Access |
|---|---|---|---|---|---|---|---|---|---|
| **IPv6 Mask** | FFFF | FFFF | FFFF | FFFF | FFFF | FFFF | FFFF | FFFF | The "FFFF" in each hexadecimal block of the mask specifies that only the exact value of each bit in the corresponding block of the IPv6 address is allowed. This mask allows management access only to a station having an IPv6 address of FE80::202:B3FF:FE1E:8329. |
| **IPv6 Address** | FE80 | 0000 | 0000 | 0000 | 202 | B3FF | FE1E | 8329 | |

**Figure 6-1. Mask for Configuring a Single Authorized IPv6 Manager Station**

## Configuring Multiple Station Access

To authorize multiple stations to access the switch without having to re-enter the **ipv6 authorized-managers** command for each station, carefully select the IPv6 address of an authorized IPv6 manager and an associated mask to authorize a range of IPv6 addresses.

As shown in Figure 6-2, if a bit in any of the 4-bit binary representations of a hexadecimal value in a mask is "on" (set to 1), then the corresponding bit in the IPv6 address of an authorized station must match the "on" or "off' setting of the same bit in the IPv6 address you enter with the **ipv6 authorized-managers** command.

Conversely, in a mask, a "0" binary bit means that either the "on" or "off" setting of the corresponding IPv6 bit in an authorized address is valid and does not have to match the setting of the same bit in the specified IPv6 address.

Figure 6-2 shows the binary expressions represented by individual hexadecimal values in an *ipv6-mask* parameter.

| Hexadecimal Value in an IPv6 Mask | Binary Equivalent |
|---|---|
| 0 | 0000 |
| 1 | 0001 |
| 2 | 0010 |
| 3 | 0011 |
| 4 | 0100 |
| 5 | 0101 |
| 6 | 0110 |
| 7 | 0111 |
| 8 | 1000 |
| 9 | 1001 |
| A | 1010 |
| B | 1011 |
| C | 1100 |
| D | 1101 |
| E | 1110 |
| F | 1111 |

**Figure 6-2. Hexadecimal Mask Values and Binary Equivalents**

**Example.** Figure 6-3 shows an example in which a mask that authorizes switch access to four management stations is applied to the IPv6 address: **2001:DB8:0000:0000:244:17FF:FEB6:D37D**. The mask is: **FFFF:FFFF:FFFF:FFF8:FFFF:FFFF:FFFF:FFFC**.

| | 1st Block | 2nd Block | 3rd Block | 4th Block | 5th Block | 6th Block | 7th Block | 8th Block | Manager- or Operator-Level Access |
|---|---|---|---|---|---|---|---|---|---|
| **IPv6 Mask** | FFFF | FFFF | FFFF | FFFF | FFFF | FFFF | FFFF | FFFC | The "F" value in the first 124 bits of the mask specifies that only the exact value of each corresponding bit in an authorized IPv6 address is allowed. However, the "C" value in the last four bits of the mask allows four possible combinations (D37C, D37D, D37E, and D37F) in the last block of an authorized IPv6 address. |
| **IPv6 Address** | 2001 | DB8 | 0000 | 0000 | 244 | 17FF | FEB6 | D37D | |

**Figure 6-3. Example: Mask for Configuring Four Authorized IPv6 Manager Stations**

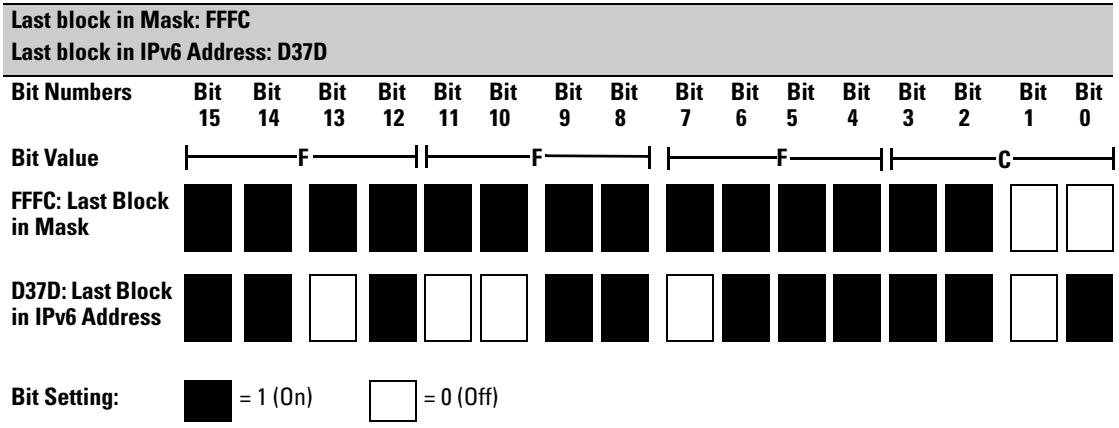



**Figure 6-4. Example: How a Mask Determines Four Authorized IPv6 Manager Addresses**

As shown in Figure 6-4, if you use a mask of **FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFC** with an IPv6 address, you can authorize four IPv6-based stations to access the switch. In this mask, all bits except the last two are set to 1 ("on"); the binary equivalent of hexadecimal **C** is 1100.

Therefore, this mask requires the first corresponding 126 bits in an authorized IPv6 address to be the same as in the specified IPv6 address: 2001:DB8:0000:0000:244:17FF:FEB6:D37C. However, the last two bits are set

to 0 (“off”) and allow the corresponding bits in an authorized IPv6 address to be either “on” or “off”. As a result, only the four IPv6 addresses shown in Figure 6-5 are allowed access.

| | 1st Block | 2nd Block | 3rd Block | 4th Block | 5th Block | 6th Block | 7th Block | 8th Block |
|---|---|---|---|---|---|---|---|---|
| **IPv6 Mask** | FFFF | FFFF | FFFF | FFFF | FFFF | FFFF | FFFF | FFFC |
| **IPv6 Address Entered with the “ipv6 authorized-managers” Command** | 2001 | DB8 | 0000 | 0000 | 244 | 17FF | FEB6 | D37D |
| **Other Authorized IPv6 Addresses** | 2001 | DB8 | 0000 | 0000 | 244 | 17FF | FEB6 | D37C |
| | 2001 | DB8 | 0000 | 0000 | 244 | 17FF | FEB6 | D37E |
| | 2001 | DB8 | 0000 | 0000 | 244 | 17FF | FEB6 | D37F |

**Figure 6-5. Example: How Hexadecimal C in a Mask Authorizes Four IPv6 Manager Addresses**

**Example.** Figure 6-6 shows an example in which a mask is applied to the IPv6 address: **2001:DB8:0000:0000:244:17FF:FEB6:D37D/64**. The specified mask **FFFF:FFFF:FFFF:FFF8:FFFF:FFFF:FFFF:FFFF** configures eight management stations as authorized IP manager stations.

Note that, in this example, the IPv6 mask is applied as follows:

- Eight management stations in different subnets are authorized by the value of the fourth block (**FFF8**) in the 64-bit prefix ID (**FFFF:FFFF:FFFF:FFF8**) of the mask. (The fourth block of the prefix ID is often used to define subnets in an IPv6 network.)

  The binary equivalent of **FFF8** that is used to specify valid subnet IDs in the IPv6 addresses of authorized stations is: 1111 1111 1111 1000.

  The three “off” bits (1000) in the last part of the this block (**FFF8**) of the mask allow for eight possible authorized IPv6 stations:
  2001:DB8:0000:0000:244:17FF:FEB6:D37D
  2001:DB8:0000:0001:244:17FF:FEB6:D37D
  2001:DB8:0000:0002:244:17FF:FEB6:D37D
  2001:DB8:0000:0003:244:17FF:FEB6:D37D
  2001:DB8:0000:0004:244:17FF:FEB6:D37D
  2001:DB8:0000:0005:244:17FF:FEB6:D37D
  2001:DB8:0000:0006:244:17FF:FEB6:D37D
  2001:DB8:0000:0007:244:17FF:FEB6:D37D

- Each authorized station has the same 64-bit device ID (**244:17FF:FEB6:D37D**) because the value of the last four blocks in the mask is **FFFF** (binary value 1111 1111).

  **FFFF** requires all bits in each corresponding block of an authorized IPv6 address to have the same "on" or "off" setting as the device ID in the specified IPv6 address. In this case, each bit in the device ID (last four blocks) in an authorized IPv6 address is fixed and can be only one value: 244:17FF:FEB6:D37D.

| | 1st Block | 2nd Block | 3rd Block | 4th Block | 5th Block | 6th Block | 7th Block | 8th Block | Manager- or Operator-Level Access |
|---|---|---|---|---|---|---|---|---|---|
| **IPv6 Mask** | FFFF | FFFF | FFFF | FFF8 | FFFF | FFFF | FFFF | FFFF | In this example, the IPv6 mask allows up to four stations in different subnets to access the switch. This authorized IP manager configuration is useful if only management stations are specified by the authorized IPv6 addresses. Refer to Figure 6-4 for how the bitmap of the IPv6 mask determines authorized IP manager stations. |
| **Authorized IPv6 Address** | 2001 | DB8 | 0000 | 0000 | 244 | 17FF | FEB6 | D37D | |

**Figure 6-6. Example: Mask for Configuring Authorized IPv6 Manager Stations in Different Subnets**

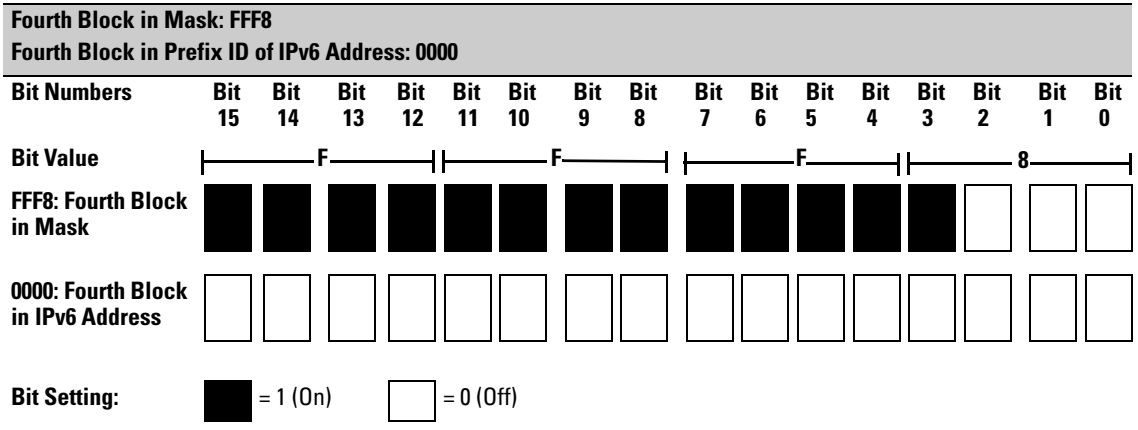



**Figure 6-7. Example: How a Mask Determines Authorized IPv6 Manager Addresses by Subnet**

Figure 6-7 shows the bits in the fourth block of the mask that determine the valid subnets in which authorized stations with an IPv6 device ID of **244:17FF:FEB6:D37D** reside.

**FFF8** in the fourth block of the mask means that bits 3 - 15 of the block are fixed and, in an authorized IPv6 address, must correspond to the "on" and "off" settings shown for the binary equivalent 0000 in the fourth block of the IPv6 address. Conversely, bits 0 - 2 are variable and, in an authorized IPv6 address, may be either "on" (1) or "off" (0).

As a result, assuming that the seventh and eighth bytes (fourth hexadecimal block) of an IPv6 address are used as the subnet ID, only the following binary expressions and hexadecimal subnet IDs are supported in this authorized IPv6 manager configuration:

| Authorized Subnet ID in Fourth Hexadecimal Block of IPv6 Address | Binary Equivalent |
|---|---|
| 0000 | 0000 0000 |
| 0001 | 0000 0001 |
| 0002 | 0000 0010 |
| 0003 | 0000 0011 |
| 0004 | 0000 0100 |
| 0005 | 0000 0101 |
| 0006 | 0000 0110 |
| 0007 | 0000 0111 |

**Figure 6-8. Binary Equivalents of Authorized Subnet IDs (in Hexadecimal)**

## Displaying an Authorized IP Managers Configuration

Use the **show ipv6 authorized-managers** command to list the IPv6 stations authorized to access the switch; for example:

```
ProCurve# show ipv6 authorized-managers

 IPv6 Authorized Managers
-------------------------------------

 Address : 2001:db8:0:7::5
 Mask    : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff
 Access  : Manager

 Address : 2001:db8::a:1c:e3:3
 Mask    : ffff:ffff:ffff:ffff:ffff:ffff:ffff:fffe
 Access  : Manager

 Address : 2001:db8::214:c2ff:fe4c:e480
 Mask    : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff
 Access  : Manager

 Address : 2001:db8::10
 Mask    : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ff00
 Access  : Operator
```

**Figure 6-9. Example of "show ipv6 authorized-managers" Output**

By analyzing the masks displayed in Figure 6-9, the following IPv6 stations are granted access:

| Mask | Authorized IPv6 Addresses | Number of Authorized Addresses |
|---|---|---|
| FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFC | 2001:db8:0:7::4 through 2001:db8:0:7::7 | 4 |
| FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFE | 2001:db8::a:1c:e3:2 and 2001:db8::a:1c:e3:3 | 2 |
| FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF | 2001:db8::214:c2ff:fe4c:e480 | 1 |
| FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FF00 | 2001:db8::0 through 2001:db8::FF | 256 |

**Figure 6-10. How Masks Determine Authorized IPv6 Manager Addresses**

## Additional Examples of Authorized IPv6 Managers Configuration

**Authorizing Manager Access.** The following IPv6 commands authorize manager-level access for one link-local station at a time. Note that when you enter a link-local IPv6 address with the **ipv6 authorized-managers** command, you must also enter a VLAN ID in the format: **%vlan**<*vlan-id*>.

```
ProCurve(config)# ipv6 authorized-managers
fe80::07be:44ff:fec5:c965%vlan2

ProCurve(config)# ipv6 authorized-managers
fe80::070a:294ff:fea4:733d%vlan2

ProCurve(config)# ipv6 authorized-managers
fe80::19af:2cff:fe34:b04a%vlan5
```

If you do not enter an *ipv6-mask* value when you configure an authorized IPv6 address, the switch automatically uses **FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF** as the default IPv6 mask. Also, if you do not specify an **access** value to grant either Manager- or Operator-level access, by default, the switch assigns Manager access. For example:

```
ProCurve# ipv6 authorized-managers 2001:db8::a8:1c:e3:69
ProCurve# show ipv6 authorized-managers

 IPv6 Authorized Managers
---------------------------

Address : 2001:db8::a8:1c:e3:69
Mask    : ffff:ffff:ffff:ffff:ffff:ffff:ffff:ffff
Access  : Manager
```

If you do not enter a value for *ipv6-mask* in the **ipv6 authorized-managers** command, the default mask of FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF: is applied. The default mask authorizes only the specified station (see "Configuring Single Station Access" on page 6-5).

**Figure 6-11. Default IPv6 Mask**

The next IPv6 command authorizes operator-level access for sixty-four IPv6 stations: thirty-two stations in the subnets defined by 0x0006 and 0x0007 in the fourth block of an authorized IPv6 address:

```
ProCurve(config)# ipv6 authorized-managers
2001:db8:0000:0007:231:17ff:fec5:c967
ffff:ffff:ffff:fffe:ffff:ffff:ffff:ffe0 access operator
```

The following **ipv6 authorized-managers** command authorizes a single, automatically generated (EUI-64) IPv6 address with manager-level access privilege:

```
ProCurve(config)# ipv6 authorized-managers
::223:04ff:fe03:4501 ::ffff:ffff:ffff:ffff
```

**Editing an Existing Authorized IP Manager Entry.** To change the mask or access level for an existing authorized IP manager entry, enter the IPv6 address with the new value(s). Any parameters not included in the command are reset to their default values.

The following command replaces the existing mask and access level for IPv6 address 2001:DB8::231:17FF:FEC5:C967 with **FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FF00** and **operator**:

```
ProCurve(config)# ipv6 authorized-managers
2001:db8::231:17ff:fec5:c967
ffff:ffff:ffff:ffff:ffff:ffff:ffff:ff00 access operator
```

The following command replaces the existing mask and access level for IPv6 address 2001:DB8::231:17FF:FEC5:3E61 with **FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF** and **manager** (the default values). Note that it is not necessary to enter either of these parameters:

```
ProCurve(config)# ipv6 authorized-managers
2001:db8::a05b:17ff:fec5:3f61
```

**Deleting an Authorized IP Manager Entry.** Enter only the IPv6 address of the configured authorized IP manager station that you want to delete with the **no** form of the command; for example:

```
ProCurve(config)# no ipv6 authorized-managers
2001:db8::231:17ff:fec5:3e61
```

# Secure Shell for IPv6

The Secure Shell (SSH) for IPv6 feature provides the same Telnet-like functions through encrypted, authenticated transactions as SSH for IPv4. SSH for IPv6 provides CLI (console) access and secure file transfer functionality. The following types of transactions are supported:

- Client public-key authentication

  Public keys from SSH clients are stored on the switch. Access to the switch is granted only to a client whose private key matches a stored public key.

- Password-only client authentication

  The switch is SSH-enabled but is not configured with the login method that authenticates a client's public-key. Instead, after the switch authenticates itself to a client, users connected to the client authenticate themselves to the switch by providing a valid password that matches the operator- and/or manager-level password configured and stored locally on the switch or on a RADIUS or TACACS+ server.

- Secure Copy (SCP) and Secure FTP (SFTP)

  You can use an SCP or SFTP client application to perform secure file transfers to and from the switch.

## Configuring SSH for IPv6

By default, SSH is automatically enabled for IPv4 and IPv6 connections on a switch. As with SSH for IPv4, you can enter the **ip ssh** command to reconfigure the default SSH settings to:

- Restrict access to the SSH server running on the switch to only IPv4 or IPv6 clients.
- Modify the TCP port number and timeout period used in SSH authentication in IPv4 and IPv6 connections.

***Syntax:.*** [no] ip ssh

*Enables SSH on the switch and activates the connection with a configured SSH server (RADIUS or TACACS+). To disable SSH on the switch, enter the* **no ip ssh** *command.*

[ip-version < 4 | 6 | 4or6 >]

*IP version used for SSH connections on the switch:*
**4** *accepts SSH connections only from IPv4 clients.*
**6** *accepts SSH connections only from IPv6 clients.*
**4or6** *accepts SSH connections from either IPv4 or IPv6 clients. (Default:* **4or6***).*
*To disable SSH connections with IPv4 clients, enter the* **ip ssh ip-version 6** *command; to disable SSH connections with IPv6 clients, enter the* **ip ssh ip-version 4** *command.*

[port < 1-65535 | default >]

*TCP port number used for SSH sessions in IPv4 and IPv6 connections (Default: 22).*
*Valid port numbers are from 1 to 65535, except for port numbers 23, 49, 80, 280,443, 1506, 1513 and 9999, which are reserved for other subsystems.*

[timeout < 5 - 120 >]

*Timeout value allowed to complete an SSH authentication and login on the switch (Default: 120 seconds).*

[filetransfer]

*Enables SSH on the switch to connect to an SCP or SFTP client application to transfer files to and from the switch over IPv4 or IPv6.*
*For more information, see "Secure Copy and Secure FTP for IPv6" on page 6-18.*

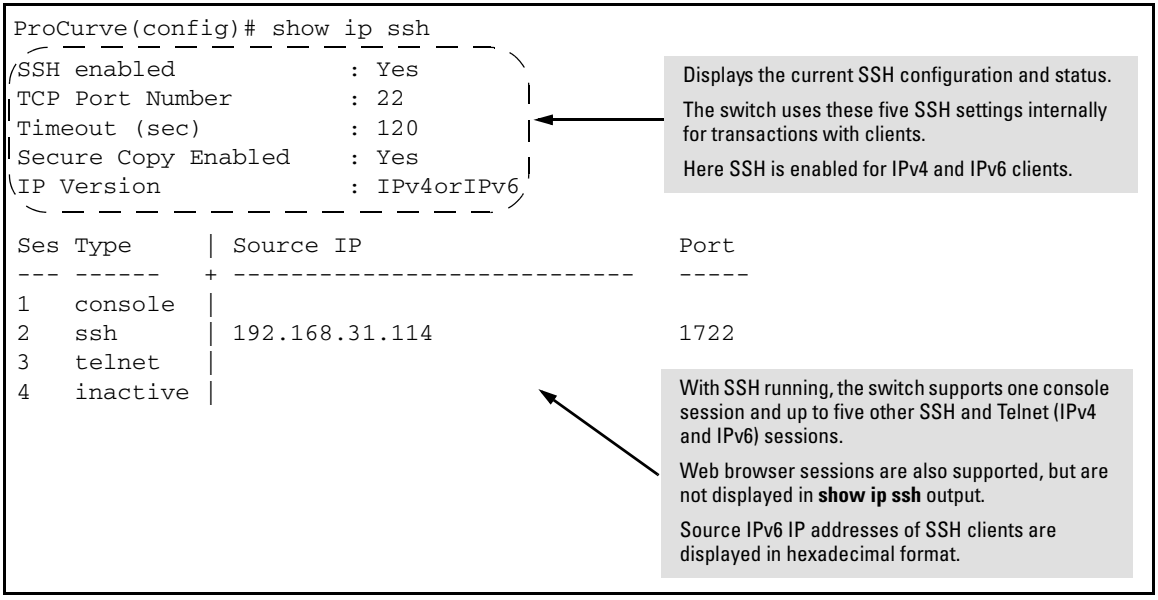
**Note**

As with IPv4, the switch only supports SSH version 2. You cannot set up an SSH session with a client device running SSH version 1.

For complete information on how to configure SSH for encrypted, authenticated transactions between the switch and SSH-enabled client devices, refer to the "*Configuring Secure Shell (SSH)*" chapter in the *Access Security Guide*.

## Displaying an SSH Configuration

To verify an SSH for IPv6 configuration and display all SSH sessions running on the switch, enter the **show ip ssh** command. Information on all current SSH sessions (IPv4 and IPv6) is displayed.



# Secure Copy and Secure FTP for IPv6

You can take advantage of the Secure Copy (SCP) and Secure FTP (SFTP) client applications to provide a secure alternative to TFTP for transferring sensitive switch information, such as configuration files and login information, between the switch and an administrator workstation.

SCP and SFTP run over an encrypted SSH session allowing you to use a secure SSH tunnel to:

- Transfer files and update ProCurve software images.
- Distribute new software images with automated scripts that make it easier to upgrade multiple switches simultaneously and securely.

By default, SSH is enabled for IPv4 and IPv6 connections on a switch. If you have not disabled SSH connections from IPv6 clients (by entering the **ip ssh ip-version 4** command), you can perform secure file transfers to and from IPv6 client devices by entering the **ip ssh filetransfer** command.

***Syntax:*** [no] ip ssh filetransfer

*Enables SSH on the switch to connect to an SCP or SFTP client application to transfer files to and from the switch.*

*Use the* **no ip ssh filetransfer** *command to disable the switch's ability to perform secure file transfers with an SCP or SFTP client, without disabling SSH on the switch.*

After an IPv6 client running SCP/SFTP successfully authenticates and opens an SSH session on the switch, you can copy files to and from the switch using secure, encrypted file transfers. Refer to the documentation that comes with an SCP or SFTP client application for information on the file transfer commands and software utilities to use.

**Notes**

The switch supports one SFTP session or one SCP session at a time.

All files on the switch have read-write permission. However, several SFTP commands, such as **create** or **remove**, are not supported and return an error message.

For complete information on how to configure SCP or SFTP in an SSH session to copy files to and from the switch, refer to the *"File Transfers"* appendix in the *Management and Configuration Guide* for your switch.

# Multicast Listener Discovery (MLD) Snooping

## Contents

# Overview

Multicast addressing allows one-to-many or many-to-many communication among hosts on a network. Typical applications of multicast communication include audio and video streaming, desktop conferencing, collaborative computing, and similar applications.

Multicast Listener Discovery (MLD) is an IPv6 protocol used on a local link for multicast group management. MLD is enabled per VLAN, and is analogous to the IPv4 IGMP protocol.

MLD snooping is a subset of the MLD protocol that operates at the port level and conserves network bandwidth by reducing the flooding of multicast IPv6 packets.

This chapter describes concepts of MLD snooping and the CLI commands available for configuring it and for viewing its status.

# Introduction to MLD Snooping

There are several roles that network devices may play in an IPv6 multicast environment:

- **MLD host**—a network node that uses MLD to "join" (subscribe to) one or more multicast groups
- **multicast router**—a router that routes multicast traffic between subnets
- **querier**—a switch or multicast router that identifies MLD hosts by sending out MLD queries, to which the MLD hosts respond

Curiously enough, a network node that acts as a *source* of IPv6 multicast traffic is only an indirect participant in MLD snooping—it just provides multicast traffic, and MLD doesn't interact with it. (Note, however, that in an application like desktop conferencing a network node may act as both a source and an MLD host; but MLD interacts with that node only in its role as an MLD host.)

A source node creates multicast traffic by sending packets to a multicast address. In IPv6, addresses with the first eight bits set (that is, "FF" as the first two characters of the address) are multicast addresses, and any node that listens to such an address will receive the traffic sent to that address. Application software running on the source and destination systems cooperates to determine what multicast address to use. (Note that this is a function of the application software, not of MLD.)

For example, if several employees engage in a desktop conference across the network, they all need application software on their computers. At the start of the conference, the software on all the computers determines a multicast address of, say, FF3E:30:2001:DB8::101 for the conference. Then any traffic sent to that address can be received by all computers listening on that address.

**General operation.** Multicast communication can take place without MLD, and by default MLD is disabled. In that case, if a switch receives a packet with a multicast destination address, it floods the packet to all ports in the same VLAN (except the port that it came in on). Any network nodes that are listening to that muticast address will see the packet; all other hosts ignore the packet.

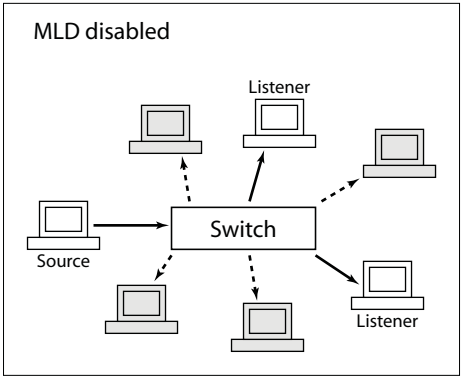



**Figure 7-1. Without MLD, multicast traffic is flooded to all ports.**

When MLD snooping is enabled on a VLAN, the switch acts to minimize unnecessary multicast traffic. If the switch receives multicast traffic destined for a given multicast address, it forwards that traffic only to ports on the VLAN that have MLD hosts for that address. It drops that traffic for ports on the VLAN that have no MLD hosts (except for a few special cases explained below).

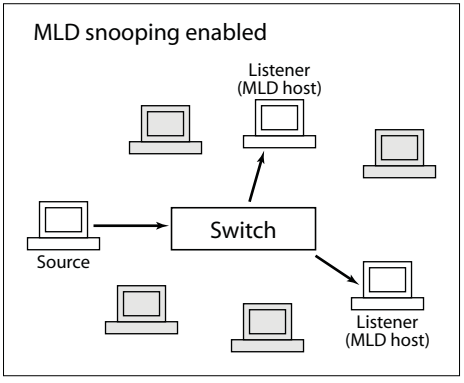



**Figure 7-2. With MLD snooping, traffic is sent to MLD hosts.**

Note that MLD snooping operates on a single VLAN (though there can be multiple VLANs, each running MLD snooping). Cross-VLAN traffic is handled by a multicast router.

**Forwarding in MLD snooping.** When MLD snooping is active, a multicast packet is handled by the switch as follows:

- forwarded to ports that have nodes that have joined the packet's multicast address (that is, MLD hosts on that address)
- forwarded toward the querier—If the switch is not the querier, the packet is forwarded out the port that leads to the querier.
- forwarded toward any multicast routers—If there are multicast routers on the VLAN, the packet is forwarded out any port that leads to a router.
- forwarded out administratively forwarded ports—The packet will be forwarded through all ports set administratively to forward mode. (See the description of forwarding modes, below.)
- dropped for all other ports

Each individual port's forwarding behavior can be explicitly set using a CLI command to one of these modes:

- auto (the default mode)—The switch forwards packets through this port based on the MLD rules and the packet's multicast address. In most cases, this means that the switch forwards the packet only if the port connects to a node that is joined to the packet's multicast address (that is, to an MLD host). There is seldom any reason to use a mode other than "auto" in normal operation (though some diagnostics may make use of "forward" or "block" mode).
- forward—The switch forwards all IPv6 multicast packets through the port. This includes IPv6 multicast data and MLD protocol packets.
- block—The switch drops all MLD packets received by the port and blocks all outgoing IPv6 multicast packets through the port, except those packets destined for well known IPv6 multicast addresses. This has the effect of preventing IPv6 multicast traffic from moving through the port.

Note that the switch floods all packets with "well known" IPv6 multicast destination addresses through all ports. Well known addresses are permanent addresses defined by the Internet Assigned Numbers Authority (www.iana.org). IPv6 standards define any address beginning with FF0x/12 (binary 1111 1111 0000) as a well known address.

**Listeners and joins.** The "snooping" part of MLD snooping arises because a switch must keep track of which ports have network nodes that are MLD hosts for any given multicast address. It does this by keeping track of "joins" on a per-port basis.

A network node establishes itself as an MLD host by issuing a multicast "join" request (also called a multicast "report") for a specific multicast address when it starts an application that listens to multicast traffic. The switch to which the node is connected sees the join request and forwards traffic for that multicast address to the node's port.

**Queries.** The querier is a multicast router or a switch that periodically asks MLD hosts on the network to verify their multicast join requests. There is one querier for each VLAN, and all switches on the VLAN listen to the responses of MLD hosts to multicast queries, and forward or block multicast traffic accordingly.

All of the ProCurve switches described by this guide have the querier function enabled by default. If there is another device on the VLAN that is already acting as querier, the switch defers to that querier. If there is no device acting as querier, the switch enters an election state and negotiates with other devices on the network (if any) to determine which one will act as the querier.

The querier periodically sends general queries to MLD hosts on each multicast address that is active on the VLAN. The time period that the querier waits between sending general queries is known as the query interval; the MLD standard sets the default query interval to 125 seconds.

Network nodes that wish to remain active as MLD hosts respond to the queries with join requests; in this way they continue to assert their presence as MLD hosts. The switch through which any given MLD host connects to the VLAN sees the join requests and continues forwarding traffic for that multicast address to the MLD host's port.

**Leaves.** A node acting as an MLD host can be disconnected from a multicast address in two ways:

- It can stop sending join requests to the querier. This might happen if the multicast application quits or the node is removed from the network. If the switch goes for slightly more than two query intervals without seeing a join request from the MLD host, it stops sending multicast traffic for that multicast address to the MLD host's port.
- It can issue a "leave" request. This is done by the application software running on the MLD host. If the MLD host is the only node connected to its switch port, the switch sees the leave request and stops sending multicast packets for that multicast address to that port. (If there is more than one node connected to the port the situation is somewhat more complicated, as explained below under "Fast leaves and forced fast leaves".)

**Fast leaves and forced fast leaves.** The fast leave and forced fast leave functions can help to prune unnecessary multicast traffic when an MLD host issues a leave request from a multicast address. Fast leave is enabled by default and forced fast leave is disabled by default. Both functions are applied to individual ports.

Which function to use depends on whether a port has more than one node attached to it, as follows:

- If a port has only one node attached to it, then when the switch sees a leave request from that node (an MLD host) it knows that it does not need to send any more multicast traffic for that multicast address to the host's port. If fast leave is enabled (the default setting), the switch stops sending the multicast traffic immediately. If fast leave is disabled, the switch continues to look for join requests from the host in response to group-specific queries sent to the port. The interval during which the switch looks for join requests is brief and depends on the forced fast leave setting: if forced fast leave is enabled for the port, it is equal to the "forced fast leave interval" (typically a couple of seconds or less); if forced fast leave is disabled for the port, the period is about 10 seconds (governed by the MLD standard). When this process has completed the multicast traffic for the group will be stopped (unless the switch sees a new join request).
- If there are multiple nodes attached to a single port, then a leave request from one of those nodes (an MLD host) does not provide enough information for the switch to stop sending multicast traffic to the port. In this situation the fast leave function does not operate. The switch continues to look for join requests from any MLD hosts connected to the port, in response to group-specific queries sent to the port. As in the case described above for a single-node port that is not enabled for fast leave, the interval during which the switch looks for join requests is brief and depends on the forced fast leave setting. If forced fast leave is enabled for the port, it is equal to the "forced fast leave interval" (typically a couple of seconds or less); if forced fast leave is disabled for the port, the period is about 10 seconds (governed by the MLD standard). When this process has completed the multicast traffic for the group will be stopped unless the switch sees a new join request. This reduces the number of multicast packets forwarded unnecessarily.

# Configuring MLD

Several CLI commands are available for configuring MLD parameters on a switch.

## Enabling or Disabling MLD Snooping on a VLAN

***Syntax:*** [no] ipv6 mld

*Note: This command must be issued in a VLAN context.*

*This command enables MLD snooping on a VLAN. Enabling MLD snooping applies the last-saved or the default MLD configuration, whichever was most recently set.*

*The* **[no]** *form of the command disables MLD snooping on a VLAN.*

*MLD snooping is disabled by default.*

For example, to enable MLD snooping on VLAN 8:

```
ProCurve# config
ProCurve(config)# vlan 8
ProCurve(vlan-8)# ipv6 mld
```

To disable MLD snooping on VLAN 8:

```
ProCurve(vlan-8)# no ipv6 mld
```

## Configuring Per-Port MLD Traffic Filters

***Syntax:*** ipv6 mld [auto <*port-list*> | blocked <*port-list*> | forward <*port-list*>]

***Note:*** *This command must be issued in a VLAN context.*

*This command sets per-port traffic filters, which specify how each port should handle MLD traffic. Allowed settings are:*

**auto**—*follows MLD snooping rules: packets are forwarded for joined groups*

**blocked**—*all multicast packets are dropped, except that packets for well known addresses are forwarded*

**forward**—*all multicast packets are forwarded*

*The default value of the filter is* **auto**.

<*port-list*>—*specifies the affected port or range of ports*

For example:

```
ProCurve(vlan-8)# ipv6 mld forward a16-a18
ProCurve(vlan-8)# ipv6 mld blocked a19-a21
ProCurve(vlan-8)# show ipv6 mld vlan 8 config

 MLD Service Vlan Config

  VLAN ID : 8
  VLAN NAME : VLAN8
  MLD Enabled [No] : Yes
  Querier Allowed [Yes] : Yes

  Port Type      | Port Mode Forced Fast Leave Fast Leave
  ---- --------- + --------- ----------------- ----------
  A13  100/1000T | auto      No                Yes
  A14  100/1000T | auto      No                Yes
  A15  100/1000T | auto      No                Yes
  A16  100/1000T | forward   No                Yes
  A17  100/1000T | forward   No                Yes
  A18  100/1000T | forward   No                Yes
  A19  100/1000T | blocked   No                Yes
  A20  100/1000T | blocked   No                Yes
  A21  100/1000T | blocked   No                Yes
  A22  100/1000T | auto      No                Yes
  A23  100/1000T | auto      No                Yes
  A24  100/1000T | auto      No                Yes
```

**Figure 7-3. Example of an MLD Configuration with Traffic Filters**

## Configuring the Querier

***Syntax:*** [no] ipv6 mld querier

> ***Note:*** *This command must be issued in a VLAN context.*
>
> *This command enables the switch to act as querier on a VLAN.*
>
> *The* **[no]** *form of the command disables the switch from acting as querier on a VLAN.*
>
> *The querier function is enabled by default. If another switch or a multicast router is acting as the MLD querier on the VLAN, this switch will defer to that device. If an acting querier stops performing the querier function, all querier-enabled switches and multicast routers on the VLAN will enter an election to determine the next device to act as querier.*

For example, to disable the switch from acting as querier on VLAN 8:

```
ProCurve(vlan-8)# no ipv6 mld querier
```

To enable the switch to act as querier on VLAN 8:

```
ProCurve(vlan-8)# ipv6 mld querier
```

## Configuring Fast Leave

***Syntax:*** [no] ipv6 mld fastleave <*port-list*>

> *Note: This command must be issued in a VLAN context.*
>
> *This command enables the fast leave function on the specified ports in a VLAN.*
>
> *The* **[no]** *form of the command disables the fast leave function on the specified ports in a VLAN.*
>
> *The fast leave function is enabled by default.*

For example, to disable fast leave on ports in VLAN 8:

```
ProCurve(vlan-8)# no ipv6 mld fastleave a14-a15
```

To enable fast leave on ports in VLAN 8:

```
ProCurve(vlan-8)# ipv6 mld fastleave a14-a15
```

## Configuring Forced Fast Leave

***Syntax:*** [no] ipv6 mld forcedfastleave <*port-list*>

***Note:*** *This command must be issued in a VLAN context.*

*This command enables the forced fast leave function on the specified ports in a VLAN.*

*The* **[no]** *form of the command disables the forced fast leave function on the specified ports in a VLAN.*

*The forced fast leave function is disabled by default.*

For example, to enable forced fast leave on ports in VLAN 8:

```
ProCurve(vlan-8)# ipv6 mld forcedfastleave a19-a20
```

To disable forced fast leave on ports in VLAN 8:

```
ProCurve(vlan-8)# no ipv6 mld forcedfastleave a19-a20
```

# Displaying MLD Status and Configuration

## Current MLD Status

***Syntax:*** show ipv6 mld

*Displays MLD status information for all VLANs on the switch that have MLD configured.*

show ipv6 mld vlan <vid>

*Displays MLD status for the specified VLAN*
**vid**—*VLAN ID*

For example, a switch with MLD snooping configured on VLANs 8 and 9 might show the following information:

```
ProCurve# show ipv6 mld

  MLD Service Protocol Info

  Total vlans with MLD enabled                : 2
  Current count of multicast groups joined    : 37

  VLAN ID : 8
  VLAN NAME : VLAN8
  Querier Address : fe80::218:71ff:fec4:2f00 [this switch]
  Querier Up Time : 1h:37m:20s
  Querier Expiry Time : 0h:1m:44s

  Ports with multicast routers :


  Active Group Addresses                    Type ExpiryTime Ports
  ----------------------------------------- ---- ---------- --------------------
  ff02::c                                   FILT 0h:4m:9s   A15-A21
  ff02::1:2                                 FILT 0h:4m:3s   A21
  ff02::1:3                                 FILT 0h:4m:9s   A15-A21
  ff02::1:ff00:42                           FILT 0h:4m:0s   A19
  ff02::1:ff02:2                            FILT 0h:4m:2s   A15
  ff02::1:ff02:3                            FILT 0h:4m:5s   A16
  ff02::1:ff03:2                            FILT 0h:4m:2s   A17
  ff02::1:ff03:3                            FILT 0h:4m:5s   A18
```

**Figure 7-4. Example of Displaying the MLD Configuration for All Static VLANs on the Switch**

```
 ff02::1:ff04:3                          FILT 0h:4m:5s   A20
 ff02::1:ff05:1                          FILT 0h:4m:3s   A21
 ff02::1:ff0b:2dfe                       FILT 0h:3m:59s  A17
 ff02::1:ff0b:d7d9                       FILT 0h:4m:4s   A15
 ff02::1:ff0b:da09                       FILT 0h:4m:5s   A18
 ff02::1:ff0b:dc38                       FILT 0h:4m:3s   A19
 ff02::1:ff0b:dc8d                       FILT 0h:4m:4s   A20
 ff02::1:ff0b:dd56                       FILT 0h:4m:0s   A16
 ff02::1:ff12:e0cd                       FILT 0h:4m:5s   A21
 ff02::1:ff4e:98a5                       FILT 0h:4m:0s   A17
 ff02::1:ff57:21a1                       FILT 0h:3m:58s  A20
 ff02::1:ff6b:dd51                       FILT 0h:4m:0s   A15
 ff02::1:ff7b:ac55                       FILT 0h:4m:5s   A16
 ff02::1:ff8f:61ea                       FILT 0h:4m:1s   A19
 ff02::1:ffc8:397b                       FILT 0h:4m:0s   A18
ff3e:30:2001:db8:8:0:7:101               FILT 0h:4m:4s   A15,A18,A21
 ff3e:30:2001:db8:8:0:7:102              FILT 0h:4m:13s  A16,A19

 VLAN ID : 9
 VLAN NAME : VLAN9
 Querier Address : fe80::218:71ff:fec4:2f00 [this switch]
 Querier Up Time : 1h:37m:22s
 Querier Expiry Time : 0h:1m:43s

Ports with multicast routers :

 Active Group Addresses                  Type ExpiryTime Ports
 --------------------------------------- ---- ---------- --------------------
 ff02::c                                 FILT 0h:4m:12s  B3,B5,B7
 ff02::1:3                               FILT 0h:4m:12s  B3,B5,B7
 ff02::1:ff02:4                          FILT 0h:4m:4s   B3
 ff02::1:ff03:4                          FILT 0h:3m:59s  B5
 ff02::1:ff04:4                          FILT 0h:4m:12s  B7
 ff02::1:ff0b:dc64                       FILT 0h:4m:0s   B7
 ff02::1:ff0b:dcf3                       FILT 0h:4m:2s   B3
 ff02::1:ff0b:dd5c                       FILT 0h:4m:4s   B5
 ff02::1:ff34:a69e                       FILT 0h:4m:1s   B5
 ff02::1:ff8e:11d5                       FILT 0h:3m:57s  B7
 ff02::1:ffea:2c4f                       FILT 0h:3m:58s  B3
```

**Figure 7-5. Continuation of Figure 7-4**

The following information is shown for each VLAN that has MLD snooping enabled:

- VLAN ID number and name
- Querier address: IPv6 address of the device acting as querier for the VLAN
- Querier up time: the length of time in seconds that the querier has been acting as querier
- Querier expiry time: If this switch is the querier, this is the amount of time until the switch sends the next general query. If this switch is not the querier, this is the amount of time in seconds until the current querier is considered inactive (after which a new querier election is held).
- Ports with multicast routers: ports on the VLAN that lead toward multicast routers (if any)
- Multicast group address information for each active group on the VLAN, including:
  - the multicast group address
  - the type of tracking for multicast joins: standard or filtered. If MLD snooping is enabled, port-level tracking results in filtered groups. If MLD snooping is not enabled, joins result in standard groups being tracked by this device. In addition, if hardware resources for multicast filtering are exhausted, new joins may result in standard groups even though MLD snooping is enabled.
  - expiry time: the time until the group expires if no joins are seen
  - the ports that have joined the multicast group

The group addresses you see listed typically result from several network functions. In our example, several of the addresses at the top of the list for each VLAN are IANA well known addresses (see www.iana.org/assignments/ipv6-multicast-addresses); the addresses in the form of ff02::1:ffxx:xxxx are solicited-node multicast addresses (used in IPv6 Neighbor Discovery); and the addresses beginning with ff3e are group addresses used by listeners to streaming video feeds.

## Current MLD Configuration

***Syntax:*** show ipv6 mld config

*Displays current global MLD configuration for all MLD-enabled VLANS on the switch.*

show ipv6 vlan *<vid>* config

*Displays current MLD configuration for the specified VLAN, including per-port configuration information.*
**vid**—*VLAN ID*

For example, the general form of the command might look like this:

```
ProCurve# show ipv6 mld config

 MLD Service Config

  Control unknown multicast  [Yes] : Yes
  Forced fast leave timeout  [4] : 4

  VLAN ID VLAN NAME       MLD Enabled Querier Allowed
  ------- --------------- ----------- ---------------
  8       VLAN8           Yes         Yes
  9       VLAN9           Yes         Yes
```

**Figure 7-6. Example of a Global MLD Configuration**

The following information, for all MLD-enabled VLANs, is shown:

- Control unknown multicast: If this is set to YES, any IPv6 multicast packets that are not joined by an MLD host will be sent only to ports that have detected a multicast router or ports that are administratively forwarded. If this is set to NO (or if MLD snooping is disabled), unjoined IPv6 multicast packets will be flooded out all ports in the VLAN.
- Forced fast leave timeout: the interval between an address specific query and a forced fast leave (assuming no response), in tenths of seconds
- For each VLAN that has MLD enabled:
  - VLAN ID and name
  - whether MLD is enabled on the VLAN (default NO, but the VLAN will not show up on this list unless MLD is enabled)
  - whether the switch can act as querier for the VLAN (default YES)

The specific form of the command might look like this:

```
ProCurve# show ipv6 mld vlan 8 config

 MLD Service Vlan Config

  VLAN ID : 8
  VLAN NAME : VLAN8
  MLD Enabled [No] : Yes
  Querier Allowed [Yes] : Yes

  Port Type      | Port Mode Forced Fast Leave Fast Leave
  ---- --------- + --------- ----------------- ----------
  A13  100/1000T | auto      No                Yes
  A14  100/1000T | auto      No                Yes
  A15  100/1000T | auto      No                Yes
  A16  100/1000T | auto      No                Yes
  A17  100/1000T | auto      No                Yes
  A18  100/1000T | auto      No                Yes
  A19  100/1000T | auto      No                Yes
  A20  100/1000T | auto      No                Yes
  A21  100/1000T | auto      No                Yes
  A22  100/1000T | auto      No                Yes
  A23  100/1000T | auto      No                Yes
  A24  100/1000T | auto      No                Yes
```

**Figure 7-7. Example of an MLD Configuration for a Specific VLAN**

The following information is shown, if the specified VLAN is MLD-enabled:

- VLAN ID and name
- whether MLD is enabled on the VLAN (default NO, but the information for this VLAN will be listed only if MLD is enabled)
- whether the switch is allowed to act as querier on the VLAN

## Ports Currently Joined

***Syntax:*** show ipv6 vlan *<vid>* group

*Lists the ports currently joined for all IPv6 multicast group addresses in the specified VLAN*
**vid**—*VLAN ID*

show ipv6 vlan *<vid>* group *<ipv6-addr>*

*Lists the ports currently joined for the specified IPv6 multicast group address in the specified VLAN*
**vid**—*VLAN ID*
**ipv6-addr**—*address of the IPv6 multicast group for which you want information*

For example, the general form of the command is shown below. The specific form the the command is similar, except that it lists the port information for only the specified group.

```
ProCurve# show ipv6 mld vlan 9 group

 MLD Service Protocol Group Info

 VLAN ID : 9
 VLAN Name : VLAN9

  Filtered Group Address : ff02::c
 Last Reporter : fe80::7061:4b38:dbea:2c4f
  ExpiryTime : 0h:2m:19s

  Port  Port Type  | Port Mode ExpiryTime
  ----  --------- + --------- --------------------
  B3    100/1000T | auto      0h:2m:19s
  B5    100/1000T | auto      0h:2m:18s

    .
    .
    .

 Filtered Group Address : ff3e:30:2001:db8:9:0:7:111
  Last Reporter : fe80::7061:4b38:dbea:2c4f
  ExpiryTime : 0h:4m:14s

  Port  Port Type  | Port Mode ExpiryTime
  ----  --------- + --------- --------------------
  B3    100/1000T | auto      0h:4m:14s
  B5    100/1000T | auto      0h:4m:09s
```

**Figure 7-8. Example of Ports Joined to Multicast Groups in a Specific VLAN**

The following information is shown:

- VLAN ID and name
- port information for each IPv6 multicast group address in the VLAN (general group command) or for the specified IPv6 multicast group address (specific group command):
  - group multicast address
  - last reporter: last MLD host to send a join to the group address
  - group expiry time: the time until the group expires if no further joins are seen
  - port name for each port
  - port type for each port: Ethernet connection type
  - port mode for each port: auto (follows MLD snooping rules; that is, packets are forwarded for joined groups), forward (all multicast packets are forwarded to this group), or blocked (all multicast packets are dropped, except that packets for well-known addresses are forwarded)
  - expiry time for each port: amount of time until this port is aged out of the multicast address group, unless a join is received

## Statistics

***Syntax:*** show ipv6 mld statistics

*Shows MLD statistics for all MLD-enabled VLANs*

***Syntax:*** show ipv6 mld vlan <vid> statistics

*Shows MLD statistics for the specified VLAN*
**vid**—*VLAN ID*

The general form the of the command shows the total number of MLD-enabled VLANs and a count of multicast groups currently joined. Both forms of the command show VLAN IDs and names, as well as the number of filtered and standard multicast groups and the total number of multicast groups.

For example, the general form of the command:

```
ProCurve# show ipv6 mld statistics

 MLD Service Statistics

  Total vlans with MLD enabled                : 2
  Current count of multicast groups joined    : 36


 MLD Joined Groups Statistics

  VLAN ID VLAN NAME    filtered     standard     total
  ------- ------------ ------------ ------------ ------------
  8       VLAN8        26           0            26
  9       VLAN9        10           0            10
```

**Figure 7-9. Example of MLD Statistics for All VLANs Configured**

And the specific form of the command:

```
ProCurve# show ipv6 mld vlan 8 statistics

 MLD Statistics

  VLAN ID : 8
  VLAN NAME : VLAN8

  Number of Filtered Groups     : 26
  Number of Standard Groups     : 0
  Total Multicast Groups Joined : 26
```

**Figure 7-10. Example of MLD Statistics for a Single VLAN**

## Counters

***Syntax:*** show ipv6 mld vlan *<vid>* counters

*Displays MLD counters for the specified VLAN*
**vid**—*VLAN ID*

```
ProCurve# show ipv6 mld vlan 8 counters

 MLD Service Vlan Counters

  VLAN ID : 8
  VLAN NAME : VLAN8

     General Query Rx                    : 2
     General Query Tx                    : 0
     Group Specific Query Rx             : 0
     Group Specific Query Tx             : 0
     V1 Member Report Rx                 : 1589
     V2 Member Report Rx                 : 15
     Leave Rx                            : 30
     Unknown MLD Type Rx                 : 0
     Unknown Pkt Rx                      : 0
     Forward to Routers Tx Counter       : 83
     Forward to Vlan Tx Counter          : 48
     Port Fast Leave Counter             : 4
     Port Forced Fast Leave Counter      : 0
     Port Membership Timeount Counter    : 28
```

**Figure 7-11. Example of MLD Counters for a Single VLAN**

The following information is shown:

- VLAN number and name
- For each VLAN:
  - number of general queries received
  - number of general queries sent
  - number of group-specific queries received
  - number of group-specific queries sent
  - number of MLD version 1 member reports (joins) received
  - number of MLD version 2 member reports (joins) received
  - number of leaves received
  - number of MLD packets of unknown type received
  - number of packets of unknown type received
  - number of packets forwarded to routers on this VLAN
  - number of times a packet has been forwarded to all ports on this VLAN
  - number of fast leaves that have occurred
  - number of forced fast leaves thathave occurred
  - number of times a join has timed out on this VLAN

# 8

# IPv6 Diagnostic and Troubleshooting

## Contents

# Introduction

| Feature | Default | CLI |
|---|---|---|
| IPv6 ICMP Message Interval and Token Bucket | 100 ms<br>10 max tokens | 8-3 |
| ping6 | Enabled | |
| traceroute6 | n/a | |

The IPv6 ICMP feature enables control over the error and informational message rate for IPv6 traffic, which can help mitigate the effects of a Denial-of-service attack. Ping6 enables verification of access to a specific IPv6 device, and traceroute6 enables tracing the route to an IPv6-enabled device on the network.

# ICMP Rate-Limiting

ICMP rate-limiting controls the rate at which ICMPv6 generates error and informational messages for features such as:

- neighbor solicitations
- neighbor advertisements
- multicast listener discovery (MLD)
- path MTU discovery (PMTU)
- duplicate address discovery (DAD)
- neighbor unreachability detection (NUD)
- router discovery
- neighbor discovery (NDP)

ICMPv6 error message generation is enabled by default. The rate of message generation can be adjusted, or message generation can be disabled.

Controlling the frequency of ICMPv6 error messages can help to prevent DoS (Denial- of- Service) attacks. With IPv6 enabled on the switch, you can control the allowable frequency of these messages with ICMPv6 rate-limiting.

***Syntax:.*** ipv6 icmp error-interval < 0 - 2147483647 > [bucket-size < 1 - 200 >]
no ipv6 icmp error-interval

*This command is executed from the global configuration level, and uses a "token bucket" method for limiting the rate of ICMP error and informational messages. Using this method, each ICMP message uses one token, and a message can be sent only if there is a token available. In the default configuration, a new token can be added every 100 milliseconds, and a maximum of 10 tokens are allowed in the token bucket. If the token bucket is full, a new token cannot be added until an existing token is used to enable sending an ICMP message. You can increase or decrease both the the frequency with which used tokens can be replaced and (optionally) the number of tokens allowed to exist.*

**error-interval:** *Specifies the time interval in milliseconds between successive token adds. Increasing this value decreases the rate at which tokens can be added. A setting of* **0** *disables ICMP messaging.*
***Default:*** *100;* ***Range:*** *0 - 2147483647.*

**bucket-size:** *This optional keyword specifies the maximum number of tokens allowed in the token bucket at any time. Decreasing this value decreases the maximum number of tokens that may be available at any time.*
***Default:*** *10;* ***Range:*** *1 - 200.*

*You can change the rate at which ICMP messages are allowed by changing the error-interval with or without a corresponding change in the bucket-size.*

*The* **no ipv6 icmp error-interval** *command resets both the* **error-interval** *and the* **bucket-size** *values to their defaults.*

*Use the* **show run** *command to view the current ICMP error interval settings.*

For example, the following command limits ICMP error and informational messages to no more than 20 every 1 second:

```
ProCurve(config)# ipv6 icmp error-interval 1000000 bucket-size
20
```

# Ping for IPv6 (Ping6)

The Ping6 test is a point-to-point test that accepts an IPv6 address or IPv6 host name to see if an IPv6 switch is communicating properly with another device on the same or another IP network. A ping test checks the path between the switch and another device by sending IP packets (ICMP Echo Requests).

To use a **ping6** command with an IPv6 host name or fully qualified domain names, refer to "DNS Resolver for IPv6" on page 8-9.

You can issue single or multiple ping tests with varying repetitions and timeout periods to wait for a ping reply.

Replies to each ping test are displayed on the console screen. To stop a ping test before it finishes, press **[Ctrl] [C]**.

For more information about using a ping test, refer to the "Troubleshooting" appendix in the current *Management and Configuration Guide* for your switch.

***Syntax:*** ping6 < *ipv6-address* | *hostname* | *switch-number* >
[repetitions < 1 - 10000 >] [timeout < 1 - 60 >] [data-size < 0 - 65507 >]
[data-fill < 0 - 1024 >]
ping6 <*link-local-address*%vlan<*vid*> | *hostname* | *switch-number*>
[repetitions < 1 - 10000 >] [timeout < 1 - 60 >] [data-size < 0 - 65507 >]
[data-fill < 0 - 1024 >]

*Pings the specified IPv6 host by sending ICMP version 6 (ICMPv6) echo request packets to the specified host.*

**<*ipv6-address*>**: *IPv6 address of a destination host device.*

**< *link-local-address* >%vlan**<*vlan-id*>: *IPv6 link-local address, where* **%vlan**<*vlan-id*> *specifies the VLAN ID number.*

**< *hostname* >:** *Host name of an IPv6 host device configured on an IPv6 DNS server.*

**< *switch-number* >:** *Number of an IPv6-based switch that is a member of a switch stack (IPv6 subnet). Valid values: 1 - 16.*

**[repetitions]**: *Number of times that IPv6 ping packets are sent to the destination IPv6 host. Valid values: 1 - 10000. Default: 1.*

**[timeout]**: *Number of seconds within which a response is required from the destination host before the ping test times out. Valid values: 1 - 60. Default: 1 second.*

**[data-size]**: *Size of data (in bytes) to be sent in ping packets. Valid values: 0 - 65507. Default: 0.*

**[data-fill]**: *Text string used as data in ping packets. You can enter up to 1024 alphanumeric characters in the text.* ***Default:*** **0** *(no text is used).*

```
ProCurve# ping6 fe80::2:1%vlan10
fe80:0000:0000:0000:0000:0000:0002:0001 is alive, time = 975 ms

ProCurve# ping6 2001:db8::a:1c:e3:3 repetitions 3
2001:0db8:0000:0000:000a:001c:00e3:0003 is alive, iteration 1, time = 15 ms
2001:0db8:0000:0000:000a:001c:00e3:0003 is alive, iteration 2, time = 15 ms
2001:0db8:0000:0000:000a:001c:00e3:0003 is alive, iteration 3, time = 15 ms
3 packets transmitted, 3 packets received, 0% packet loss
round-trip (ms) min/avg/max = 15/15/15

ProCurve# ping6 2001:db8::214:c2ff:fe4c:e480 repetitions 3 timeout 2
2001:db8:0000:0000:0214:c2ff:fe4c:e480 is alive, iteration 1, time = 15 ms
2001:db8:0000:0000:0214:c2ff:fe4c:e480 is alive, iteration 2, time = 10 ms
2001:db8:0000:0000:0214:c2ff:fe4c:e480 is alive, iteration 3, time = 15 ms

ProCurve# ping6 2001:db8::10
Request timed out.
```

**Figure 8-1. Examples of IPv6 Ping Tests**

# Traceroute for IPv6

The **traceroute6** command enables you to trace the route from a switch to a host device that is identified by an IPv6 address or IPv6 host name. In the command output, information on each (router) hop between the switch and the destination IPv6 address is displayed.

To use a **traceroute6** command with an IPv6 host name or fully qualified domain names, refer to "DNS Resolver for IPv6" on page 8-9.

Note that each time you perform a traceroute operation, the **traceroute** command uses the default settings unless you enter different values with each instance of the command.

Replies to each traceroute operation are displayed on the console screen. To stop a traceroute operation before it finishes, press **[Ctrl] [C]**.

For more information about how to configure and use a traceroute operation, refer to the "Troubleshooting" appendix in the *Management and Configuration Guide*.

***Syntax:*** traceroute6 < *ipv6-address* | *hostname* >
[minttl < 1-255 > [maxttl < 1-255 > [timeout < 1 - 60 >] [probes < 1-5 >]
traceroute6 <*link-local-address*%vlan<*vid*> | *hostname* >
[minttl < 1-255 >] [maxttl < 1-255 >] [timeout < 1 - 60 >] [probes < 1-5 >]

*Displays the IPv6 address of each hop in the route to the specified destination host device with the time (in microseconds) required for a packet reply to be received from each next-hop device.*

**<*ipv6-address*>:** *IPv6 address of a destination host device.*

**<*link-local-address*>%vlan<*vlan-id*>:** *IPv6 link-local address, where* **%vlan<*vlan-id*>** *specifies the VLAN ID number.*

**<*hostname*>:** *Host name of an IPv6 host device configured on an IPv6 DNS server.*

**minttl:** *Minimum number of hops allowed for each probe packet sent along the route.* ***Default:*** *1;* ***Range:*** *1 - 255.*

- *If the* **minttl** *value is greater than the actual number of hops, the traceroute output displays only the hops equal to or greater than the configured* **minttl** *threshold value. The hops below the threshold value are not displayed.*
- *If the* **minttl** *value is the same as the actual number of hops, only the final hop is displayed in the command output.*
- *If the* **minttl** *value is less than the actual number of hops, all hops to the destination host are displayed.*

**maxttl**: *Maximum number of hops allowed for each probe packet sent along the route. Valid values: 1 - 255. Default: 30.*

- *If the* **maxttl** *value is less than the actual number of hops required to reach the host, the traceroute output displays only the IPv6 addresses of the hops detected by the configured* **maxttl** *value.*

**timeout**: *Number of seconds within which a response is required from the IPv6 device at each hop in the route to the destination host before the traceroute operation times out.* ***Default:*** *5 seconds;* ***Range:*** *1 - 60.*

**probes**: *Number of times a traceroute is performed to locate the IPv6 device at any hop in the route to the specified host before the operation times out.* ***Default:*** *3;* ***Range:*** *1 - 5.*

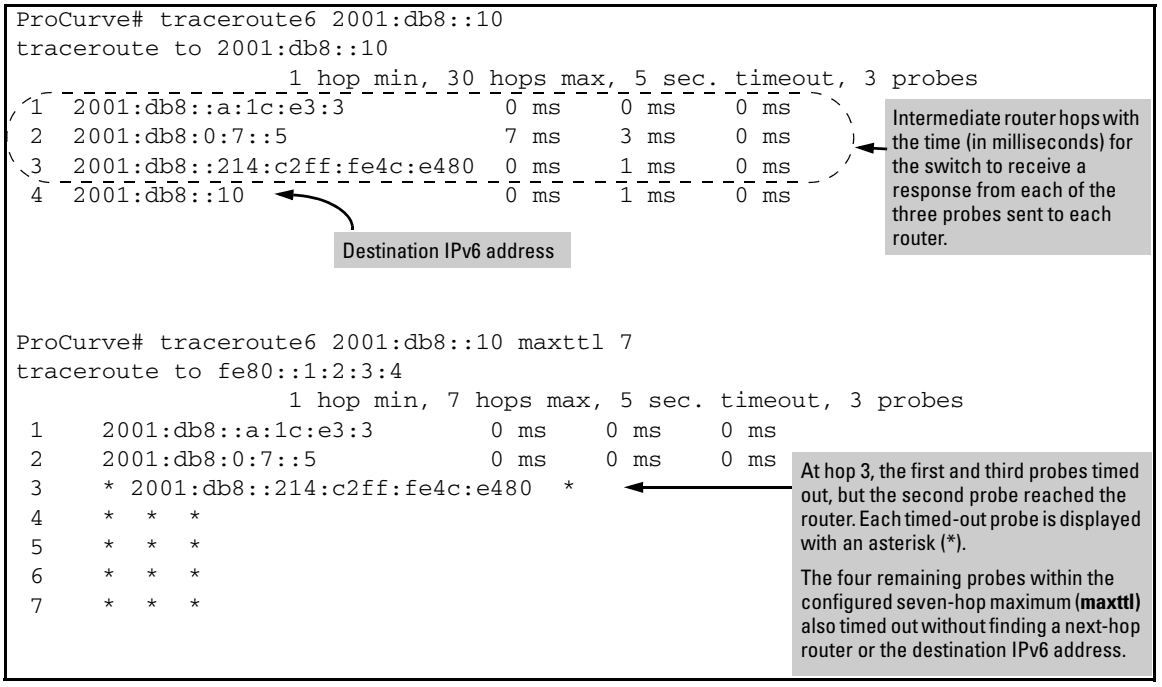



**Figure 8-2. Examples of IPv6 Traceroute Probes**

# DNS Resolver for IPv6

The Domain Name System (DNS) resolver is designed for local network domains where it enables use of a host name or fully qualified domain name to support DNS-compatible commands from the switch. Beginning with software release K.13.01,DNS operation supports these features:

- dual-stack operation: IPv6 and IPv4 DNS resolution
- DNS-compatible commands: **ping**, **ping6**, **traceroute**, and **traceroute6**
- multiple, prioritized DNS servers (IPv4 and IPv6)

## DNS Configuration

Up to three DNS servers can be configured. The addresses must be prioritized, and can be for any combination of IPv4 and IPv6 DNS servers.

**Note**

This section describes the commands for configuring DNS operation for IPv6 DNS applications. For further information and examples on using the DNS feature, refer to “DNS Resolver” in appendix C, “Troubleshooting”, in the current *Management and Configuration Guide* for your switch.

***Syntax:.*** [no] ip dns server-address priority < 1 - 3 > < *ip-addr* >

*Used at the global config level to configure the address and priority of a DNS server. Allows for configuring up to three servers providing DNS service. (The servers must all be accessible to the switch.) The command allows both IPv4 and IPv6 servers in any combination and any order of priority.*

**priority < 1 - 3 >:** *Identifies the order in which the specified DNS server will be accessed by a DNS resolution attempt. A resolution attempt tries each configured DNS server address, in ascending order of priority, until the attempt is successful or all configured server options have been tried and failed. To change the priority of an existing server option, you must remove the option from the switch configuration and re-enter it with the new priority. If another server address is configured for the new priority, you must also remove that address from the configuration before re-assigning its priority to another address.*

*— Continued on the next page. —*

*— Continued from the previous page. —*

*The* **no** *form of the command removes the specified address from the server address list configured on the switch.*

**< *ip-addr*>:** *Specifies the address of an IPv6 or IPv4 DNS server.*

***Syntax:.*** [no] ip dns domain-name < *domain-name-suffix* >

*Used at the global config level to configure the* domain suffix *that is automatically appended to the host name entered with a command supporting DNS operation. Configuring the domain suffix is optional if you plan to use fully qualified domain names in all cases instead of just entering host names.*

*You can configure up to three addresses for DNS servers in the same or different domains. However, you can configure only one domain name suffix. This means that a fully qualified domain name must be used to resolve addresses for hosts that do not reside in the same domain as the one you configure with this command. That is, if the domain name suffix and the address of a DNS server for that same domain are both configured on the switch, then you need to enter only the host name of the desired target when executing a command that supports DNS operation. But if the DNS server used to resolve the host name for the desired target is in a different domain than the domain configured with this command, then you need to enter the fully qualified domain name for the target.*

*The* **no** *form of the command removes the configured domain name suffix.*

For example, suppose you want to configure the following on the switch:

- the address **2001:db8::127:10** which identifies a DNS server in the domain named mygroup.procurve.net
- a priority of 1 for the above server
- the domain suffix **mygroup.procurve.net**

Assume that the above, configured DNS server supports an IPv6 device having a host name of “mars-1” (and an IPv6 address of fe80::215:60ff:fe7a:adc0) in the “mygroup.procurve.net” domain. In this case you can use the device's host name alone to ping the device because the mygroup.procurve.net domain has

been configured as the domain name on the switch and the address of a DNS server residing in that domain is also configured on the switch. The commands for these steps are as follows:

```
ProCurve(config)# ip dns server priority 1 2001:db8::127:10
ProCurve(config)# ip dns domain-name mygroup.procurve.net
ProCurve(config)# ping6 mars-1
fe80::215:60ff:fe7a:adc0 is alive, time = 1 ms
```

**Figure 8-1. Example of Configuring for a Local DNS Server and Pinging a Registered Device**

However, for the same "mars-1" device, if mygroup.procurve.net was not the configured domain name, you would have to use the fully qualified domain name for the device named mars-1:

```
ProCurve# ping6 mars-1.mygroup.procurve.net
```

For further information and examples on using the DNS feature, refer to "DNS Resolver" in appendix C, "Troubleshooting", in the current *Management and Configuration Guide* for your switch.

## Viewing the Current Configuration

Use the **show ip dns** command to view the current DNS server configuration.

Use the **show run** command to view both the current DNS server addresses and the current DNS domain name in the active configuration.

## Operating Notes

In software release K.13.01, DNS addressing is not configurable from a DHCPv6 server.

# Debug/Syslog for IPv6

The Debug/System logging (*Syslog*) for IPv6 feature provides the same logging functions as the IPv4 version, allowing you to record IPv4 and IPv6 Event Log and debug messages on a remote device to troubleshoot switch or network operation. For example, you can send messages about routing misconfigurations and other network protocol details to an external device, and later use them to debug network-level problems.

## Configuring Debug and Event Log Messaging

To specify the types of debug and Event Log messages that you want to send to an external device:

- Use the **debug** < *debug-type* > command to send messaging reports for the following types of switch events:
  - ACL “deny” matches
  - DHCP snooping events
  - Dynamic ARP protection events
  - Events recorded in the switch’s Event Log
  - IPv4 OSPF and RIP routing events
  - IPv6 DHCPv6 client and Neighbor Discovery events
  - LLDP events
- Use the **logging** < **severity** *severity-level* | **system-module** *system-module*> command to select a subset of Event Log messages to send to an external device for debugging purposes according to:
  - Severity level
  - System module

# Debug Command

***Syntax:*** [no] debug < *debug-type* >

*Configures the types of IPv4 and IPv6 messages that are sent to Syslog servers or other debug destinations, where* <*debug-type*> *is any of the following event types:*

acl

*When a match occurs on an ACL "deny" statement with a* **log** *parameter, an ACL message is sent to configured debug destinations. (Default: Disabled - ACL messages for traffic that matches "deny" entries are not sent.)*

all

*Configures all IPv4 and IPv6 debug message types to be sent to configured Idebug destinations. (Default: Disabled - No debug messages are sent.)*

arp-protect

*Configures messages for Dynamic ARP Protection events to be sent to configured debug destinations. (Default: Disabled - No debug messages are sent.)*

event

*Configures Event Log messages to be sent to configured debug destinations.*

*Event Log messages are enabled to be automatically sent to debug destinations in the following conditions:*

- *If no Syslog server address is configured and you enter the* **logging** *command to configure a destination address.*
- *If at least one Syslog server address is configured in the startup configuration and the switch is rebooted or reset.*

*Event log messages are the default type of debug message sent to configured debug destinations.*

ip

*Configures IPv4 OSPF and RIP routing messages to be sent to configured debug destinations.*

***Syntax:.*** [no] debug < *debug-type* > *(Continued)*

ip [ ospf < adj | event | flood | lsa-generation | packet | retransmission | spf > ]

*Configures specified IPv4 OSPF message types to be sent to configured debug destinations:*
**adj** — *Adjacency changes.*
**event** — *OSPF events.*
**flood** — *Information on flood messages.*
**lsa-generation** — *New LSAs added to database.*
**packet** — *Packets sent/received.*
**retransmission** — *Retransmission timer messages.*
**spf** — *Path recalculation messages*

ip [ rip < database | event | trigger > ]

*Configures specified IPv4 RIP message types to be sent to configured debug destinations:*
**database**— *Database changes*
**event**— *RIP events*
**trigger**— *Trigger messages*

ipv6

*Configures messages for IPv6 DHCPv6 client and neighbor discovery events to be sent to configured debug destinations.*

ipv6 [ dhcpv6-client <events | packets> | nd ]

*Configures one of the following IPv6 message types to be sent to configured debug destinations:*
**dhcpv6-clients events** — *DHCPv6 client events*
**dhcpv6-clients packets** — *Statistics on DHCPv6 packets transmitted on a switch configured as a DHCPv6 client*
**nd**— *Events during IPv6 neighbor discovery*

lldp

*Configures all LLDP message types to be sent to configured debug destinations.*

wireless-services

*Configures messages about the operation of wireless-services modules to be sent to configured debug destinations.*

## Configuring Debug Destinations

A Debug/Syslog destination device can be a Syslog server (up to six maximum) and/or a console session:

- Use the **debug destination** < **logging** | **session** | **buffer** > command to enable (and disable) Syslog messaging on a Syslog server or to a CLI session for the debug message types configured with the **debug** and **logging** commands (see "Configuring Debug and Event Log Messaging" on page 8-12):
  - **debug destination logging** enables the configured debug message types to be sent to Syslog servers configured with the **logging** command.
  - **debug destination session** enables the configured debug message types to be sent to the CLI session that executed this command. The session can be on any one terminal emulation device with serial, Telnet, or SSH access to the CLI at the Manager level prompt.
  - **debug destination buffer** enables the configured debug message types to be sent to a buffer in switch memory.

## Logging Command

***Syntax:*** [no] logging < *syslog-ipv4-addr* >

*Enables or disables Syslog messaging to the specified IPv4 address. You can configure up to six addresses. If you configure an address when none are already configured, this command enables destination logging (Syslog) and the Event debug type. Therefore, at a minimum, the switch begins sending Event Log messages to configured Syslog servers. If other debug message types are configured, they are also sent to the Syslog server.*

**no logging** *removes all currently configured Syslog logging destinations from the running configuration.*

**no logging** < *syslog-ipv4-address* > *removes only the specified Syslog logging destination from the running configuration.*

***Note:*** *The* **no logging** *command does not delete the Syslog server addresses stored in the startup configuration. To delete Syslog addresses in the startup configuration, you must enter the* **no logging** *command followed by the* **write memory** *command. To verify the deletion of a Syslog server address, display the startup configuration by entering the* **show config** *command.*

*To block the messages sent to configured Syslog servers from the currently configured debug message type, enter the* **no debug** < *debug-type* > *command.*

*To disable Syslog logging on the switch without deleting configured server addresses, enter the* **no debug destination logging** *command.*

For complete information on how to configure a Syslog server and Debug/Syslog message reports, refer to the “Troubleshooting” appendix in the *Management and Configuration Guide.*

# Terminology

| | |
|---|---|
| **DAD** | Duplicate Address Detection. Refer to “Duplicate Address Detection (DAD)” on page 4-18. |
| **Device Identifier** | The low-order bits in an IPv6 address that identify a specific device. For example, in the link-local address 2001:db8:a10:101:212:79ff:fe88:a100/64, the bits forming 212:79ff:fe88:a100 comprise the device identifier. |
| **DoS** | Denial-of-Service. |
| **EUI-64** | Extended Unique Identifier. Refer to “Extended Unique Identifier (EUI)” on page 3-14. |
| **Manual Address Configuration** | Configures an IPv6 address by using the CLI to manually enter a static address. Referred to as “Static Address Configuration” in this guide. See **Static Address Configuration**, below. |
| **MLD** | Multicast Listener Discovery. Refer to the chapter titled “Multicast Listener Discovery (MLD) Snooping”. |
| **MTU** | Maximum Transmission Unit. The largest frame size allowed on a given path or device. Refer to “Path MTU (PMTU) Discovery” on page 2-16. |
| **RA** | Router Advertisement. Refer to “Router Advertisements” on page 4-27. |
| **SLAAC** | Stateless Address Autoconfiguration. Refer to “SLAAC (Stateless Automatic Address Configuration)” on page 2-7. |
| **Static Address** | A permanently configured IPv6 address, as opposed to an autoconfigured address. |
| **Static Address Configuration** | Configures an IPv6 address by using the CLI to manually enter the address instead of using an automatically generated or DHCPv6-assigned address. Same as “Manual Address Configuration”. See also **Manual Address Configuration**, above. |

# Index

## Symbols



## A



## B



## C

## D



## E



## F

## G



## I

## L



## M

## T

## U



## V



## W




January 2008

Manual Part Number
5992-3067